UNIVERSIDADE FEDERAL DO CEARÁ
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE CIÊNCIA DA COMPUTAÇÃO

FELIPE TIMBÓ BRITO

UMA ABORDAGEM DISTRIBUÍDA PARA PRESERVAÇÃO DE PRIVACIDADE NA PUBLICAÇÃO DE DADOS DE TRAJETÓRIA

FORTALEZA, CEARÁ

2015

FELIPE TIMBÓ BRITO

UMA ABORDAGEM DISTRIBUÍDA PARA PRESERVAÇÃO DE PRIVACIDADE NA PUBLICAÇÃO DE DADOS DE TRAJETÓRIA

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Ceará, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Área de concentração: Banco de Dados

Orientador: Prof. Dr. Javam de Castro Machado

FORTALEZA, CEARÁ

2015

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca de Ciências e Tecnologia

B875a Brito, Felipe Timbó.
Uma abordagem distribuída para preservação de privacidade na publicação de dados de trajetória. / Felipe Timbó Brito. – 2016.
66 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Departamento de Computação, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2016.
Área de Concentração: Banco de dados
Orientação: Prof. Dr. Javam de Castro Machado.

1. Computadores – Medidas de segurança. 2. Tecnologia da informação. I. Título.

CDD 005

FELIPE TIMBÓ BRITO

UMA ABORDAGEM DISTRIBUÍDA PARA PRESERVAÇÃO DE PRIVACIDADE NA PUBLICAÇÃO DE DADOS DE TRAJETÓRIA

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Ceará, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação. Área de concentração: Banco de Dados

Aprovada em: __/__/____

BANCA EXAMINADORA

Prof. Dr. Javam de Castro Machado
Universidade Federal do Ceará - UFC
Orientador

Prof. Dr. João Eduardo Ferreira
Universidade de São Paulo - USP

Prof. Dr. Jose Antonio Fernandes de Macedo
Universidade Federal do Ceará - UFC

A todos aqueles que contribuíram direta e indiretamente para que eu superasse mais esse desafio.

AGRADECIMENTOS

Agradeço primeiramente a Deus por me conceder o dom da vida e por ter me dado força, perseverança e sabedoria para eu conseguir vencer mais esta etapa em minha vida profissional;

Agradeço imensamente à minha noiva que tanto amo, Isabelle, por compartilhar momentos difíceis e por estar sempre ao meu lado durante toda essa conquista;

Agradeço à minha família pelo apoio, amor e incentivo durante toda a elaboração deste trabalho;

Agradeço em especial ao meu orientador e amigo, Prof. Javam Machado, pela confiança, dedicação, orientação, e principalmente pela paciência em momentos que tive de conciliar atividades de mestrado com o meu trabalho no LSBD;

Agradeço aos professores José Macedo e João Eduardo pela participação na minha banca de defesa de mestrado;

Agradeço à Profa. Rosélia pelos ensinamentos e conselhos partilhados durante toda esta etapa que me fizeram refletir muitas vezes em tomadas de decisões importantes;

Agradeço ao LSBD por ter fornecido uma estrutura bastante satisfatória para minha pesquisa, como também pelo apoio financeiro para participação em congressos científicos;

Agradeço a todos os meus amigos pela compreensão em muitas vezes não poder estar presente nos momentos importantes, devido ao tempo dedicado a este trabalho;

Agradeço aos meus colegas de mestrado e doutorado da UFC por compartilharem alegrias e tristezas durante as disciplinas;

Agradeço a todos os colaboradores do LSBD pela partilha diária e pelo aprendizado profissional e pessoal visível em cada um;

Por fim, agradeço ainda aquelas pessoas que contribuíram de uma forma indireta para a realização deste trabalho.

“Tudo começa pequeno nessa vida; E só cresce se o permitimos!”

(*Pe. Fábio de Melo*)

RESUMO

Avanços em técnicas de computação móvel aliados à difusão de serviços baseados em localização têm gerado uma grande quantidade de dados de trajetória. Tais dados podem ser utilizados para diversas finalidades, tais como análise de fluxo de tráfego, planejamento de infraestrutura, entendimento do comportamento humano, etc. No entanto, a publicação destes dados pode levar a sérios riscos de violação de privacidade. Semi-identificadores são pontos de trajetória que podem ser combinados com informações externas e utilizados para identificar indivíduos associados à sua trajetória. Por esse motivo, analisando semi-identificadores, um usuário malicioso pode ser capaz de restaurar trajetórias anonimizadas de indivíduos por meio de aplicações de redes sociais baseadas em localização, por exemplo. Muitas das abordagens já existentes envolvendo anonimização de dados foram propostas para ambientes de computação centralizados, assim elas geralmente apresentam um baixo desempenho para anonimizar grandes conjuntos de dados de trajetória. Neste trabalho propomos uma estratégia distribuída e eficiente que adota o modelo de privacidade k^m-anonimato e utiliza o escalável paradigma MapReduce, o qual permite encontrar semi-identificadores em um grande volume de dados. Nós também apresentamos uma técnica que minimiza a perda de informação selecionando localizações chaves a serem removidas a partir do conjunto de semi-identificadores. Resultados de avaliação experimental demonstram que nossa solução de anonimização é mais escalável e eficiente que trabalhos já existentes na literatura.

Keywords: Preservação de Privacidade. Dados de Trajetória. Anonimização. MapReduce.

ABSTRACT

Advancements in mobile computing techniques along with the pervasiveness of location-based services have generated a great amount of trajectory data. These data can be used for various data analysis purposes such as traffic flow analysis, infrastructure planning, understanding of human behavior, etc. However, publishing this amount of trajectory data may lead to serious risks of privacy breach. Quasi-identifiers are trajectory points that can be linked to external information and be used to identify individuals associated with trajectories. Therefore, by analyzing quasi-identifiers, a malicious user may be able to trace anonymous trajectories back to individuals with the aid of location-aware social networking applications, for example. Most existing trajectory data anonymization approaches were proposed for centralized computing environments, so they usually present poor performance to anonymize large trajectory data sets. In this work we propose a distributed and efficient strategy that adopts the k^m-anonymity privacy model and uses the scalable MapReduce paradigm, which allows finding quasi-identifiers in larger amount of data. We also present a technique to minimize the loss of information by selecting key locations from the quasi-identifiers to be suppressed. Experimental evaluation results demonstrate that our proposed approach for trajectory data anonymization is more scalable and efficient than existing works in the literature.

Keywords: Privacy-Preserving. Trajectory Data. Anonymity. MapReduce.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

Recentemente, devido ao enorme crescimento de dispositivos que utilizam tecnologia GPS, serviços baseados em localização tornaram-se cada vez mais comuns em domínios sociais e empresariais (HU et al., 2013). Estes numerosos serviços, tais como navegação, redes sociais, serviços de recomendação, entre outros, têm sido desenvolvidos e integrados às atividades diárias das pessoas, provendo informações bastante úteis sobre seus arredores e sendo capazes de responder perguntas do dia a dia como: qual a melhor rota a ser percorrida para um determinado endereço? Quais os pontos turísticos mais próximos da minha localização atual? Em quanto tempo o táxi que eu solicitei irá demorar para chegar em meu apartamento? Tais perguntas podem ser respondidas facilmente por meio de serviços baseados em localização e suas informações geradas.

Muitos destes serviços necessitam armazenar informações georeferenciadas, i.e., informações contendo coordenadas geográficas conhecidas em um dado sistema de referência, gerando assim dados de trajetórias de objetos móveis. Estes dados, quando publicados, são úteis no processo de mineração e descoberta de padrões. Mineração de dados é o processo de extrair informações não conhecidas a priori a partir de grandes conjuntos de dados. O sucesso da mineração de dados ocorre devido à disponibilidade de dados com qualidade e ao compartilhamento efetivo de informações. Neste cenário, disponibilizar dados de trajetória para fins de mineração e/ou análise, tem proporcionado descobertas bastante relevantes sobre padrões frequentes de objetos móveis, tais como: vendas referenciadas, predição de congestionamento de tráfego, entendimento do comportamento humano, planejamento de infraestrutura, migração de animais, entre outros (SILVA et al., 2014).

Entretanto, quando ocorre uma publicação de tais dados de trajetória, seja para qualquer tipo de análise, um usuário malicioso pode ser capaz de descobrir informações sensíveis sobre indivíduos que estão contidos nos dados, em virtude da existência dos semi-identificadores. Com o conhecimento dos semi-identificadores (SIs), i.e., pontos que podem ser combinados com informações externas e utilizados para reidentificar indivíduos (TERROVITIS; MAMOULIS, 2008), um adversário poderá predizer que o registro no conjunto de dados publicado pertence a um indivíduo com uma probabilidade alta de certeza. Dessa forma, caso uma publicação aconteça de maneira ingênua, tal fato pode levar a sérios riscos de violação de privacidade, uma vez que esses dados fornecem informações sobre localizações as quais indivíduos percorreram e, potencialmente, informações sensíveis, tais como costumes sociais, doenças, preferências religiosas, sexuais, etc.,

Com o objetivo de exemplificar como uma trajetória pode ser identificada quando um usuário malicioso conhece informações provenientes de outras fontes, considere o exemplo ilustrado na Figura 1. A figura mostra quatro trajetórias de indivíduos se deslocando ao longo de seis estações de metrô (a, b, c, d, e, f), coletadas a cada instante em que um usuário inseria seu cartão de acesso à estação. As diferentes cores representam as diferentes linhas de metrô

disponíveis nesse sistema. Suponha que o dono do dados liberou o conjunto de trajetórias da Figura 1 para a comunidade realizar algum tipo de mineração de dados. Agora, assuma que um adversário utilizou fontes externas, por exemplo redes sociais, que afirmavam a presença de Bob nas localizações b e e. Assim, o adversário sabe que Bob passou pelas estações b e e naquele período de tempo. Com esta informação, o adversário não pode inferir a trajetória de Bob, pois as trajetórias T_1 e T_2 contém as localizações b e e. Entretanto, caso o adversário saiba que um indivíduo frequentou as estações e e f, certamente ele poderá inferir que sua trajetória no conjunto de dados publicado refere-se a T_3. Neste caso, o par (ef) é tido como um semi-identificador de tamanho dois, pois através destes dois pontos, um adversário é capaz de reidentificar um indivíduo com 100% de certeza.

id	trajetória
T_1	(b, e, c, a)
T_2	(d, b, c, e)
T_3	(a, c, e, f)
T_4	(f, d, b, a)

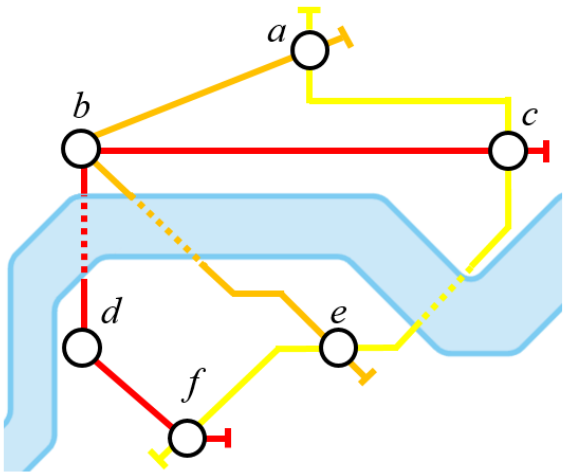


Figura 1 – Exemplo de quatro trajetórias de indivíduos ao longo de estações de metrô. As diferentes cores representam as diferentes linhas de metrô disponíveis.

Pesquisas recentes mostraram que a abordagem mais promissora para proteger a privacidade dos indivíduos é anonimizar os dados antes de publicá-los (FUNG et al., 2010; WILLISON et al., 2008). Para isso, o dono dos dados deve modificá-los de tal forma que nenhuma informação sensível sobre indivíduos possa ser descoberta a partir de uma publicação. Além disso, ele deve garantir que os dados sejam úteis para que eventuais análises possam ser efetuadas com qualidade. Este é um problema desafiador, uma vez que qualquer alteração sobre os dados distorce sua utilidade. Por esse motivo, o dono dos dados deve buscar uma solução que preserve ao máximo a utilidade das informações a qual ele deseja publicar. Uma abordagem convencional para anonimizar dados tem sido praticada com a remoção dos identificadores explícitos de indivíduos, como nome, CPF, email, etc. do conjunto de dados antes de uma publicação. Contudo, o trabalho em (SWEENEY, 2002) demonstra que, simplesmente removendo esses identificadores, não é suficiente para proteger a privacidade dos indivíduos, devido à existência dos semi-identificadores.

O objetivo da preservação de privacidade na publicação de dados (*Privacy Preserving Data Publishing* - PPDP), discutido em (FUNG et al., 2010), é fornecer métodos e ferramentas para publicação de dados de tal forma que a privacidade de indivíduos contidos naqueles dados

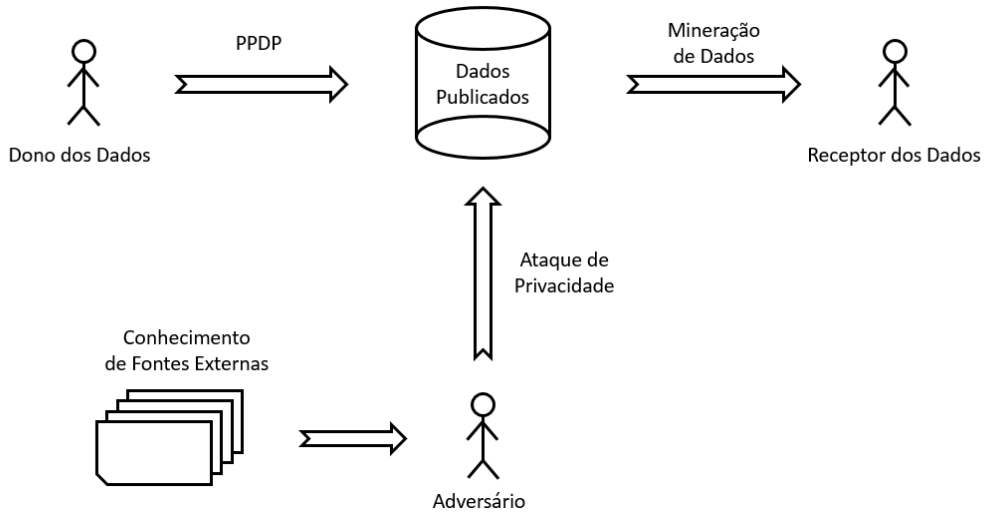


Figura 2 – Cenário de preservação de privacidade na publicação de dados.

seja protegida, enquanto a utilidade dos mesmos, após anonimização, é preservada. A Figura 2 mostra como se dá o cenário de preservação de privacidade na publicação de dados. Em geral, este cenário possui três personagens principais: o dono dos dados, o receptor dos dados e o adversário. O dono dos dados pode ser interpretado por uma organização, por exemplo, um hospital, que tem por objetivo publicar dados sem que a privacidade de indivíduos seja violada. O receptor dos dados pode ser visto também por uma organização, ou grupo de pesquisa, ou mesmo uma pessoa, que tem por objetivo utilizar os dados para mineração e/ou análise. Um adversário é uma pessoa, ou um grupo, que pretende atacar um indivíduo específico pertencente àquele dado, e assim descobrir informações sensíveis que viole sua privacidade. Este adversário utiliza conhecimento de fontes externas para realizar seu ataque.

Outro fato importante neste cenário é que o volume dos conjuntos de dados tem crescido excessivamente, e algoritmos tradicionais de preservação de privacidade, os quais rodam em ambientes de computação centralizada, tornam-se inapropriados devido ao elevado tempo de execução no processamento desses conjuntos. É comum processar tais dados com a ajuda de plataformas distribuídas, como MapReduce (DEAN; GHEMAWAT, 2004; LÄMMEL, 2008), a fim de facilitar o processamento de coleções de dados em maior escala de maneira distribuída e garantir uma melhoria considerável no desempenho desses algoritmos.

Abordagens para preservação de privacidade na publicação de dados de trajetória propostas anteriormente têm focado na qualidade da privacidade gerada (quantificação da vulnerabilidade) e na utilidade dos dados publicados. Contudo, em conjuntos de dados volumosos, tais abordagens apresentam baixo desempenho na computação dos semi-identificadores. Além disso, a qualidade dos dados anonimizados pode ser comprometida devido ao aumento da quantidade de localizações anonimizadas, uma vez que esse processo de anonimização pode não se tornar factível ao se utilizar algoritmos centralizados em grandes conjuntos de dados. Dessa forma, é fundamental uma solução que processe maiores volumes de dados, apresentando um bom desempenho e ainda assim, não comprometendo a utilidade dos dados no processo de anonimização.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Diante do cenário apresentado na motivação, o objetivo geral deste trabalho consiste em produzir uma solução distribuída para publicação de dados, que preserve a privacidade de indivíduos pertencentes a um conjunto de dados de trajetória, enquanto se mantém a utilidade dos dados.

1.2.2 Objetivos Específicos

Como forma de atender ao objetivo geral deste trabalho, estabelecemos os seguintes objetivos específicos:

- Definir uma estratégia para computar semi-identificadores em conjuntos de dados volumosos de maneira eficiente;
- Dado um modelo de privacidade, definir um método de anonimização que atenda a este modelo, minimizando a perda de informação gerada;
- Avaliar a eficiência da solução proposta utilizando dados de trajetória reais e sintéticos em termos de utilidade e tempo de execução.

1.3 CONTRIBUIÇÕES

Como resultado desta dissertação, nós implementamos um algoritmo, denominado Tranon, que tem por objetivo anonimizar conjuntos de dados visando preservar o máximo de informação original possível. Nós assumimos que a perda de informação é medida pela quantidade de localizações anonimizadas pela nossa técnica.

Em particular, as principais contribuições desse trabalho são:

- Uma estratégia distribuída para computar semi-identificadores utilizando o paradigma MapReduce;
- Uma estratégia de anonimização que adota o modelo de privacidade k^m-anonimato com o objetivo de minimizar a perda de informação.

1.3.1 Produção Científica

As contribuições científicas apresentadas neste trabalho possibilitaram a publicação do seguinte artigo:

- Felipe T. Brito, Antônio C. Araújo Neto, Camila F. Costa, André L. C. Mendonça, Javam C. Machado. A Distributed Approach for Privacy Preservation in the Publication of Trajectory Data. In: 2nd Workshop on Privacy in Geographic Information Collection and Analysis (GEOPRIVACY 2015).

1.4 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está organizada da seguinte forma: No Capítulo 2 são apresentados conceitos e definições fundamentais sobre preservação de privacidade no cenário da publicação de dados. O Capítulo 3 ressalta e discute os trabalhos relacionados mais relevantes, caracterizando modelos de privacidade anteriormente pesquisados. Em seguida, o Capítulo 4 apresenta a nossa solução proposta, utilizando uma abordagem distribuída para preservação de privacidade em dados de trajetória. Ela é constituída por (i) uma estratégia distribuída de descoberta de semi-identificadores e por (ii) uma estratégia gulosa de anonimização de dados. O Capítulo 5 apresenta os resultados obtidos por um conjunto de experimentos realizados, utilizando tanto dados de trajetória reais quanto sintéticos. Finalmente, o Capítulo 6 concui o trabalho apresentando um resumo dos resultados alcançados e mostrando direções de pesquisa futuras.

2 FUNDAMENTAÇÃO TEÓRICA

Quando um conjunto de dados é disponibilizado para fins estatísticos ou de pesquisa, técnicas de preservação de privacidade são necessárias para evitar a descoberta de informações por meio de usuários maliciosos, sobre indivíduos contidos naquele dado. Por exemplo, um hospital publica dados sobre seus pacientes para auxiliar pesquisadores da área médica a descobrirem causas de doenças, ou para estatísticos afirmarem a frequência da ocorrência de um determinado vírus. Uma vez que estes dados contém informações sensíveis sobre pacientes, tal hospital não deve liberá-los de uma maneira ingênua, devido ao alto risco de violação da privacidade dos indivíduos.

Como forma de proteger efetivamente a privacidade de indivíduos, o dono dos dados precisa garantir que eventuais descobertas não ocorram no conjunto de dados publicado. Conforme discutido no Capítulo 1, uma abordagem promissora para proteger a privacidade dos invidíduos seria anonimizá-los antes de qualquer publicação.

A anonimização de dados é uma abordagem de preservação de privacidade que modifica valores de dados com o objetivo de ocultar a identidade e/ou informações sensíveis de indivíduos. Contudo, essa modificação implica em perda de informação e, consequentemente, diminui a utilidade dos mesmos. Portanto, o desafio na publicação de dados é anonimizá-los de tal forma que a privacidade dos indivíduos é protegida, enquanto a utilidade dos dados é mantida.

Neste capítulo são apresentados conceitos e definições sobre preservação de privacidade no cenário da publicação de dados. Inicialmente, na Seção 2.1 são apresentados os diferentes tipos de dados que estão sendo publicados e que necessitam de preservação de privacidade. A Seção 2.2 traz aspectos referentes ao conhecimento de adversários, que têm por objetivo realizar ataques de ligação. Já na Seção 2.3 são abordadas as técnicas de anonimização mais utilizadas na conjuntura de publicação de dados. Na Seção 2.4 são discutidos aspectos relacionados a perda de informação e utilidade dos dados após sua publicação. Por fim, a Seção 2.5 apresenta conceitos sobre o paradigma MapReduce, utilizado na solução proposta para prover privacidade em conjuntos de dados volumosos.

2.1 TIPOS DE DADOS

2.1.1 Dados Relacionais

Um conjunto de dados relacionais D pode ser representado por uma tabela, onde cada coluna corresponde a um atributo e cada linha um registro. Em geral, esse tipo de dado tem um conjunto fixo de atributos que são comuns em uma coleção de registros $t_1, t_2, ..., t_n$. Quatro tipos de atributos podem existir em um conjunto de dados desse tipo (FUNG et al., 2010): identificadores explícitos, semi-identificadores (SI), atributos sensíveis e atributos não sensíveis.

- **Identificadores:** são atributos que identificam unicamente indivíduos, tais como "nome", "CPF", "e-mail", etc. e são sempre removidos antes de serem publicados;
- **Semi-identificadores (SI):** são todos aqueles atributos que não são identificado-

res explícitos mas podem potencialmente identificar um indivíduo, especialmente quando agrupados. São exemplos de semi-identificadores em dados relacionais "data de nascimento"e "CEP".

- **Atributos sensíveis:** contém informações sensíveis sobre indivíduos, tais como "doença", "salário", etc.
- **Atributos não sensíveis:** é qualquer tipo de atributo que não se enquadra em nenhuma das categorias anteriores.

A Tabela 1 mostra um exemplo de um conjunto de dados relacionais publicados por um Departamento Estadual de Trânsito. Identificadores explícitos foram removidos desta tabela. As colunas "Data de nascimento"e "Data da Infração"são exemplos de semi-identificadores, uma vez que podem identificar um indivíduo quando combinadas com informações externas. Já as colunas "Tipo de infração"e "Valor da multa"são exemplos de atributos sensíveis.

ID	Data de Nascimento	Data da Infração	Tipo de Infração	Valor da Multa (R$)
1	14/03/1984	05/07/2015	Média	85,13
2	27/04/1978	05/07/2015	Gravíssma	574,62
3	07/02/1971	05/07/2015	Gravíssima	574,62
4	19/04/1969	06/07/2015	Gravíssima	191,54
5	10/10/1988	06/07/2015	Grave	127,69

Tabela 1 – Exemplo de um conjunto de dados relacionais publicado por um Departamento Estadual de Trânsito

Devido a existência de semi-identificadores, adversários podem obter informações de uma vítima e serem capazes de explorar conhecimento sobre ela. Para isso, eles associam registros públicos a um indivíduo alvo, cujas informações estão contidas no conjunto de dados publicado, violando assim sua privacidade. Além disso, adversários também são capazes de inferir valores de atributos sensíveis a partir desse conhecimento.

2.1.2 Dados de Transações

Assim como um conjunto de dados relacionais, dados de transações D consistem em uma coleção de registros $t_1, t_2, ..., t_n$. Contudo, esse tipo de dados não possui uma estrutura fixa, sendo muitas vezes expresso em dados esparsos e com grande dimensionalidade. Dessa forma, cada registro t_i, denominado transação, possui um conjunto de itens caracterizados por um universo U. Por exemplo, uma transação pode ser uma consulta *web* contendo vários termos, ou uma cesta de itens comprados em um comércio, ou mesmo um conjunto de documentos contendo diferentes termos.

Um conjunto de dados de transações pode ser representado por uma tabela em que cada coluna é um atributo correspondente a um item e cada linha é uma transação. Itens dessa tabela podem ser tanto sensíveis quanto não sensíveis. Itens não sensíveis podem atuar como semi-identificadores e inferirem itens sensíveis de um indivíduo a partir da sua transação. O

número de transações que contém um conjunto de itens *i* é denominado suporte de *i*. A Tabela 2 mostra um exemplo de um conjunto de dados de transações realizadas em uma farmácia. Sabonete e xarope são exemplos de itens não sensíveis e teste de gravidez e viagra são considerados itens sensíveis. Observe que itens não sensíveis, como xarope, pode atuar como semi-identificador, uma vez que, se um adversário conhecer o indivíduo que comprou xarope na Tabela 2, ele poderá afirmar que este indivíduo possui impotência sexual, já que a única transação que contém o item "Xarope"e "Viagra"é a transação de identificador $ID = 2$.

ID	Itens Adquiridos
1	Sabonete, Shampoo, Condicionador
2	Xarope, Viagra
3	Sabonete, Shampoo
4	Condicionador, Teste de gravidez
5	Sabonete, Condicionador

Tabela 2 – Exemplo de um conjunto de dados de transações contendo itens adiquiridos em uma farmácia.

2.1.3 Dados de Grafo

Dados de grafo podem ser utilizados para representar relacionamentos entre instâncias. Nesse tipo de dado, cada nó corresponde a um registro e relações entre dois registros são representadas por ligações. Baseada em propriedades dos relacionamentos, ligações podem ser direcionadas e não direcionadas, ou ponderadas e não ponderadas (TAN et al., 2005). Um exemplo de um conjunto de dados em grafo é uma rede social onde indivíduos correspondem a nós do grafo e relações de amizades entre eles representam as ligações. Outro exemplo é um grafo que representa relações entre páginas na *web* por meio de *hyperlinks*.

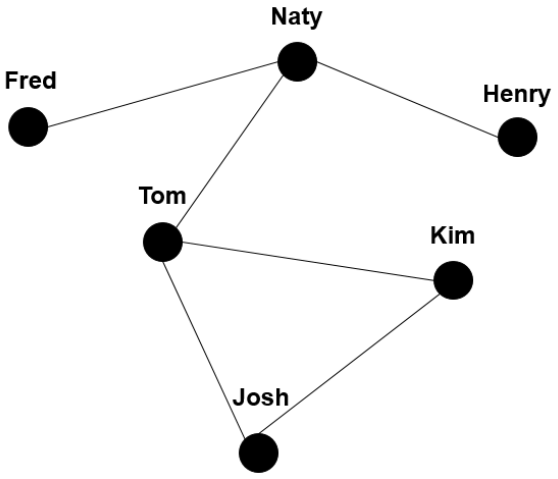


Figura 3 – Exemplo de um conjunto de dados em grafo representando relações de amizade entre indivíduos em uma rede social.

Nesse tipo de dados, qualquer conhecimento adversário sobre relações de indivíduos pode atuar como semi-identificador, seja informações de ligações direcionadas ou ponderadas. A Figura 3 mostra um exemplo de um conjunto de dados em grafo representando relações de amizade entre indivíduos em uma rede social.

2.1.4 Dados de Trajetórias

Nos últimos anos, observamos o enorme crescimento de dispositivos que utilizam sistemas baseados em localização, como o sistema de posicionamento global (*Global Positioning System* - GPS), ou sistemas *wireless*, ou mesmo sistemas de identificação por radiofrequência (*Radio Frequency IDentification* - RFID). Estes sistemas possibilitaram a captura de informações sobre objetos móveis e, consequentemente, o rastreamento de objetos com precisão, gerando assim grandes quantidades de dados de trajetória.

Uma trajetória pode ser vista como um conjunto de localizações pertencentes a um objeto móvel, coletada por dispositivos tais como GPS, celulares, *tags* RFID, etc. No contexto deste trabalho, cada localização é uma posição particular de um indivíduo, modelado pelas coordenadas X (longitude) e Y (latitude), podendo representar pontos de interesses (*Points of Interest - POIs*), como shoppings, restaurantes, parques, etc., e até mesmo incorporar uma dimensão temporal. Formalmente, dado um conjunto L de localizações, uma trajetória T representa uma ou mais localizações em L e a ordem as quais estas localizações são visitadas por um objeto móvel (por exemplo, indivíduos, ônibus, táxis), conforme Definição 1.

Definição 1 *Uma trajetória T é uma lista ordenada de localizações $(l_1, l_2, ..., l_n)$, onde $l_i \in L$, $1 \leq i \leq n$. O tamanho de uma trajetória $T = (l_1, l_2, ..., l_n)$, denotado por $|T|$, equivale ao número de localizações em T, i.e., $|T| = n$.*

A Figura 4 exemplifica um conjunto de dados de trajetória publicados por um hospital e a representação visual da trajetória T_1. Cada registro na tabela publicada contém uma trajetória T_i especificando que aquele indivíduo cruzou diversos locais de uma cidade, por exemplo. Os pontos da Figura 4 representam os seguintes locais (POIs):

- a: aeroporto;
- b: supermercado;
- c: universidade;
- d: parque;
- e: zoológico;
- f: posto de combustível;
- g: rodoviária;
- h: lanchonete;

Portanto, o indivíduo pertencente à trajetória T_1 iniciou seu percurso no aeroporto, atravessou o parque, passou pela lanchonete, e finalizou seu trajeto no zoológico. Além das localizações, o hospital, que detém os dados dos indivíduos, publicou informações referentes a

um diagnóstico médico realizado, sendo esta informação um atributo sensível associado a um indivíduo no conjunto de dados publicado.

ID	Trajetória	Diagnóstico
T_1	(a, d, h, e)	Dengue
T_2	(c, f, d, a)	Pneumonia
T_3	(a, b, f)	Pneumonia
T_4	(b, e, h, g)	Diabetes
T_5	(d, g)	AIDS
T_6	(h, c, b)	AIDS
T_7	(e, d, f, a)	Dengue

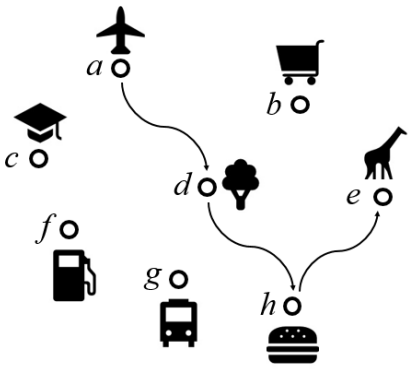


Figura 4 – Exemplo de um conjunto de dados de trajetórias publicados e a representação visual da trajetória T_1.

Observe que na Figura 4 foram removidos os identificadores explícitos de cada trajetória, como nome, RG, CPF, etc., restando apenas um identificador numérico para diferenciar as trajetórias.

Nesse tipo de dados, qualquer conjunto de localizações pode ser considerado um semi-identificador, visto que adversários podem ser capazes de reidentificar indivíduos através de informações adquiridas por fontes externas, seja de outros serviços baseados em localizações, ou de serviços baseados em redes sociais, por exemplo. Nesse contexto, semi-identificadores são representados por subtrajetórias de uma trajetória T.

2.1.4.1 Subtrajetória

Uma subtrajetória t é formada pela remoção de uma ou mais localizações de uma trajetória T, enquanto se mantém a ordem de todas as outras localizações. A Definição 2 traz o conceito formal de subtrajetória.

Definição 2 *Uma subtrajetória t de T é uma lista ordenada de localizações* $(l_u, l_{u+1}, ..., l_v)$*, onde* $\forall j$ *tal que* $u \leq j \leq v$*,* $l_j \in T$ *e* $|t| \leq |T|$*.*

Em outras palavras, t é um subconjunto de T. Por exemplo, $t = (d,e)$ é uma subtrajetória de $T_1 = (a,d,h,e)$ na Figura 4. Claramente, (d,e) pode ser obtido de T_1 pela remoção das localizações a, h.

2.1.4.2 Suporte de Subtrajetória

Definição 3 *Dado um conjunto de dados de trajetória* $D = (T_1, T_2, ..., T_n)$*, o suporte de uma subtrajetória t, denotado por* $sup(t,D)$ *é definido como o número de trajetórias distintas em D, que contém todos os elementos de t.*

Em outras palavras, o suporte de uma subtrajetória t mede o número de trajetórias que percorrem as localizações de t. Esse número representa o suporte de uma subtrajetória e está diretamente relacionado à frequência de t. Por exemplo, na Figura 4 a subtrajetória $t = (d,e)$ possui suporte 2, uma vez que os elementos de t estão contidos nas trajetórias T_1 e T_7. Além disso, se a mesma trajetória T_i possuir repetidas subtrajetórias t, considera-se $sup(t,T_i) = 1$. Assim, o número de vezes que uma subtrajetória t aparece em uma trajetória T_i não é contabilizado para o cálculo do suporte.

Outro aspecto a ser considerado no cálculo do suporte de uma subtrajetória é a propriedade reflexiva: se $t_1 = (d,e)$ e $t_2 = (e,d)$, $sup(t_1,D) = sup(t_2,D)$. Essa propriedade é decorrente da Definição 3, uma vez que o suporte de uma subtrajetória t mede o número de trajetórias distintas que cruzam os elementos de t. Dessa forma, o número de trajetórias distintas que cruzam d e e é o mesmo em e e d. Assim, a ordem das localizações em t também não é considerada para o cálculo do suporte.

Uma subtrajetória t é dita frequente se, dado um limiar de suporte mínimo k e um conjunto de trajetórias D, $sup(t,D) \geq k$ (SRIKANT; AGRAWAL, 1996). Caso contrário, t é classificada como infrequente. A frequência de uma subtrajetória pode constatá-la como sendo um semi-identificador. Dado um limiar de suporte mínimo k, se $sup(t,D) < k$, t é considerado semi-identificador. Em outras palavras, se t é infrequente em um conjunto de dados, então t é semi-identificador. Baseado no exemplo da Figura 4, seja o limiar $k = 3$, $t_1 = (a,d)$ é uma subtrajetória considerada frequente, pois possui suporte 3, enquanto $t_2 = (d,e)$ é considerada infrequente, já que $sup(t_2,D) = 2$, e consequentemente t_2 é semi-identificador.

2.2 CONHECIMENTO ADVERSÁRIO

Este trabalho considera que uma violação de privacidade ocorre por meio de um ataque, i.e., quando o adversário é capaz de associar o proprietário de um dado a um registro em um conjunto de dados publicado, utilizando um conhecimento previamente adquirido de fontes externas. Por exemplo, o adversário pode saber que a vítima mora ao lado de sua casa, assim ele pode inferir informações como endereço, CEP, gênero da vítima, etc. O adversário pode também utilizar dados de outros serviços baseados em localização, como um *checkin* em uma rede social realizado por uma vítima em uma determinada localização. O adversário pode também ter acesso a dados abertos de uma vítima, caso ela seja funcionária de órgãos públicos, por exemplo. Dessa forma, o conhecimento adversário é tido muitas vezes como imprevisível. No entanto, esse conhecimento deve ser considerado na solução de preservação de privacidade, mesmo diante da incerteza de como ele foi obtido. Por esse motivo, assumimos que um adversário conhece todos os semi-identificadores da vítima em questão. Além disso, considera-se também que um adversário sabe que o registro de uma determinada vítima pertence ao conjunto de dados publicado e procura identificá-la a partir desta divulgação.

A publicação de dados de trajetória pode levar a violação de privacidade se um adversário conhecer localizações relacionadas a um trajeto percorrido por um indivíduo. Este tipo de ataque é denominado ataque de ligação (CHEN et al., 2009). Para ilustrá-lo, considere os

dados publicados na Figura 4. Um ataque de ligação ocorre se há um trajeto específico no qual poucos indivíduos o percorreram no conjunto de dados publicado. Suponha que Camila estava na lanchonete localizada na posição h e encontrou seu amigo João. Camila também sabe que João estuda na universidade localizada em c, e que sempre depois de realizar seu lanche, ele se dirige a universidade. Uma vez que Camila tem acesso aos dados publicados na Tabela 4, ela consegue inferir que a trajetória T_6 pertence a João, já que ele é o único a percorrer os locais h e c, no conjunto de dados divulgado (Figura 4). Além disso, ela é capaz associar essa informação ao diagnóstico pertencente a T_6 e assim descobrir que João foi diagnosticado com AIDS.

2.3 TÉCNICAS DE ANONIMIZAÇÃO

Conforme mencionado anteriormente, a publicação de dados pode levar a sérios riscos de violação de privacidade devido à existência dos semi-identificadores. Isso pode acarretar em consequências graves por causa do uso não autorizado de informações sensíveis pertencentes aos indivíduos. Como forma de solucionar este problema, uma estratégia ingênua seria a não publicação dos dados, para qualquer finalidade (WONG; FU, 2010). Contudo, isso evitaria que governos, organizações, etc. pudessem tirar proveito de análises importantes de padrões e tendências para a sociedade, dificultando o possível crescimento da mesma. Outra maneira de evitar o problema da publicação de dados seria disponibilizar apenas dados estatísticos para análise, porém esta estratégia é limitada ao conhecimento estatístico que o dono dos dados possui, uma vez que ele deseja apenas publicar o dado, e não analisá-lo previamente antes de liberá-lo. A abordagem mais promissora para solucionar o problema da preservação de privacidade em uma publicação é anonimizar os dados antes de qualquer liberação (FUNG et al., 2010). Em um processo de anonimização, um conjunto de dados D é transformado em um novo conjunto D' através de um série de modificações, com o objetivo de previnir a descoberta de informações por um adversário. Existem várias técnicas de anonimização para os tipos de dados da Seção 2.1. As mais utilizadas são Generalização, Supressão e Perturbação.

2.3.1 Generalização

Nesta técnica, os valores específicos dos atributos que são considerados semi-identificadores são substituídos por valores mais genéricos, i.e., mais generalizados. O objetivo é aumentar a incerteza de um adversário em atribuir um indivíduo alvo a seu registro, ou a informações sensíveis, no conjunto de dados publicado. Para isso, um indivíduo deve pertencer a um grupo de outros registros com o mesmo valor de um atributo semi-identificador. O grupo de todos os registros contendo o mesmo valor de um atributo semi-identificador é denominado classe de equivalência (WONG; FU, 2010)

Para cada categoria de atributos, pode-se existir uma hierarquia de generalização que representa a semântica desses atributos (WONG; FU, 2010). Ao utilizar esta hierarquia, os valores de registros de uma determinada categoria são substituídos por valores menos específicos, abrangendo assim, um maior domínio de valores para aquele atributo. Por exemplo, o valor "25"(domínio numérico) pode ser substituído por "[20, 30]"(intervalo). A Figura 5 demonstra

uma hierarquia de generalização para o atributo "data de nascimento", no domínio "data".

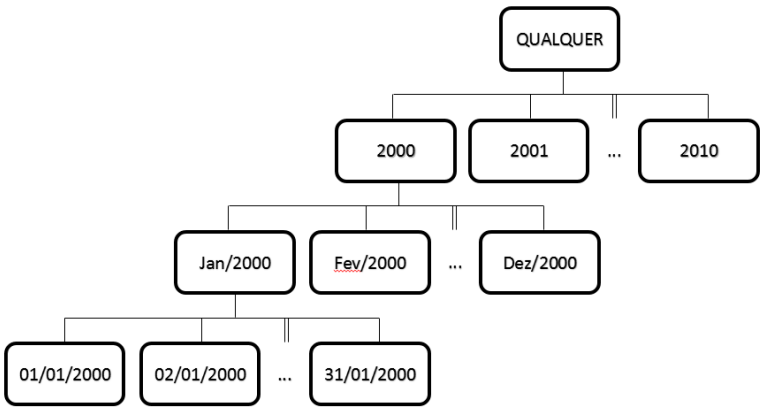


Figura 5 – Hierarquia de generalização para o atributo "Data de Nascimento"considerando dados relacionais.

Uma determinada generalização pode ser aplicada tanto para todos os seus valores quanto apenas para alguns. Dessa forma, todos os valores de registros de um atributo são mapeados para um mesmo valor generalizado (mais específico), obedecendo sua hierarquia de generalização. Este processo é denominado Generalização Global (JR.; AGRAWAL, 2005). Por outro lado, diferentes registros com o mesmo valor de atributo podem ser generalizados com diferentes valores em uma hierarquia de generalização. Este processo é chamado de Generalização Local (HE; NAUGHTON, 2009). Em resumo, a generalização global anonimiza todos os registros de um atributo da mesma forma, utilizando sempre os mesmos valores da hierarquia de generalização, enquanto a generalização local anonimiza diferentes registros de um atributo com diferentes valores seguindo sua hierarquia de generalização.

2.3.2 Supressão

Esta técnica garante que o valor de um atributo será removido ou substituído por um valor especial, como por exemplo: "*", em um conjunto de dados D. Os tipos mais comuns de aplicação dessa técnica são:

- Supressão de registro: um registro é removido inteiramente do conjunto de dados;
- Supressão de valor: refere-se a substituição de todas as instâncias de um valor de um atributo por "*";
- Supressão de célula: apenas algumas instâncias de valores de um atributo são removidas do conjunto de dados.

De maneira semelhante a técnica de Generalização, a Supressão pode ser aplicada de maneira global ou local. A supressão global refere-se a remoção de todas as instâncias de um valor de atributo, garantindo que aqueles valores não serão descobertos em um conjunto de dados publicado, uma vez que todos foram removidos. A supressão local é caracterizada pela remoção de apenas algumas instâncias de um valor de atributo, contudo deve-se garantir que os valores restantes não possam ser descobertos utilizando alguma outra técnica. No contexto de

dados de trajetória, uma supressão global é especificada pela remoção de uma localização l_i por inteira em um conjunto de dados D. Já a supressão local é evidenciada pela remoção de uma localização l_i apenas de algumas trajetórias. Neste trabalho é utilizada uma técnica de supressão global para anonimizar um conjunto de dados de trajetória.

2.3.3 Perturbação

Esta abordagem tem sido comumente utilizada em controle de descoberta estatística (IYENGAR, 2002), devido à sua simplicidade, eficiência e capacidade de preservar informações estatísticas. A ideia geral desta técnica é substituir os valores dos atributos semi-identificadores originais por valores sintéticos, de modo que informações estatísticas calculadas a partir dos dados originais não se diferenciam significativamente de informações estatísticas calculadas sobre os dados perturbados. Dependendo do grau de perturbação, registros podem ou não corresponder a registros reais.

Ao contrário das técnicas de generalização e supressão, que preservam a veracidade dos dados, a perturbação resulta em um conjunto de dados com valores sintéticos. Muitas vezes isso acarreta em informações sem sentido para aqueles que irão utilizá-las. Em contrapartida, a generalização e a supressão tornam os dados menos precisos, mas semanticamente coerentes com os dados originais.

2.4 PERDA DE INFORMAÇÃO

A anonimização de dados causa perda de informação e muitas vezes compromete a utilidade dos dados. Por esse motivo, a utilidade da informação publicada é inversamente proporcional ao grau de anonimização a qual ela é submetida. Quanto mais anonimizados forem os dados, menos úteis serão para o usuário final.

Como forma de preservar a utilidade dos dados publicados, deve-se assegurar que o mínimo de distorção deva ser gerada na anonimização. Esta distorção causada por um processo de anonimização é denominada perda de informação. Há algumas métricas para se medir perda de informação. Tais métricas podem ser utilizadas tanto para medir a utilidade do conjunto de dados publicado em relação aos dados originais, ou então serem utilizadas como uma métrica de busca, com o objetivo de guiar os passos em busca da melhor solução de anonimização no espaço de todas as possibilidades (FUNG et al., 2010).

Neste trabalho, assume-se como perda de informação a quantidade de localizações suprimidas de um conjunto de dados de trajetória. Uma vez que é aplicada a técnica de anonimização envolvendo supressão global, quanto menor for a quantidade de pontos de trajetória removidos globalmente, maior será a utilidade das trajetórias publicadas.

2.5 PARADIGMA MAPREDUCE

Armazenar, processar e gerenciar grandes volumes de dados não tem sido uma tarefa trivial, pois muitas vezes essas operações se tornam inviáveis de execução devido ao modelo

de computação tradicional, que utiliza tecnologias baseadas em banco de dados relacionais, e processamento em máquinas com baixa escalabilidade. Além disso, desenvolver soluções para ambientes distribuídos envolve uma série de obstáculos que devem ser considerados pelos programadores, como concorrência, tolerância a falhas, balanceamento de carga, entre outros. Por esse motivo, cada vez mais faz-se necessário explorar paradigmas de programação e processamento distribuído.

Diversas abordagens têm sido propostas como forma de solucionar essa dificuldade, como por exemplo o desenvolvimento de aplicações no ambiente de Computação nas Nuvens, ou mesmo sistemas baseados em *Distributed Hash Table* (DHT), e ainda *arrays* multidimensionais (SOUSA et al., 2010). Nesse âmbito, surge então o paradigma *MapReduce*, um modelo de programação paralela para processamento distribuído de grandes volumes de dados, proposto inicialmente pela empresa *Google*. Esta abordagem tem se destacado devido a sua facilidade de utilização, uma vez que a tarefa do programador consiste em implementar as duas funções principais pertencentes ao framework, *Map* e *Reduce*, indicando como o mapeamento e a redução dos dados serão realizados. Além disso, todo o trabalho de distribuição do sistema, incluindo problemas de comunicação, concorrência, tolerância a falhas, etc., é conduzido pelo próprio framework.

Em termos gerais, o framework *MapReduce* divide uma determinada tarefa em várias tarefas menores, que são executadas em paralelo em máquinas distintas (nós), e então combinadas para a realização da tarefa inicial por inteira. Na etapa de mapeamento é gerado um conjunto intermediário de pares <chave, valor> de cada entrada de dados. O framework os ordena pelas chaves e agrupa todos os valores de mesma chave em um outro par, criando um conjunto de pares intermediário. Esta etapa é denominada *shuffle*. Pares com a mesma chave são conduzidos para a etapa de redução pelo próprio framework. Nesta etapa, cada par intermediário irá processar todos os valores de uma mesma chave e gerará um novo conjunto final de pares <chave, valor>. Esse conjunto, contendo todas as saídas da etapa de redução, é utilizado para compor a solução final.

Para esclarecer o funcionamento do paradigma *MapReduce*, considere uma coleção de transações cujo objetivo é descobrir o núumero de ocorrências de cada item existente nessa coleção. Suponha que itens foram adquiridos em uma padaria conforme a entrada da Figura 6. Cada transação é representada por um ou mais itens. O objetivo é separar os itens e suas quantidades. Caso essa contagem fosse processada de forma centralizada, apenas uma máquina realizaria a tarefa completa considerando todas as transações. Contudo, no exemplo da Figura 6, este processo é dividido em quatro nós distintos, cabendo a cada um deles a tarefa de contar apenas uma transação. Esse método corresponde a fase de Map. Para cada item encontrado na transação, a função emite um par <chave, valor>, onde a chave é o item em si e o valor é a constante 1. Por sua vez, a função Reduce recebe como chave um item e como valor um iterador para todos os valores emitidos pela função Map, associados com o item em questão. Dessa forma, todos os valores são somados e um par <chave, valor> contendo o item e o seu total de ocorrências é emitido.

Diversas iniciativas de implementações do paradigma *MapReduce* foram propostas

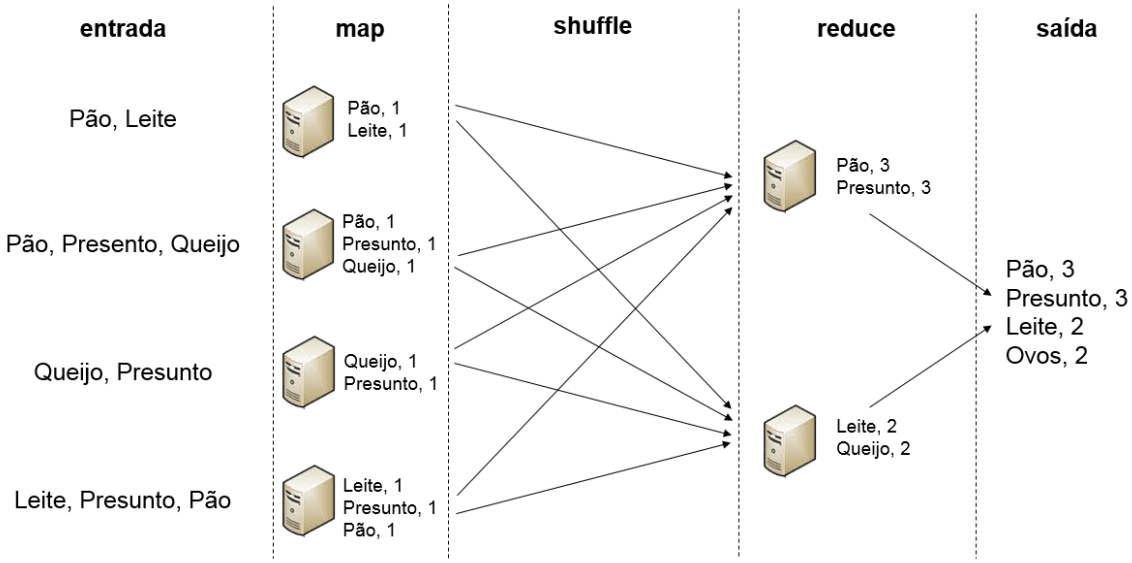


Figura 6 – Exemplo de contagem de itens utilizando MapReduce.

em diferentes linguagens de programação. Dentre elas, destaca-se o framework Hadoop (WHITE, 2009), desenvolvido pela *Apache Software Foundation*, que provê processamento distribuído de grandes quantidades de dados em nós computacionais. Neste framework, problemas como integridade dos dados, disponibilidade dos nós computacionais, escalabilidade da aplicação e recuperação de falhas ocorrem de forma transparente ao usuario.

O *Hadoop* é um framework de código aberto, implementado em Java e que funciona em uma arquitetura mestre-escravo. O nó mestre é responsavel pela atribuição das tarefas de mapeamento e redução aos nós escravos. Além de escalonar tarefas aos nós escravos, o nó mestre também é responsável por gerenciar as execuções solicitadas pelo usuário, sendo responsável pelo controle de falhas, reiniciando as tarefas que não puderam ser completadas pelos nós escravos. O *Hadoop* também possui um sistema de arquivos distribuído nativo denominado HDFS (Hadoop File System). Este sistema de arquivos permite o armazenamento e transmissão de grandes volumes de dados em nós de baixo custo, possuindo mecanismos que o caracterizam como um sistema extremamente tolerante a falhas.

2.6 CONCLUSÃO

Neste capítulo apresentamos conceitos e definições sobre preservação de privacidade no cenário da publicação de dados. Foram apontados diversos tipos de dados, os quais estão sendo publicados, e que necessitam de preservação de privacidade. Um maior enfoque foi dado aos dados referentes a trajetórias, por ser o objeto maior deste trabalho. Foram apresentados aspectos referentes ao conhecimento adversário e as técnicas de anonimização mais utilizadas para previnir esse tipo de ataque em uma publicação. Um balanço entre o quesito perda de informação e utilidade dos dados foi traçado e por fim foram exibidos fundamentos do paradigma *MapReduce*, utilizado na nossa solução para prover privacidade em conjuntos de dados volumosos de forma distribuída.

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados alguns trabalhos relacionados à preservação de privacidade no cenário da publicação de dados relacionais e dados de trajetória. Conforme discutido no Capítulo 2, um adversário sabe que o registro de uma possível vítima está contido no conjunto de dados publicado. Assim, ele pode descobrir tanto a identidade da vítima a qual o registro pertence, quanto atributos sensíveis relacionados aquele registro. Dessa forma, os trabalhos relacionados são classificados por tipo de descoberta de informação, considerando Descoberta de Identidade e Descoberta de Atributo como abordagens para preservação de privacidade. Ao final do capítulo, realizamos uma breve discussão destacando as limitações dos trabalhos apresentados e uma comparação com a nossa contribuição.

3.1 DESCOBERTA DE IDENTIDADE

Em um ataque de descoberta de identidade o objetivo de um adversário é reidentificar o registro de um indivíduo específico ou de um indivíduo qualquer, cujas informações aparecem no conjunto de dados publicado. Mais precisamente, o adversário tem como premissa que o registro de um determinado indivíduo foi divulgado e, com base no seu conhecimento de fontes externas, ele é capaz de inferir que aquele registro pertence à vítima a qual ele está interessado, violando assim sua privacidade.

Como forma de previnir a descoberta de identidade uma prática bastante utilizada é a remoção de identificadores explícitos, tais como nome, CPF, ID, e-mail, endereço, etc., antes de qualquer publicação. Entretanto, (SWEENEY, 2002) demonstrou que simplesmente remover tais identificadores explícitos não é o suficiente para proteger a privacidade dos indivíduos contra a descoberta de identidade. A seguir, são apontados trabalhos que tem por finalidade previnir esse tipo de descoberta, bem como métodos para proteção de dados contra ataques de ligação.

3.1.1 k-anonimato

Vários avanços em modelos de privacidade foram desenvolvidos para solucionar o problema da anonimização de dados. O mais conhecido deles é o k-anonimato, proposto em (SWEENEY, 2002). A princípio esse modelo de privacidade era focado apenas em esquemas de banco de dados relacionais, como forma de evitar a descoberta de identidade, e assegurando que, para cada combinação de valores de semi-identificadores, existem pelo menos k registros no conjunto de dados publicado. Isto é, cada registro em um conjunto de dados k-anônimo é indistinguível de, pelo menos, $k-1$ outros registros em relação ao conjunto de semi-identificadores. Dessa forma, cada registro não pode ser ligado a um indivíduo por um atacante com probabilidade maior que $1/k$.

Considere a Tabela 3 publicada por um determinado hospital. Ele mostra um exemplo de um conjunto de dados 3-anônimo, onde CEP e ano de nascimento são considerados semi-identificadores. Note que cada registro é indistinguível de outros 2 em relação ao conjunto de semi-identificadores. Em outras palavras, cada classe de equivalência contém pelo menos 3

registros. Por exemplo, o indivíduo de ID = 1 é indistinguível dos registros de ID = 2 e ID = 3, uma vez que seus semi-identificadores possuem os mesmos valores.

ID	CEP	Ano de Nascimento	Doença
1	6082****	1984	Dengue
2	6082****	1984	Pneumonia
3	6082****	1984	AIDS
4	6082****	1969	Dengue
5	6082****	1969	Câncer
6	6082****	1969	Câncer
7	6011****	1977	AIDS
8	6011****	1977	Hepatite
9	6011****	1977	Diabetes
10	6011****	1977	Pneumonia

Tabela 3 – Dados 3-anônimo

Para um conjunto de dados atender ao modelo de privacidade *k*-anonimato, o primeiro passo é reconhecer seu conjunto de semi-identificadores. Conforme (SWEENEY, 2002), uma forma simples de fazê-lo é considerar semi-identificadores como sendo todos os atributos que podem existir em fontes externas e assim serem utilizados por adversários como forma de ataque de ligação. Em seguida deve-se empregar alguma técnica de anonimização para ofuscar esses semi-identificadores. As técnicas mais utilizadas para atender ao modelo original (SAMARATI, 2001) e muitas de suas subsequentes versões melhoradas (JR.; AGRAWAL, 2005), (IYENGAR, 2002), são: generalização, o qual substitui o valor de semi-identificadores por valores mais gerais; e supressão, que remove o semi-identificador por inteiro.

K-anonimato também tem sido proposto recentemente com o objetivo de oferecer privacidade não só na publicação de dados relacionais, mas também na publicação de dados de trajetória (BONCHI et al., 2011). Vale ressaltar que o problema de encontrar uma *k*-anonimização ótima é NP-difícil, conforme demonstrado por (MEYERSON; WILLIAMS, 2004). Neste contexto, generalização e supressão também são os métodos de anonimização mais utilizados para garantir a privacidade de dados de trajetória utilizando k-anonimato (ABUL et al., 2008; YAROVOY et al., 2009). Os trabalhos propostos por esses autores são descritos a seguir.

3.1.2 NWA - Never Walk Alone

O trabalho proposto em (ABUL et al., 2008), denominado NWA (*Never Walk Alone*), emprega um modelo de privacidade conhecido como (k, δ)-anonimato. A ideia geral desse método é generalizar trajetórias em agrupamentos cilíndricos, ou seja, trajetórias que possuem propriedades em comum são agrupadas em cilindros de raio δ, os quais contêm pelo menos k trajetórias. Cada trajetória que pertence a um grupo anonimizado (cilindro), gerado pelo NWA, é protegida da descoberta de identidade devido à existência de outras trajetórias que aparecem no mesmo grupo. Para produzir tais cilindros, o algoritmo em (ABUL et al., 2008) faz com que

trajetórias sejam divididas em grupos disjuntos, de forma que todas as trajetórias de um grupo tenham aproximadamente o mesmo tempo de início e fim, ou seja, trajetórias que iniciaram e finalizaram por volta do mesmo instante. Consequentemente, as trajetórias de cada grupo são agrupadas utilizando distância euclidiana. A distância está relacionada ao parâmetro δ e às coordenadas geográficas de cada trajetória. Cada cilindro gerado deve conter pelo menos k trajetórias com raio no máximo δ. Esta abordagem utilizando incerteza é ilustrada na Figura 7. A trajetória de cor azul representa a trajetória publicada com raio δ, enquanto a trajetória em vermelho pode ser considerada como um possível deslocamento de uma das trajetórias originais dentro do cilindro. Esse método tem como objetivo minimizar o volume dos agrupamentos cilíndricos gerados com base na área de incerteza *delta*. A estratégia NWA assume que todos os usuários possuem semi-identificadores comuns.

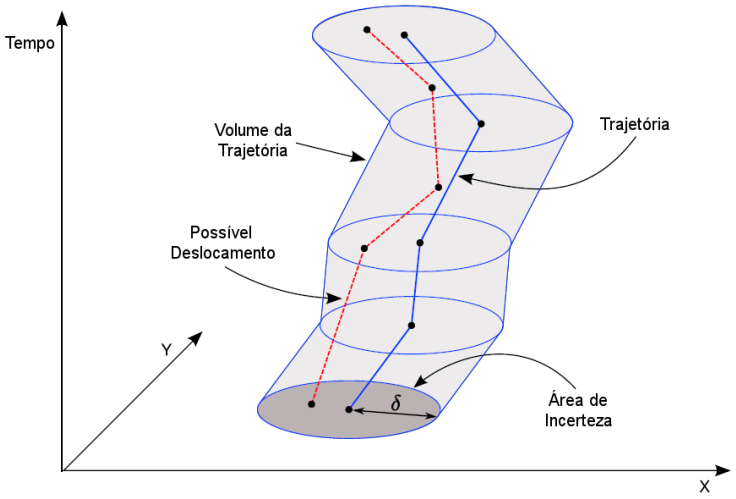


Figura 7 – Idéia da estratégia NWA.

Uma limitação desse método pode ser observada no primeiro passo do algoritmo em (ABUL et al., 2008), no qual a partição de trajetórias em grupos distintos pode resultar em pequenos grupos com menos de k trajetórias (TRUJILLO-RASUA; DOMINGO-FERRER, 2013), não oferecendo garantias de privacidade significativas na prática. Além disso, o método de anonimização utilizado altera a localização atual de um objeto móvel, o que não preserva a veracidade dos dados quando estes são publicados.

3.1.3 Estratégia proposta em (YAROVOY et al., 2009)

O trabalho propõe uma abordagem baseada no modelo de privacidade k-anonimato utilizando generalização para criar grupos de dados anônimos. Os autores consideraram o tempo como um semi-identificador e suportam privacidade personalizada. Diferentemente da abordagem proposta por (ABUL et al., 2008), a qual assume que todos os usuários compartilham semi-identificadores comuns, (YAROVOY et al., 2009) assume que cada usuário tem um conjunto distinto de QIDs e de tempos que necessitam de proteção, permitindo que a privacidade de cada trajetória seja preservada de maneira diferente. Consequentemente, cada trajetória tem seu

próprio conjunto de semi-identificadores. Devido ao fato de diferentes objetos móveis possuírem QIDs distintos, grupos anônimos associados a diferentes objetos móveis podem não ser disjuntos, conforme exemplo a seguir.

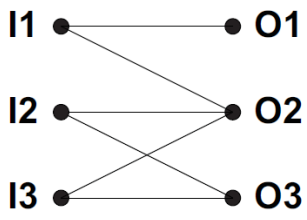
Considere as trajetórias da Figura 8(a). Suponha $k = 2$ e $QID(O_1) = \{t_1\}$, $QID(O_2) = QID(O_3) = \{t_2\}$. Intuitivamente, o melhor grupo anônimo para O_1, uma vez que $QID(O_1) = \{t_1\}$ é O_1, O_2. Isto significa que no conjunto de dados foi atribuída a região $[(1,2),(2,3)]$ para O_1 e O_2 no tempo t_1. Já o melhor grupo anônimo para O_2 e O_3, uma vez que $QID(O_2) = QID(O_3) = \{t_2\}$ é O_2, O_3. Assim, no conjunto de dados foi atribuída a região $[(2,6),(3,7)]$ para O_2 e O_3 no tempo t_2. Tal anonimização é ilustrada na Figura 8(b). Claramente, os grupos anônimos de O_1 e O_2 se sobrepõem, i.e., não são disjuntos.

MOB	t_1	t_2
O_1	$(1,2)$	$(5,3)$
O_2	$(2,3)$	$(2,7)$
O_3	$(6,6)$	$(3,6)$

(a)

MOB	t_1	t_2
O_1	$[(1,2),(2,3)]$	$(5,3)$
O_2	$[(1,2),(2,3)]$	$[(2,6),(3,7)]$
O_3	$(6,6)$	$[(2,6),(3,7)]$

(b)



(c)

Figura 8 – (a) Dados originais (b) Dados distorcidos (c) Grafo de ataque.

Os autores apresentam a noção de k-anonimato para dados de trajetória, definindo um grafo de ataque para adversários conforme Figura 8(c). Um grafo de ataque associado a um conjunto de dados D e sua versão distorcida D^* é um grafo bipartido G consistindo de nós para cada indivíduo I em D e nós para cada objeto móvel O no conjunto de dados D^*. G contém uma aresta (I,O) se $D(O,t) \sqsubseteq D^*(O,t), \forall t \in QID(I)$. Pelo grafo, o objeto móvel O1 é representado apenas pelo suposto indivíduo I1, possibilitando a associação do indivíduo 1 ao obejto móvel O1. Um objeto móvel satisfaz o modelo k-anonimato se o grafo de ataque é simétrico e se cada nó no grafo possui grau maior ou igual a k. A anonimização é obtida pela identificação das classes de equivalência e pela generalização de todos os registros de cada classe. Dessa forma, grupos de anonimização passam a não ser mais disjuntos. Os autores assumem que o dono dos dados conhece os semi-identificadores para cada trajetória, entretanto eles não mencionam como esse

conhecimento pode ser obtido.

3.1.4 seqAnon

Em (POULIS et al., 2013a) os autores propõem um algoritmo, denominado *seqAnon*, baseado no modelo de privacidade k^m-anonimato para dados de trajetória. Diferentemente dos métodos em (ABUL et al., 2008) e (YAROVOY et al., 2009), essa abordagem assume que um adversário pode conhecer até m localizações de qualquer trajetória. Em outras palavras, ele considera como conhecimento adversário no máximo m localizações de um indivíduo, não podendo associá-lo a menos de k trajetórias no conjunto de dados publicado.

Os autores utilizam generalização baseada em distância como técnica de anonimização. Eles argumentam que taxonomias de localizações podem não refletir adequadamente na distância entre as mesmas, e trajetórias generalizadas com base nessas taxonomias podem levar a perdas bastante significativas de informação. Taxonomias nesse contexto representam a semântica das trajetórias. Em vez disso, a generalização proposta é definida como conjuntos de pelo menos duas localizações. Se uma trajetória T_i contém uma localização generalizada $L = \{l_1, l_2, ..., l_n\}$, isso significa que T_i contém exatamente uma localização composta por $l_1, l_2, ..., l_n$, desconsiderando a semântica das localizações no conjunto de dados publicado. Por exemplo, considere uma trajetória publicada $T_i = (a, bc, def, g)$, tal que (a, b, c, d, e, f, g) representam localizações originais ainda no conjunto de dados original. Observe que a generalização proposta em T_i une duas ou mais localizações em uma mesma localização, ignorando a semântica de trajetória T_i.

É importante ressaltar que esse algoritmo é implementado com base no princípio *apriori* (TAN et al., 2005) no que tange algoritmos de mineração de dados. Por esse motivo, ele visa proteger um número maior de indivíduos contra descoberta de identidade, sendo capaz de processar um volume de dados maior.

3.2 DESCOBERTA DE ATRIBUTO

Todos os métodos discutidos na seção anterior visam proteger dados contra descoberta de identidade, garantindo que cada indivíduo é indistinguível de um grupo de outros indivíduos no conjunto de dados publicado. Contudo, em alguns casos, um adversário pode ser capaz de inferir atributos sensíveis de uma vítima mesmo sem reidentificar seu registro, por exemplo, quando a maioria dos registros dentro de uma mesma classe de equivalência possuem o mesmo valor para um atributo sensível. Esse tipo de descoberta é denominada Descoberta de Atributo.

Com o objetivo de evitar este tipo de descoberta, vários modelos de privacidade foram propostos visando aumentar a incerteza de um adversário em descobrir informações sensíveis a partir de dados publicados. Tais trabalhos são apresentados a seguir.

3.2.1 l-diversidade

O modelo de privacidade l-diversidade proposto em (MACHANAVAJJHALA et al., 2007) adverte que um adversário pode inferir informações sensíveis sobre registros mesmo sem identificá-los. Para evitar esse tipo de descoberta, Machanavajjhala et al. propõe o princípio l-diversidade para dados relacionais, com o intuito de contornar as limitações existentes no k-anonimato.

Um conjunto de dados é dito l-diverso se cada classe de equivalência possui l "bem representados"valores para seus atributos sensíveis. Por exemplo, a Tabela 3 mostra dados 2-diverso, pois cada classe de equivalência possui pelo menos 2 valores distintos de atributos sensíveis em seus registros. Formalmente, este modelo requer que o número de valores distintos de atributos sensíveis em cada classe de equivalência seja no mínimo l.

3.2.2 LKC e $(K,C)_L$-privacidade

Outro método proposto a fim de evitar a descoberta de atributo é apresentado por (MOHAMMED et al., 2009), o qual emprega o modelo LKC-privacidade para dados de trajetória. Este modelo garante que para cada sequência de localizações com um tamanho máximo L, existem pelo menos K registros em um grupo, e a porcentagem de valores sensíveis em cada grupo não é maior que C. Formalmente, um conjunto de dados T satisfaz o modelo LKC-privacidade se para qualquer semi-identificador q, dado $|q| \leq L$:

- $|T[q]| \geq K$, onde $|T[q]|$ representa o número de registros no conjunto de dados T que contém q, e $K > 0$ é o limiar de anonimização;
- $P(s|q) \leq C$ para cada $s \in S$, onde S é o conjunto de valores sensíveis, C é o limiar de confiança e $P(s|q)$ é a probabilidade de se inferir algum valor sensível s a partir de q.

Por exemplo, um hospital publica um conjunto de dados de trajetória de pacientes com seus respectivos diagnósticos de saúde, conforme Tabela 4. Considere que cada registro contém um atributo sensível (diagnóstico), e que uma trajetória é representada por uma sequência de localizações $L = (a,b,c,d,e,f,g,h)$. Se um valor sensível aparece frequentemente em alguma sequência de localizações, informações sensíveis podem ser inferidas a partir de tal sequência. Suponha que um adversário sabe que João visitou as localizações b e f. Uma vez que dois registros, de três (IDs $1,7,8$), no conjunto de dados publicado, contém as localizações b e f e possuem valor sensível AIDS, o adversário pode inferir que João foi diagnosticado com AIDS com $2/3 = 67\%$ de certeza. Por outro lado, tal inferência não seria possível com valor acima de 50% caso o conjunto de dados da Tabela 5 fosse publicado, já que foi anonimizado utilizando o modelo de privacidade LKC-privacidade com o valores $L = 2$, $K = 2$ e $C = 50\%$.

Para garantir que um conjunto de dados de trajetória atenda a esse modelo, os autores identificam sequências de violação e aplicam supressão global nos pares selecionados de violações.

ID	Trajetória	Diagnóstico	...
1	$b \rightarrow d \rightarrow c \rightarrow f \rightarrow h$	AIDS	...
2	$f \rightarrow h \rightarrow e$	Dengue	...
3	$d \rightarrow c \rightarrow f \rightarrow e$	Pneumonia	...
4	$b \rightarrow g \rightarrow h \rightarrow e$	Dengue	...
5	$d \rightarrow h \rightarrow e$	Generalização	...
6	$g \rightarrow f \rightarrow e$	Diabetes	...
7	$b \rightarrow f \rightarrow h \rightarrow e$	Diabetes	...
8	$b \rightarrow g \rightarrow f \rightarrow h$	AIDS	...

Tabela 4 – Dados publicados de trajetórias de pacientes e seus diagnósticos de saúde

ID	Trajetória	Diagnóstico	...
1	$d \rightarrow f \rightarrow h$	AIDS	...
2	$f \rightarrow h \rightarrow e$	Dengue	...
3	$d \rightarrow f \rightarrow e$	Pneumonia	...
4	$g \rightarrow h \rightarrow e$	Dengue	...
5	$d \rightarrow h \rightarrow e$	Generalização	...
6	$g \rightarrow f \rightarrow e$	Diabetes	...
7	$f \rightarrow h \rightarrow e$	Diabetes	...
8	$g \rightarrow f \rightarrow h$	AIDS	...

Tabela 5 – Versão anonimizada dos dados com $L = 2$, $K = 2$, $C = 50\%$

Assumindo que cada registro em um conjunto de dados é composto por trajetórias de usuários e seus atributos sensíveis, os autores em (CHEN et al., 2013) propõe um modelo muito semelhante ao apresentado em (MOHAMMED et al., 2009), que adota o modelo $(K,C)_L$-privacidade. Eles introduzem um algoritmo de anonimização o qual emprega supressão local e supressão global a fim de melhorar a utilidade dos dados, se comparado a (MOHAMMED et al., 2009), no momento da publicação. Esta abordagem permite a adoção de várias métricas de utilidade dos dados para diferentes cenários.

Ambos os modelos sugeridos permitem proteger a privacidade de indivíduos evitando tanto a descoberta de identidade quanto a descoberta de atributo. Ou seja, os dois modelos de privacidade que utilizam os parâmetros L, K e C, e são aplicáveis para anonimização de dados de trajetória com ou sem atributos sensíveis.

3.2.3 Select-Organize-Anonymize Framework

Recentemente, (POULIS et al., 2013b) apresentaram um novo *framework* de anonimização de trajetórias introduzindo o modelo de privacidade $(k, l)^m$-anonimato, oferecendo privacidade a indivíduos também contra a descoberta de identidade e de atributo. Os autores utilizam três fases para consolidar seu *framework*. São elas: seleção, organização e anonimização.

- **Seleção:** identifica trajetórias similares baseada nas informações de cada uma delas;

- **Organização:** ordena as trajetórias selecionadas com base na similaridade e as agrupa em *clusters*;
- **Anonimização:** constrói *clusters* anonimizados a partir da fase anterior aplicando generalização.

O método de anonimização utilizado é o mesmo proposto em (POULIS et al., 2013a), adotando generalização baseada em distância para atender ao modelo $(k, l)^m$-anonimato. Os autores também implementam seus algoritmos com base no princípio *apriori* e consideram atributos sensíveis no contexto do trabalho.

A propriedade $(k, l)^m$-anonimato garante que um adversário que possui conhecimento prévio de uma subtrajetória t de m localizações não sensíveis, nem pode violar a privacidade de um indivíduo a menos de k trajetórias no conjunto de dados publicado, nem pode associá-lo a uma localização sensível com uma probabilidade maior que $1/l$.

3.3 DISCUSSÃO

Os trabalhos apresentados neste capítulo utilizam generalização ou supressão como técnicas de anonimização antes da publicação de dados. Algumas abordagens visam previnir a descoberta de identidade, outras visam previnir a descoberta de atributo, e outras previnem ambas. Dentre os trabalhos relacionados apontados, seis técnicas foram propostas para publicação de dados de trajetória, sendo apresentados diferentes modelos de privacidade para cada uma das estratégias.

A nossa contribuição adota o mesmo modelo de privacidade proposto em (POULIS et al., 2013a), denominado k^m-anonimato, para dados de trajetória. A escolha do k^m-anonimato deve-se ao fato de usuários maliciosos possuírem um conhecimento limitado sobre indivíduos, na prática. Dessa forma, adversários possuem um conhecimento de no máximo m localizações a qual um determinado indivíduo percorreu. A partir deste modelo, propomos um método que previne ataques de ligação e que levam à descoberta de identidade, sendo esta descoberta a mais comum em conjuntos de dados de trajetórias publicados. Ainda com o objetivo de atender ao modelo proposto, utilizamos supressão como técnica de anonimização.

Dentre as desvantagens das estratégias apresentadas nas Seções 3.1 e 3.2 está o fato de todas serem elaboradas para executarem algoritmos em ambientes de computação centralizada. Por esse motivo, os trabalhos relacionados estudados não apresentam bom desempenho quando se deseja preservar a privacidade de dados em maior escala. Além disso, lidar com técnicas de anonimização requer que alguma função de perda de informação seja minimizada. Nesta dissertação, adotamos como perda de informação a quantidade de localizações removidas do conjunto de dados após a etapa de anonimização. Para isso, propomos uma abordagem distribuída que preserva a privacidade de indivíduos na publicação de dados de trajetória, utilizando o paradigma MapReduce e um método de seleção de localizações, que minimiza a perda de informação a qual os dados estão submetidos no processo de anonimização.

A Tabela 6 traz um resumo comparativo entre as abordagens propostas e a nossa contribuição (BRITO et al, 2015).

Trabalho	Modelo de Privacidade	Técnica(s) de Anonimização	Previne Descoberta	Tipo de Dado	Abordagem
Sweeney et al.	k-anonimato	Generalização	Identidade	Relacional	Centralizada
Abul et al.	(k,δ)-anonimato	Generalização	Identidade	Trajetória	Centralizada
Yarovoy et al.	k-anonimato	Generalização	Identidade	Trajetória	Centralizada
Poulis et al.	k^m-anonimato	Generalização	Identidade	Trajetória	Centralizada
Machanavajjhala et al.	l-diversity	Generalização	Atributo	Relacional	Centralizada
Mohammed et al.	LKC-privacidade	Supressão	Identidade, Atributo	Trajetória	Centralizada
Chen et al.	$(K,C)_L$-privacidade	Supressão	Identidade, Atributo	Trajetória	Centralizada
Poulis et al.	$(k, l)^m$-anonimato	Generalização	Identidade, Atributo	Trajetória	Centralizada
Brito et al.	k^m-anonimato	Supressão	Identidade	Trajetória	Distribuída

Tabela 6 – Análise Comparativa entre os Trabalhos Relacionados

3.4 CONCLUSÃO

Este capítulo apresentou os principais trabalhos relacionados com o tema desta dissertação. Embora existam diversas abordagens que visam proteger a privacidade de indivíduos na publicação de dados de trajetória, tais trabalhos foram elaborados para executarem algoritmos em ambientes de computação centralizada, não apresentando um bom desempenho quando executados sobre conjuntos de dados em maior escala. A contribuição apresentada nesta dissertação é comparada com o trabalho relacionado *seqAnon*, uma vez que o modelo de privacidade k^m-anonimato é utilizado por ambas as abordagens, e visam proteger a privacidade de indivíduos contra usuários maliciosos que conhecem até m localizações.

4 ABORDAGEM DISTRIBUÍDA PARA PRESERVAÇÃO DE PRIVACIDADE NA PUBLICAÇÃO DE DADOS DE TRAJETÓRIA

4.1 TRANON: VISÃO GERAL

Com o intuito de solucionar o problema de preservação de privacidade em conjuntos de dados de trajetórias públicos, este capítulo apresenta Tranon (**Tra**jectory **Anon**ymity), uma estratégia de publicação que mantém a privacidade de indivíduos pertencentes a um conjunto de dados de trajetória, garantindo a utilidade dos mesmos. Para isso, propomos uma abordagem distribuída para calcular semi-identificadores e uma técnica de supressão para anonimizar dados de trajetória e assim preservar a identidade de indivíduos contra ataque de ligação.

O Algoritmo Tranon tem como objetivo atacar um conjunto de dados por meio da descoberta de seus semi-identificadores, e, com base neste conhecimento, aplicar uma técnica de anonimização para previnir que usuários maliciosos descubram a identidade de indivíduos. Após a anonimização dos dados, caso um novo ataque de descoberta de semi-identificadores seja efetuado com os mesmos parâmetros utilizados anteriormente, nenhum valor é retornado ao usuário. Ou seja, um adversário não será mais capaz de descobrir os semi-identificadores da vítima e, consequentemente, violar sua privacidade. Nosso algoritmo é composto por duas grandes etapas:

- **Descoberta dos SI:** calcula o conjunto de semi-identificadores Q utilizando o Paradigma MapReduce, como forma de ataque a um conjunto de dados D;
- **Anonimização:** transforma um conjunto de dados D em um conjunto D' k^m-anônimo, utilizando uma estratégia de supressão de localizações a partir de Q.

O Algoritmo Tranon anonimiza conjuntos de semi-identificadores de tamanho i de maneira iterativa, onde i representa a quantidade de localizações que um adversário conhece. Ou seja, assumindo que um adversário conhece ao todo m localizações de um indivíduo, nosso algoritmo primeiramente anonimiza semi-identificadores de tamanho $i = 1$, em seguida de tamanho $i = 2$, até que i alcance o valor m, garantindo que o novo conjunto anonimizado D' atende a propriedade k^m-anonimato. Ambas as fases de Descoberta dos SI e Anonimização são executadas para cada valor de i, iniciando em conjuntos de tamanho 1 e incrementando até o valor final m. O Algoritmo 1 mostra o pseudocódigo da nossa estratégia.

O algoritmo tem como entrada um conjunto de dados de trajetória D, um número $m > 0$ de localizações conhecidas por um adversário e um suporte mínimo $k > 1$. A saída do Algoritmo Tranon é um conjunto de dados D' que obedece ao modelo de privacidade k^m-anonimato, garantindo que um adversário que conhece até m localizações de qualquer trajetória em D' não é capaz de reidentificar um indivíduo com probabilidade maior que $1/k$.

Primeiramente, o algoritmo inicializa o conjunto D' com os dados originais em D (linha 2). Logo após, uma fase de MapReduce é executada em D' com o objetivo de encontrar os semi-identificadores Q, de tamanho i (linha 5). Após esta etapa, Q é utilizado como entrada para o processo de anonimização de D'. É nesta fase que ocorre a supressão das localizações de

Algoritmo 1: Algoritmo Tranon

Entrada: Um conjunto de dados D, parâmetros k e m
Saída: Uma versão k^m-anônima D' do conjunto original D

```
1  início
2      D' ← D;
3      para i ← 1 até m faça
4          // Etapa I: Descoberta dos semi-identificadores;
5          Q ← CalcularSI(D', k, i);
6          // Etapa II: Anonimização;
7          D' ← Anonimizar(D', Q);
8      fim
9      retorne D'
10 fim
```

tamanho i (linha 7). Ao final das m iterações, D' é retornado ao usuário e pode então ser utilizado para publicação.

4.2 DESCOBERTA DE SEMI-IDENTIFICADORES

O número de possíveis semi-identificadores existentes em um conjunto de trajetórias de tamanho i pode ser calculado pela seguinte fórmula:

$$|Q| = \sum_{i=1}^{m} C \binom{L}{i}$$

C representa as combinações de L localizações dado i escolhas. A complexidade dessa descoberta é da ordem do número de combinações produzidas, ou seja, O(2^m), uma vez que o cálculo de Q produz exatamente $2^m - 1$ combinações. Devido ao alto esforço computacional envolvido no cálculo destes semi-identificadores, a computação de Q na nossa solução é realizada de maneira distribuída, utilizando o paradigma MapReduce. As fases de mapeamento e redução da nossa solução são descritas nos Algoritmos 2 e 3, respectivamente.

4.2.1 Função Map

Nossa função de mapeamento recebe como entrada um conjunto de trajetórias D, além dos parâmetros k e i. Primeiramente, para cada trajetória no conjunto D, a função de mapeamento gera todas as combinações de localizações de tamanho i. Para cada combinação gerada, são retornados pares <chave, valor>, onde chaves representam combinações de localizações de tamanho i e o valor sempre igual a 1. Esta informação será utilizada para descobrir quantas vezes subtrajetórias iguais aparecem no conjunto D.

Considere o exemplo da Figura 9. Suponha que existam quatro nós computacionais que irão processar as funções de mapeamento e redução. Neste exemplo, cada nó é responsável por processar uma trajetória. Contudo na prática, um nó é capaz de computar n trajetórias.

Algoritmo 2: CalcularSI - Map

Entrada: Um conjunto de dados D, parâmetros k e i
Saída: Pares <chave, valor>, onde a chave representa combinações de localizações de tamanho i e valor igual a 1

```
1 início
2     para cada T em D faça
3         para cada t em combinacao(T, i) faça
4             emita(t, 1)
5         fim
6     fim
7 fim
```

Considera-se que os parâmetros de entrada são $i = 2$ e $k = 2$. Conforme a Seção 2.1.4.2, a ordem das localizações de uma subtrajetória não é considerada no cálculo do suporte, pois assume-se que o conhecimento adversário sobre indivíduos é independente da ordem em que essas localizações aparecem. Por esse motivo, o conhecimento adversário é dito reflexivo: $(a,b) = (b,a)$. Antes de iniciar o mapeamento, as localizações das trajetórias são ordenadas com base nos seus identificadores, para atender a propriedade reflexiva na geração das chaves. No exemplo da Figura 9 considera-se uma ordenação alfabética das localizações.

Seguindo o Algoritmo 2 (linhas 2 e 3), para cada trajetória são computadas todas as combinações de localizações de tamanho 2 (dois pontos de trajetória), e assim é emitido um par <t, 1> (linha 4), onde a chave t representa localizações de tamanho 2 com valor igual a 1. Em outras palavras, é dito que t aparece uma vez na trajetória computada.

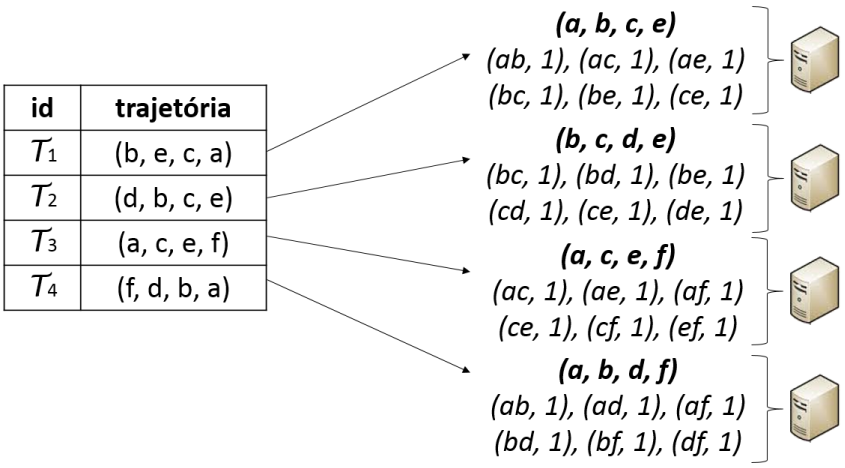


Figura 9 – Exemplo de uma função de mapeamento considerando o número de localizações conhecidas $i = 2$ e suporte $k = 2$.

4.2.2 Função Reduce

A função de redução utiliza a saída da função de mapeamento e agrega todos os pares intermediários que possuem a mesma chave para gerar o conjunto final e descobrir os semi-identificadores em D. Dessa forma, a função de redução soma todos os valores pertencentes a mesma chave, emitindo novos pares <chave, valor>. Aqui, a chave continua a representar subconjuntos de tamanho i, mas o valor é tido como o número de vezes que aquele subconjunto aparece nos dados de trajetórias. Tal informação corresponde a frequência de cada combinação em D, isto é, a frequência de uma subtrajetória.

Algoritmo 3: CalcularSI - Reduce

Entrada: Conjunto de pares $< x, 1 >$, e os parâmetros k e i.
Saída: Conjunto de semi-identificadores de tamanho i.

```
1 início
2     para cada par < x, valores > faça
3         soma ← valores.soma();
4         se soma < k então
5             emitir(x, soma)
6         fim
7     fim
8 fim
```

Como resultado da etapa de mapeamento, considere o exemplo da Figura 10. Para cada par <t, 1> emitido no passo anterior (linha 2), caso possuam a mesma chave t, seus valores são somados para produzirem um novo par <t, soma(t)> (linha 3). Contudo, como o número de combinações pode crescer exponencialmente, emite-se apenas pares cujos valores são menores que o suporte k, i.e., emite-se os semi-identificadores (linha 5). No exemplo da Figura 10, pares com valores $k \geq 2$, como $(ab, 2)$, $(ac, 2)$, $(ce, 3)$, etc. não são emitidos. Por fim, são encontrados os semi-identificadores considerando os parâmetros $i = 2$ e $k = 2$. São eles: $(ad), (bf), (cd), (cf), (de), (df)$ e (ef).

Caso um usuário malicioso utilize esse tipo de ataque e possua o conhecimento de que um indivíduo percorreu as localizações (b, f), por exemplo, ele consegue afirmar com 100% de certeza que sua trajetória é (f, d, b, a), atribuindo-a ao identificador T_4. Em contrapartida, caso ele possua o conhecimento de que um indivíduo percorreu as localizações (b, e), ele não consegue afirmar com mais de 50% de certeza qual a trajetória é referente ao indivíduo.

4.3 ANONIMIZAÇÃO

O segundo passo da solução proposta tem como objetivo produzir uma versão k^m-anônima D' a partir de D. Para isso, é utilizado o resultado da descoberta dos semi-identificadores Q, computado no passo anterior, e assim selecionadas apenas algumas localizações a serem anonimizadas. Dessa forma, ao invés de remover todas as localizações pertencentes ao conjunto

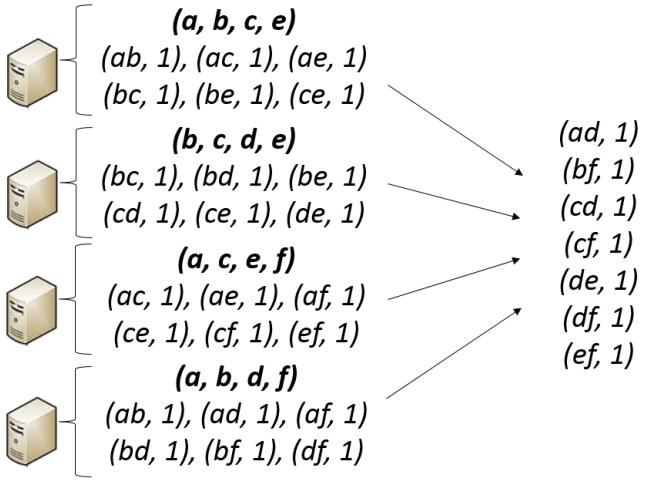


Figura 10 – Exemplo de uma função de redução considerando o número de localizações conhecidas $i = 2$ e suporte $k = 2$.

de semi-identificadores, é selecionado um conjunto H dentre os elementos de Q que minimiza a quantidade de localizações removidas e, consequentemente, a perda de informação.

A ideia da anonimização é selecionar os pontos mais frequentes dentre aqueles já infrequentes, pertencentes ao conjunto Q, e em seguida removê-los do conjunto D. Em outras palavras, uma vez que os semi-identificadores representam o conjunto de localizações mais infrequentes, o propósito é remover apenas as localizações que mais aparecem, ou seja, que são mais frequentes dentre as infrequentes, pois elas já representam um conjunto suficiente a ser anonimizado, ao invés de anonimizar todas as localizações distintas que pertencem a Q. Dessa maneira, o objetivo da solução é minimizar o tamanho do conjunto H, de forma que, após removidos do conjunto de dados original, atenda a propriedade k^m-anonimato.

O conjunto H de elementos mais frequentes dentre os semi-identificadores pode ser definido como um *Hitting Set* de Q. A seguir, define-se *Hitting Set* no contexto do problema e em seguida propõe-se uma estratégia para resolução do menor conjunto H.

4.3.1 Hitting Set

Definição 4 *Dado um conjunto $L = \{l_1, l_2, ..., l_n\}$ (universo) de diferentes localizações visitadas por objetos móveis, e um conjunto Q de semi-identificadores não vazios, cuja união é igual ao conjunto universo (i.e., $\cup_{qid \in Q}, qid = L$), um conjunto H é dito Hitting Set de Q se e somente se:*

$$H \subseteq L \wedge \forall_{qid \in Q} : H \cap qid \neq \emptyset$$

Em outras palavras, um Hitting Set H de Q é um subconjunto $H \subseteq L$ tal que H contém pelo menos um elemento de cada subconjunto em Q. Conforme mencionado anteriormente, a perda de informação está diretamente relacionada ao tamanho do conjunto H. Dessa forma, a solução proposta visa minimizar a quantidade de localizações a serem removidas, i.e., $|H|$.

Considere o conjunto $Q = \{(ad), (bf), (cd), (cf), (de), (df), (ef)\}$ de semi-identificadores calculados na etapa anterior. A Figura 11 mostra vários exemplos de Hitting Sets que contém pelo menos um elemento de cada subconjunto em Q. Esses elementos são destacados em negrito. O conjunto $H_3 = \{d, f\}$ é considerado mínimo pois não há outro Hitting Set cuja cardinalidade é menor do que 2, neste exemplo.

H_1 = {a, c, e, f}
(ad), (b**f**), (**c**d), (**cf**), (d**e**), (d**f**), (**ef**)

H_2 = {b, c, d, e}
(a**d**), (**b**f), (**cd**), (**c**f), (**de**), (**d**f), (**e**f)

H_3 = {d, f}
(a**d**), (b**f**), (c**d**), (c**f**), (**d**e), (**df**), (e**f**)

Figura 11 – Exemplos de Hitting Sets que contém pelo menos um elemento de cada subconjunto em Q.

Entretanto, conforme demonstrado em (GAREY; JOHNSON, 1979), o problema de encontrar uma solução ótima (mínima) para Hitting Set é considerado NP-difícil. Por esse motivo, utiliza-se uma estratégia gulosa para calcular um valor aproximado da solução ótima, e assim minimizar a perda de informação no momento da anonimização.

4.3.2 Estratégia Gulosa de Anonimização

Para resolver o problema de minimização do conjunto H é proposto um algortimo baseado em estratégia gulosa que recebe como entrada um conjunto de dados D e um conjunto de semi-identificadores de tamanho i, e, por meio de supressões de localizações, retorna uma versão k^i-anônima do conjunto D. O Algoritmo 4 mostra o pseudocódigo da estratégia proposta.

O Algoritmo funciona da seguinte maneira: enquanto o conjunto Q de semi-identificadores possui elementos, i.e., localizações, a frequência de cada um deles é calculada a fim de descobrir qual localização é mais frequente. Quando esse elemento é encontrado, ele é removido de todas as trajetórias do conjunto D, sendo esse o primeiro elemento do conjunto H (Hitting Set) que representa a perda de informação.

Uma vez removido do conjunto D, agora é a vez de removê-lo do conjunto de semi-identificadores. Para cada semi-identificador no conjunto Q, verifica-se a existência da localização mais frequente dentro do semi-identificador. Caso exista, o semi-identificador é removido de Q. Assim, garante-se que cada semi-identificador do conjunto Q possui pelo menos um elemento de H, atendendo a Definição 1.

Ainda considerando o conjunto $Q = \{(ad), (bf), (cd), (cf), (de), (df), (ef)\}$ de semi-identificadores calculados na etapa anterior, a Figura 12 mostra como é calculado o Hitting Set de Q seguindo o Algoritmo 4. Primeiramente (linha 3) é calculada a frequência de cada

Algoritmo 4: Anonimização - Estratégia Gulosa

Entrada: Um conjunto de dados D e um conjunto de semi-identificadores de tamanho i.
Saída: Uma versão k^i-anônima do conjunto D

```
1  início
2      enquanto Q ≠ ∅ faça
3          FQ ← calcularFrequencia(Q);
4          maisFrequente ← FQ.dequeue();
5          D ← suprimir(maisFrequente, D) ;
6          para cada qid ∈ Q faça
7              se maisFrequente ∈ qid então
8                  Q ← Q \ qid;
9              fim
10         fim
11     fim
12     retorne D
13 fim
```

elemento dentro do conjunto Q. A localização a aparece uma vez no conjunto Q, b também aparece uma vez, c duas vezes e assim por diante. Após calculada a frequência de cada elemento, eles são armazenados em uma pilha FQ, e o que possui maior valor é selecionado para compor o *Hitting Set* (linha 4). Neste caso, a localização d pois possui maior frequência em Q, $freq(d) = 4$. A localização f também possui frequência $freq(f) = 4$ em Q. Neste caso, a função retorna o primeiro elemento de maior frequência encontrado, ou seja, d é selecionado para compor o *Hitting Set*.

No passo seguinte (linha 5), a localização d é removida do conjunto de dados D. Para cada semi-identificador pertencente ao conjunto Q (linha 6), o algoritmo verifica se ele contém o elemento mais frequente d (linha 7). Se essa condição for verdadeira, o semi-identificador qid é removido do conjunto de dados Q (linha 8). Enquanto o conjunto Q não é vazio (linha 1), i.e., enquanto não há um *Hitting Set* que contém pelo menos um elemento de cada subconjunto em Q, o processo de anonimização continua. Na segunda iteração, após removidos os semi-identificadores que contém a localização d, o novo conjunto $Q = \{(bf), (cf), (ef)\}$ passa a ser o conjunto de semi-identificadores a ser considerado. Novamente é calculada a frequência das localizações no conjunto Q. Assim, as localizações b, c e e aparecem uma vez em Q, enquanto a localização f aparece três vezes. Como f possui maior frequência dentre as localizações restantes, f é selecionado para compor o *Hitting Set*. Consequentemente a localização f é removida dos conjuntos D (linha 5) e Q (linha 8). Não havendo mais elementos em Q, o algoritmo para e retorna o conjunto D anonimizado.

4.3.3 Corretude do algoritmo *Tranon*

Para verificar que a nossa estratégia gulosa de anonimização é correta, utiliza-se o seguinte Lema:

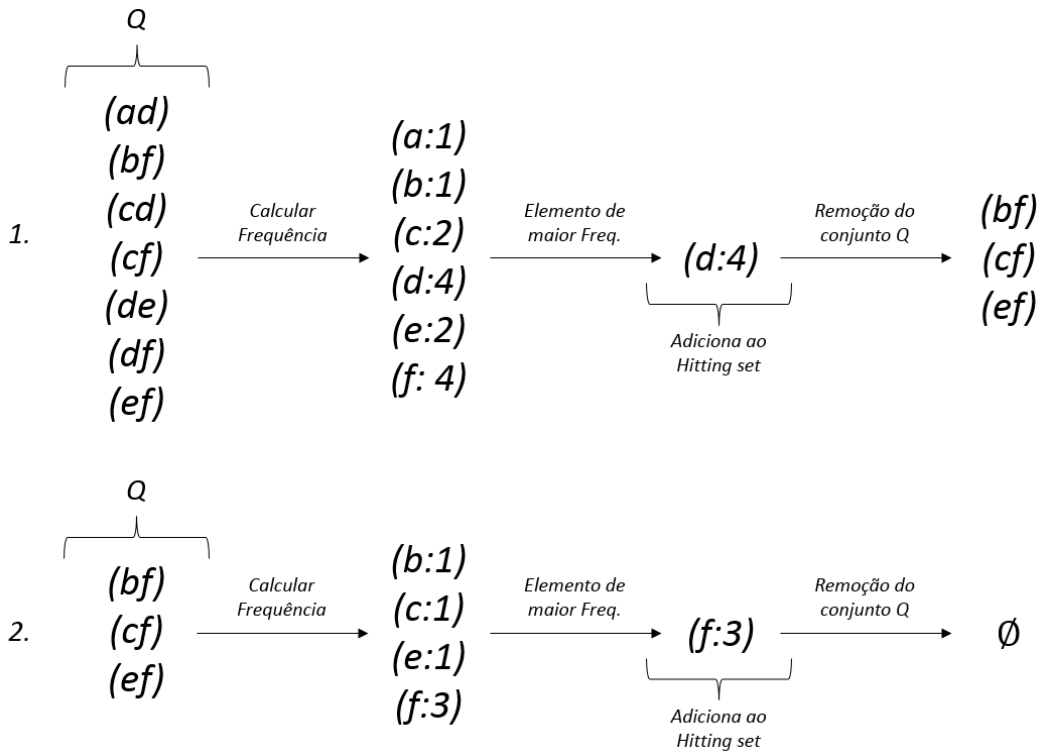


Figura 12 – Execução do algoritmo de anonimização sobre um conjunto Q de semi-identificadores.

Lema 1 *Dado um conjunto D, um limiar de anonimização $k > 1$ e um conjunto L de diferentes localizações visitadas por objetos móveis, tal que $L = \{l_1, l_2, ..., l_n\}$, um conjunto Q de semi-identificadores não vazios de tamanho m e um Hitting Set H de Q formado por localizações em L. Após a remoção das localizações pertencentes a H do conjunto de dados original D, o novo conjunto gerado D' passa a ser k^m-anônimo.*

Considere o conjunto de dados D da Figura 13 e um limiar de anonimização $k = 2$. Conforme a figura, o conjunto de localizações $L = \{a, b, c, d, e, f\}$. O conjunto de semi-identificadores de tamanho 2 em D foi calculado na etapa de descoberta de semi-identificadores conforme Figura 12, sendo representado por $Q = \{(ad), (bf), (cd), (cf), (de), (df), (ef)\}$, bem como o *Hitting Set* $H = \{d, f\}$. De acordo com o Lema 1, após removidas as localizações $\{d, f\}$ do conjunto D, o novo conjunto gerado D' é particularmente 2^2-anônimo, i.e., adversários com conhecimento de quaisquer duas localizações em L não podem associar indivíduos a seus registros em D' com probabilidade maior que $1/2$.

Prova do Lema 1: vamos provar o Lema 1 por indução na variável m, ou seja, na quantidade de localizações que um adversário conhece. Considere o caso base para $m = 1$. Dessa forma, o conjunto de semi-identificadores Q possui tamanho $m = 1$. Uma vez que H contém pelo menos uma localização de cada $qid \in Q$, o conjunto H é representado pelo próprio conjunto Q, pois Q possui apenas uma localização. Logo, o conjunto de localizações H removidas de D representam o próprio conjunto de semi-identificadores, o que garante a propriedade $k^1 - anonimato$, já que foram removidos todos os semi-identificadores Q de D. Como hipótese,

considere que após a remoção em D das localizações pertencentes a H, do conjunto de semi-identificadores Q de tamanho m, o novo conjunto gerado D' é k^m-anônimo. Como passo indutivo, considere que, se a hipótese é verdadeira para Q de tamanho m, então é verdadeira para Q de tamanho $m+1$. Mais uma vez, dado que H contém pelo menos uma localização de cada $qid \in Q$ (por definição), pelo menos uma localização será removida do conjunto Q. Se isso ocorre, os dados restantes em Q possuirão tamanho máximo de m localizações, que por hipótese, atende a propriedade do k^m-anonimato.

A seguir é demonstrado um exemplo prático de funcionamento do algoritmo Tranon, desde a etapa de descoberta de semi-identificadores até o processo de anonimização das trajetórias.

4.3.4 Exemplo prático

Neste exemplo, considera-se como parâmetros de entrada o conjunto de dados de trajetória da Figura 13 (o mesmo utilizado nas etapas anteriores), e as variáveis $k = 2$ e $m = 3$. Dessa forma, pretendemos transformar o conjunto D em um conjunto D' aplicando a nossa solução para garantir a propriedade 2^3-anonimato, ou seja, cada trajetória não pode ser associada a um indivíduo por um atacante com probabilidade menor que $1/2$, dado que ele conhece até 3 pontos de qualquer trajetória. Neste exemplo assume-se também que são utilizados 4 nós computacionais para processar os algoritmos de maneira distribuída.

id	trajetória
T_1	(b, e, c, a)
T_2	(d, b, c, e)
T_3	(a, c, e, f)
T_4	(f, d, b, a)

Figura 13 – Exemplo de um conjunto de dados de trajetória D a ser publicado contendo seis localizações distintas.

Como forma de preservar a identidade dos indivíduos pertencentes ao conjunto D, inicia-se o processo de anonimização considerando que o conhecimento adversário sobre indivíduos é de qualquer localização de tamanho $i = 1$. Na primeira etapa da solução proposta, é executada uma fase de *MapReduce* no conjunto de dados original D a fim de descobrir quais os semi-identificadores de tamanho $i = 1$ a serem considerados na fase de anonimização. Na fase de mapeamento, são gerados pares <chave, valor>, onde a chave representa a localização individual, já que combinações de tamanho 1 retornam o próprio elemento, e o valor igual a 1. Uma vez que todas as localizações individuais $L = \{a, b, c, d, e, f\}$ possuem suporte maior ou igual a 2, a fase de redução retorna nenhum semi-identificador de tamanho $i = 1$. Assim, não é necessária a etapa de anonimização das trajetórias, pois o conjunto D já atende a propriedade

$2^1 - anonimato$. Todo esse processo é mostrado na Figura 14.

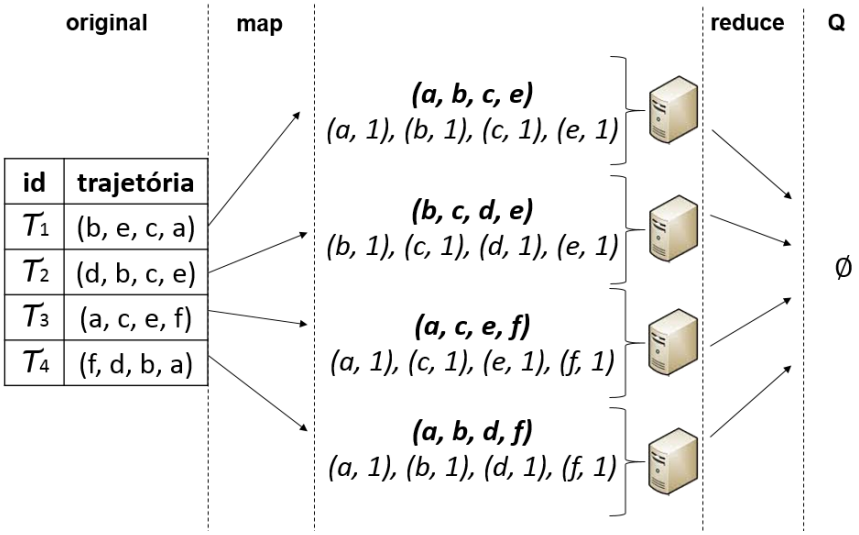


Figura 14 – Processo de anonimização do conjunto de dados de trajetória D considerando os parâmetros $i = 1$ e $k = 2$.

Considera-se agora o conhecimento adversário sobre indivíduos sendo quaisquer duas localizações em D, ou seja, $i = 2$. Na etapa de *MapReduce*, para cada trajetória no conjunto D, a função de mapeamento gera todas as combinações de localizações de tamanho 2 de cada trajetória. Para cada combinação gerada, são retornados pares <chave, valor>, onde a chave representa as combinações de localizações de tamanho 2 e o valor igual a 1. Na redução, o conjunto de subtrajetórias que possuem suporte menor que 2 representam os semi-identificadores para $i = 2$. São eles: $(ad), (bf), (cd), (cf), (de), (df), (ef)$.

Inicia-se então o processo de anonimização baseado no conjunto Q. Assim, é calculada a frequência de cada elemento no conjunto e então selecionada a localização d, que possui frequência 4, sendo esta a maior em Q. O elemento d é então adicionado ao Hitting Set e os elementos em Q que contém d são removidos. Remove-se também todas as localizações d do conjunto D. Uma vez que o conjunto Q não é vazio após a remoção dos elementos que contínham d, uma nova etapa de anonimização é executada, calculando novamente a frequência dos elementos e selecionando a localização f para compor o Hitting Set. Remove-se novamente os elementos em Q que contém f e todas as ocorrências da localização f no conjunto D. Após esse processo, o conjunto Q é vazio e a computação segue para $i = m = 3$. A Figura 15 apresenta o processamento dessa etapa para $i = 2$.

Após serem removidas as localizações d e f do conjunto D, uma nova iteração considerando $i = 3$ é executada, finalizando o processo de anonimização. Na etapa de descoberta de semi-identificadores, a função de mapeamento gera todas as combinações de elementos de tamanho 3 das localizações remanescentes, ou seja, combinações de elementos de tamanho 3 sem considerar as localizações d e f. A Figura 16 mostra como ocorre o mapeamento e a redução para $i = 3$. Note que a trajetória T_4 possui apenas duas localizações. Por esse motivo não são geradas combinações de tamanho 3. Na fase de redução, dois conjuntos de localizações são

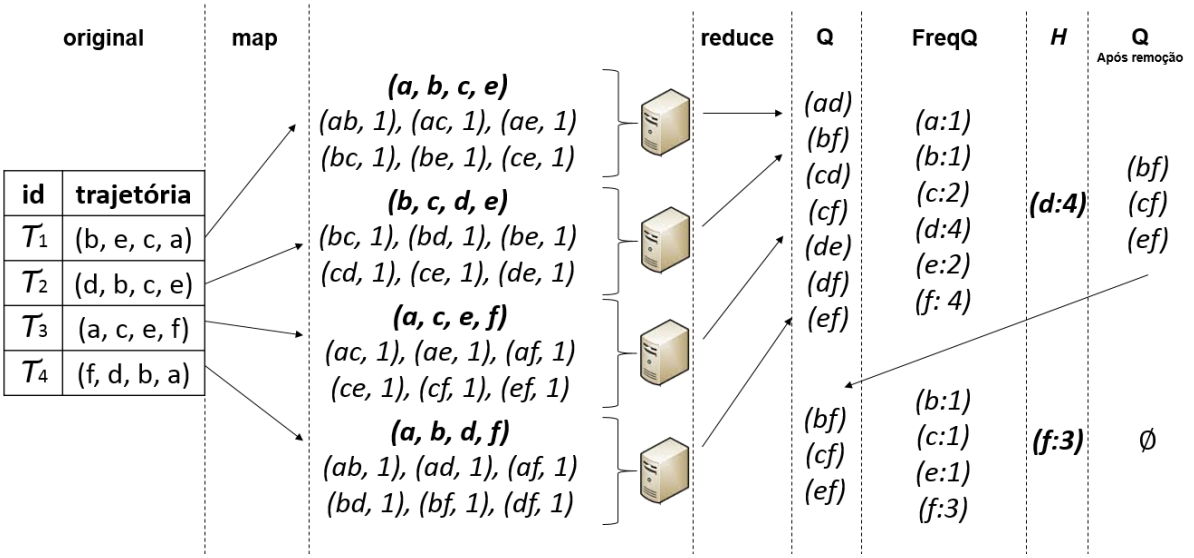


Figura 15 – Processo de anonimização do conjunto de dados de trajetória D considerando os parâmetros $i = 2$ e $k = 2$.

considerados semi-identificadores, são eles: (abc) e (abe).

Descoberto os semi-identificadores de tamanho $i = 3$, a frequência de cada localização é computada e então é selecionado o elemento a para compor o Hitting Set, pois possui frequência 2, sendo esta a maior em Q. Assim, a localização a é removida do conjunto D e todos os semi-identificadores que contém a também são removidos de Q. Finalmente, Q é vazio, i atingiu o valor máximo $m = 3$ e então é retornado para o usuário o conjunto de dados D' que atende ao modelo de privacidade $2^3 - anonimato$. A Figura 17 mostra o resultado final da anonimização, comparando o conjunto de dados original D e o conjunto a ser publicado D'.

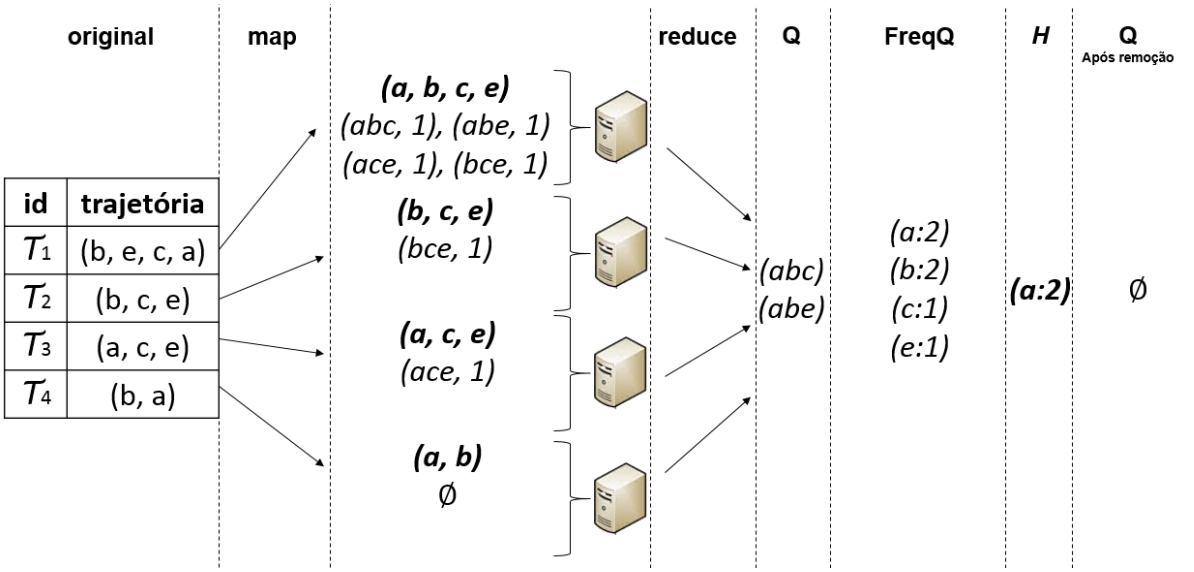


Figura 16 – Processo de anonimização do conjunto de dados de trajetória D considerando os parâmetros $i = m = 3$ e $k = 2$.

id	trajetória
T_1	(b, e, c, a)
T_2	(d, b, c, e)
T_3	(a, c, e, f)
T_4	(f, d, b, a)

id	trajetória
T_1	(b, e, c)
T_2	(b, c, e)
T_3	(c, e)
T_4	(b)

Figura 17 – Conjunto de dados original D e a transformação em um conjunto de dados anonimizado D' 2^3-anônimo.

Uma vez liberado para publicação o conjunto de dados da Figura 17, o adversário que conhece até três localizações, ou seja, todas do conjunto liberado $L = (b, c, e)$, não é capaz de inferir qualquer registro a um indivíduo com probabilidade maior que $1/2$. Por exemplo, suponha que um adversário conhece todas as três localizações (b, c, e) em D' de um determinado indivíduo. Ele não pode inferir que a trajetória desse indivíduo é T_1, uma vez que T_2 também possui essas três localizações. O mesmo acontece para adversários que possuem o conhecimento de duas, ou mesmo uma localização, em D'.

4.4 CONCLUSÃO

Neste capítulo apresentamos nossa solução de preservação de privacidade, denominada Tranon, a partir de uma publicação de dados de trajetória. A abordagem proposta utiliza o paradigma *MapReduce* de computação distribuída para calcular semi-identificadores e uma técnica de supressão para anonimizar dados de trajetórias. Nosso método de anonimização seleciona apenas as localizações essenciais que compõem os semi-identificadores por meio do *Hitting Set*, tendo como objetivo anonimizar o mínimo de localizações possíveis e assim preservar a identidade de indivíduos contra ataque de ligação.

5 AVALIAÇÃO EXPERIMENTAL

Este capítulo apresenta uma série de resultados coletados a partir de análise experimental do algoritmo Tranon. Foram observados aspectos e métricas de avaliação em termos de eficiência e tempo de execução, com o objetivo de comprovar a qualidade da solução proposta de preservação de privacidade na publicação de dados de trajetória em diferentes conjuntos de dados.

Primeiramente analisamos a utilidade de um conjunto de dados D' computado pelo Algoritmo Tranon e comparamos à estratégia seqAnon (POULIS et al., 2013a), variando os parâmetros k e m e observando a perda de informação em ambas as abordagens. A estratégia seqAnon foi escolhida para comparação por ser mais semelhante à proposta de preservação de privacidade em dados de trajetória. Ambas utilizam o modelo de privacidade k^m-anonimato e visam proteger a privacidade contra ataque de ligação, garantindo que usuários maliciosos não podem inferir registros a seus respectivos indivíduos com probabilidade maior que $1/k$, dado que esses adversários conhecem até m localizações de cada indivíduo.

Em seguida, realizou-se uma análise comparativa entre os tempos de execução dos algoritmos Tranon e seqAnon em uma abordagem centralizada, variando o tamanho do conjunto de dados. Foram utilizados dois conjuntos de dados nos experimentos. Eles são descritos na Seção 5.1. Por fim, verificou-se o tempo de execução das etapas de descoberta de semi-identificadores e de anonimização do algoritmo Tranon distribuído, variando também o tamanho do conjunto de dados e observando o desempenho da solução proposta.

5.1 CONJUNTOS DE DADOS

Os dados utilizados para avaliação experimental consistiram de dois conjuntos de trajetórias pertencentes a indivíduos em um determinado intervalo de tempo. O primeiro deles, denominado STM, é um conjunto de dados real fornecido pela *Société de Transport de Montréal* (STM)[1], agência de transporte público da cidade de Montreal, Canadá. O segundo conjunto de dados de trajetória foi gerado sinteticamente pela ferramenta *Brinkhoff's Data Generator* (BRINKHOFF, 2002), denominado *SYN*, o qual é empregado por muitos dos trabalhos relacionados já apresentados nesta dissertação (POULIS et al., 2013b; ABUL et al., 2008; YAROVOY et al., 2009).

A seguir são detalhados cada um desses conjuntos, bem com suas características mais relevantes. Ao final desta seção, é apresentada uma tabela contendo as estatísticas dos dados apresentados.

5.1.1 STM

O conjunto de dados STM consiste no deslocamento de passageiros ao longo da rede metroviária da cidade de Montreal. Toda vez que os usuários utilizavam seu cartão de acesso à

[1] <www.stm.info>

estação de metrô, dados eram coletados pela estação, armazenando assim informações sobre o identificador da trajetória e sua localização geográfica na rede de transporte pública da cidade. Assim, os dados foram liberados pela *Société de Transport de Montréal* com o intuito de serem utilizados para diversos fins, tais como análise estatística em geral, ou auxílio de pesquisadores na melhoria do transporte público, entre outros.

Em particular, o conjunto contém dados de 130.707 trajetórias de passageiros ao longo de 68 estações de metrô. Os dados foram coletados em um intervalo de 21 dias, no mês de Outubro de 2011. O tamanho médio das trajetórias é de 11,28 localizações. As 68 localizações do conjunto STM podem ser visualizadas na rede metroviária da cidade de Montreal, como mostra a Figura 18.

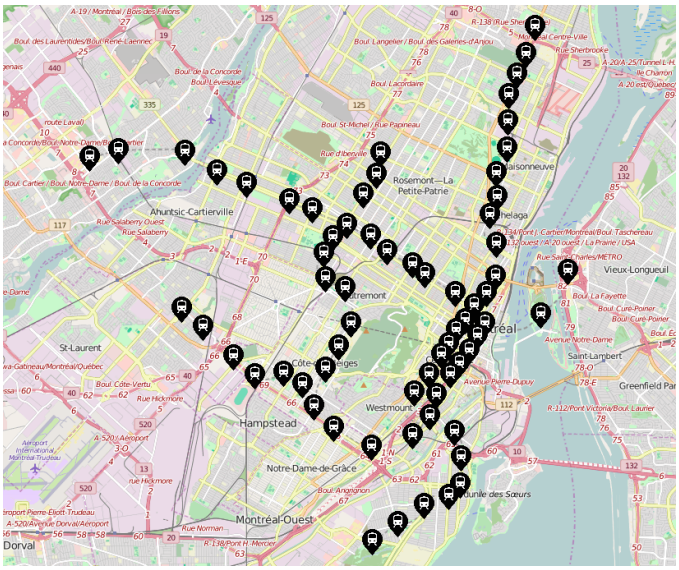


Figura 18 – Localizações geográficas da rede metroviária da cidade de Montreal utilizadas pelo conjunto de dados *STM*.

5.1.2 SYN

O segundo conjunto de dados utilizado na experimentação foi construído sinteticamente utilizando a ferramenta *Brinkhoff's Data Generator* (BRINKHOFF, 2002) e possui dados em maior escala quando comparado ao conjunto de dados reais STM. Em resumo, para gerar trajetórias utilizando a ferramenta, escolhe-se a região a qual os dados serão gerados, o número de veículos iniciais, o número de unidades de tempo, i.e., localizações percorridas a cada intervalo de tempo, e por fim o número de veículos adicionais a cada intervalo de tempo percorrido. Essa geração de dados sintéticos foi disponibilizada pelo *MNTG: Minnesota Web-based Traffic Generator* [2] e pode ser visualizada na Figura 19.

De maneira geral, o conjunto possui 400 mil trajetórias sintéticas de veículos na cidade de Oldenburg, Alemanha, distribuídas em 2.075 localizações distintas e com tamanho

[2] <mntg.cs.umn.edu/tg/index.php>

médio das trajetórias igual a 34,86.

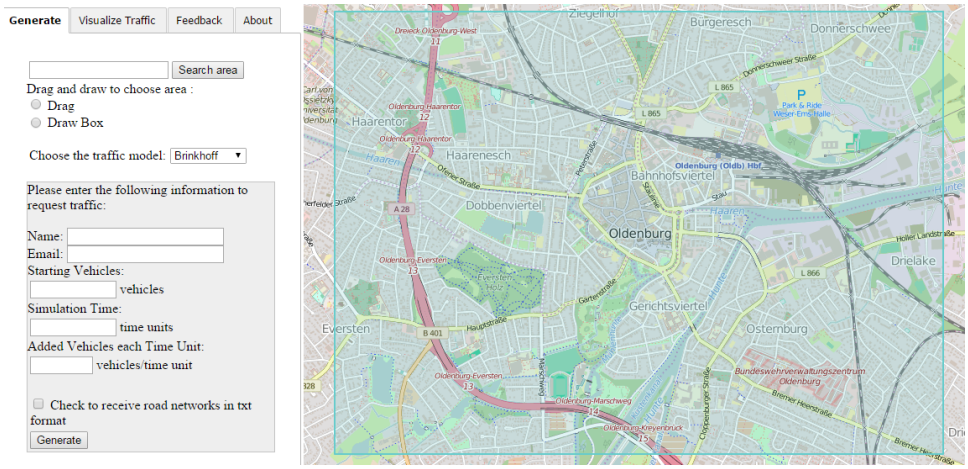


Figura 19 – Exemplo de utilização da ferramenta de geração de dados sintéticos na cidade de Oldenburg, Alemanha.

A fim de avaliar o desempenho da solução a medida que o volume de dados cresce, este conjunto foi dividido em três instâncias, denominadas SYN200k, SYN300k e SYN400k. Cada uma dessas instâncias possui 200 mil, 300 mil e 400 mil trajetórias, respectivamente. As métricas de cada instância são descritas na Tabela 7.

Conjunto de dados	Número de trajetórias	$\|L\|$	Tamanho médio das trajetórias
STM130k	130,707	68	11.28
SYN200k	200,000	185	24.74
SYN300k	300,000	650	29.90
SYN400k	400,000	2075	34.86

Tabela 7 – Métricas dos conjuntos de dados sintéticos utilizados na experimentação.

5.2 AMBIENTE E CONFIGURAÇÃO

O algoritmo Tranon proposto neste trabalho foi implementado utilizando a linguagem Java e os experimentos foram conduzidos em uma infraestrutura de nuvem privada na Universidade Federal do Ceará (UFC). Utilizou-se ao todo 9 máquinas virtuais, das quais uma atuou como nó mestre e as outras oito como nós escravos. Cada máquina virtual possuía sistema operacional Ubuntu, versão 14.04, com processador Xeon 2.4 GHz, 4 núcleos de processamento, 4 GB de memória RAM e 200 GB de capacidade em disco.

A implementação da estratégia seqAnon, escolhida para comparação, deu-se também em Java. É importante ressaltar que o algoritmo seqAnon considera a ordem das localizações de uma subtrajetória no cálculo do suporte, i.e., dadas duas subtrajetórias t_1 e t_2, e um conjunto de dados D, $sup(t_1, D) \neq sup(t_2, D)$, aumentando assim o número de localizações a serem

anonimizadas e, consequentemente, a perda de informação. Como forma de comparação mais precisa da nossa técnica com a estratégia em (POULIS et al., 2013a), ordenamos as trajetórias por ordem crescente de IDs das localizações. Uma vez que o conhecimento adversário é independente da ordem em que as localizações aparecem, trajetórias ordenadas por IDs estabelecem que $sup(t_1, D)$ seja igual ao $sup(t_2, D)$ no algoritmo seqAnon, visto que, por estarem ordenadas, as localizações sempre serão detectadas na mesma ordem. Dessa forma, ambos os algoritmos passaram a ter o mesmo tratamento no cálculo do suporte.

5.3 EXPERIMENTOS

A solução proposta neste trabalho foi avaliada em termos de eficiência e desempenho. Em termos de eficiência, analisou-se a utilidade dos dados, verificando a perda de informação, i.e., a quantidade de localizações anonimizadas após a execução dos algoritmos. Variou-se tanto o suporte k no intervalo [2, 64] utilizando potências de 2 a cada execução, quanto o parâmetro m no intervalo [1, 5]. Em termos de desempenho, analisou-se o tempo de execução dos algoritmos, variando o tamanho do conjunto de dados, conforme apresentados na Seção 5.1. Para cada experimento variamos um determinado parâmetro enquanto os outros mantiveram-se fixos, em seus respectivos valores padrões. A Tabela 8 apresenta os parâmetros utilizados na experimentação, com seus respectivos valores, tendo como valores em negrito os padrões de cada parâmetro.

k	2, **4**, 8, 16, 32, 64
m	1, 2, **3**, 4, 5
Dataset	**STM130k**, SYN200k, SYN300k, SYN400k

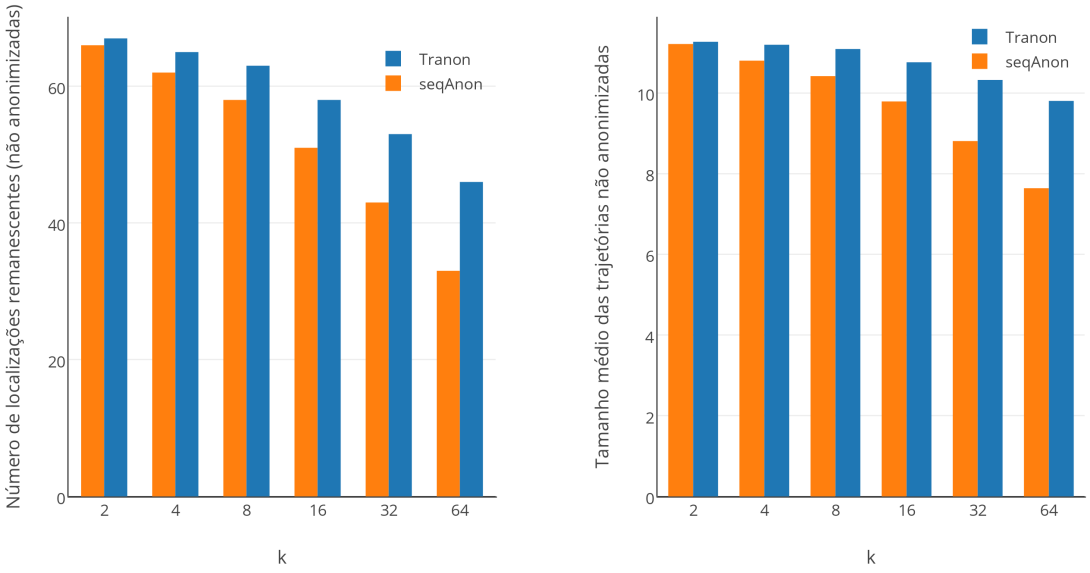
Tabela 8 – Parâmetros utilizados na experimentação. Os valores em negrito são fixados ao se variar um outro parâmetro.

5.3.1 Análise da utilidade dos dados

Para medir a eficiência da solução proposta, avaliou-se a perda de informação quando os parâmetros k e m são variados dentro dos intervalos estabelecidos, utilizando o conjunto de dados STM130k. Calculou-se então o número de localizações remanescentes (não anonimizadas) e o tamanho médio das trajetórias que também não foram anonimizadas. São essas as duas métricas avaliadas para cálculo da perda de informação dos algoritmos Tranon e seqAnon. A seguir é apresentada a análise de resultados em cada um dos experimentos variando k e m.

5.3.1.1 Variação do k

No primeiro experimento, variou-se o suporte k de acordo com os valores mostrados na Tabela 8. Estes resultados são exibidos na Figura 20, em escala logarítmica. Em particular, a Figura 20a apresenta o número de localizações não anonimizadas em função de k. Como esperado, o número de localizações remanescentes da anonimização diminui a medida que o



(a) (b)

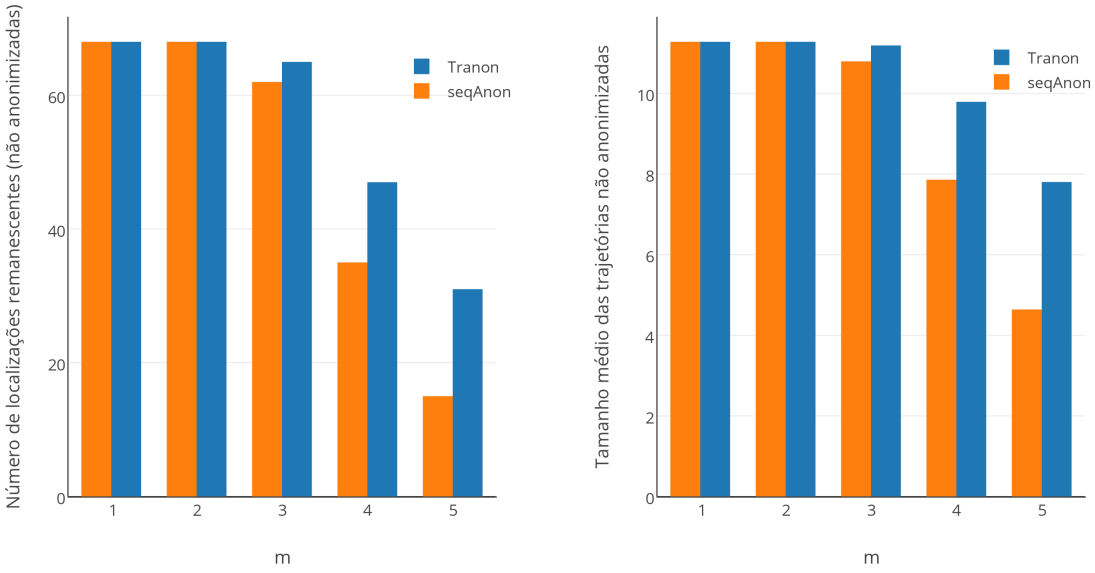
Figura 20 – (a) Número de localizações remanescentes (não anonimizadas), (b) Tamanho médio das trajetórias não anonimizadas, variando o parâmetro k.

suporte k aumenta. Isso acontece devido a probabilidade de reidentificação $1/k$ se tornar cada vez menor. Quanto menor for a probabilidade de reidentificar um indivíduo, mais localizações deverão ser anonimizadas para garantir essa afirmação. Por esse motivo, o número de localizações não anonimizadas diminui à medida que o suporte k é incrementado. Na Figura 20b, o tamanho médio das trajetórias também diminui à medida que k cresce, confirmando o resultado anterior de que mais localizações são anonimizadas quando k é incrementado, a fim de garantir a propriedade k^m-anonimato.

Observa-se que em ambos os resultados a perda de informação do algoritmo Tranon é menor quando comparada ao algoritmo seqAnon. A explicação para a anonimização de um menor número de localizações se dá devido ao fato do algoritmo Tranon adotar o *Hitting Set* como estratégia de anonimização e utilizar uma implementação gulosa de seleção de localizações a serem removidas. O algoritmo seqAnon anonimiza as localizações baseado no seu vizinho mais próximo, utilizando distância euclidiana, conforme visto em 3.1.4. Essa não é uma estratégia muito eficiente visto que o vizinho mais próximo de uma certa localização não necessariamente pertence a um semi-identificador em um conjunto de dados. Caso fosse utilizada uma métrica de seleção de localizações a serem anonimizadas, como por exemplo o *Hitting Set*, em conjunto com a estratégia de anonimização proposta pelo próprio algoritmo seqAnon, menos localizações seriam anonimizadas e, consequentemente, menor seria a perda de informação.

5.3.1.2 Variação do m

Neste experimento variamos o número de localizações m conhecidas por um adversário, no intervalo de 1 a 5. Estes resultados são mostrados na Figura 21. A Figura 21a apresenta



(a) (b)

Figura 21 – (a) Número de localizações remanescentes (não anonimizadas), (b) Tamanho médio das trajetórias não anonimizadas, variando o parâmetro m.

o número de localizações não anonimizadas em função de m, com k fixado no valor 4. Também como esperado, o número de localizações remanescentes da anonimização diminui à medida que o adversário conhece mais localizações m. Observa-se que neste experimento a curva gerada pela perda de informação é menos suave quando comparado à variação do parâmetro k. Isso ocorre devido ao fato do número de combinações possíveis, à medida que um adversário conhece mais localizações, crescer exponencialmente. Por esse motivo, o número de localizações não anonimizadas diminui mais drasticamente à medida que o parâmetro m é incrementado. A Figura 21b apresenta um comportamento semelhante. À medida que m cresce, o tamanho médio das trajetórias diminui de forma mais acentuada.

Mesmo com um impacto mais significativo na variação do parâmetro m em relação ao parâmetro k, observa-se que o algoritmo Tranon é melhor em eficiência que o algoritmo comparado, visto que o número de localizações originais remanescentes da anonimização é maior na nossa estratégia, acarretando em uma menor perda de informação. O motivo é o mesmo do experimento anterior. O algoritmo Tranon utiliza o *Hitting Set* como método de seleção de localizações a serem anonimizadas, diminuindo de forma significativa a quantidade de localizações anônimas na variação do parâmetro m. A estratégia em (POULIS et al., 2013a), por adotar uma técnica de anonimização sem seleção prévia de localizações, possui uma perda de informação bem mais considerável. Por exemplo, na Figura 21a, quando $m = 5$, o número de localizações não anonimizadas do algoritmo Tranon possui valor 31 (das 68 localizações originais), comparado ao valor 15 do algoritmo seqAnon, retendo mais que o dobro de localizações após a anonimização.

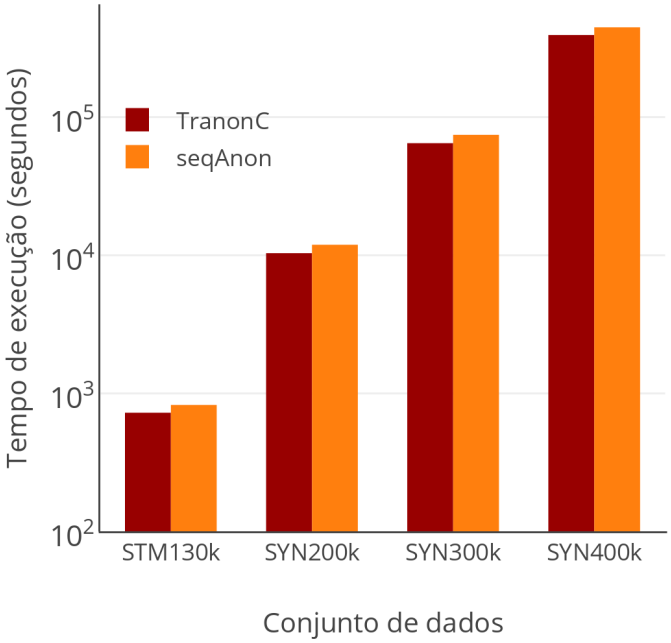


Figura 22 – Tempos de execução dos algoritmos TranonC e seqAnon variando o tamanho dos conjuntos de dados.

5.3.2 Análise de desempenho

Neste conjunto de experimentos analisou-se o tempo de execução dos algoritmos Tranon e seqAnon variando o tamanho do conjunto de dados, fixando os parâmetros $k = 4$ e $m = 3$ conforme a Tabela 8. O objetivo é analisar o comportamento da solução à medida que o volume de dados cresce.

5.3.2.1 TranonC x seqAnon

Para comparar o desempenho da solução proposta com a estratégia seqAnon, implementamos uma versão centralizada do algoritmo Tranon, denominada TranonC. Dessa forma, o cálculo dos semi-identificadores é feito apenas em uma máquina, gerando todas as combinações possíveis de cada trajetória de forma local. Com isso, o cálculo dos semi-identificadores foi realizado de forma centralizada em ambas as estratégias.

A Figura 22 mostra os tempos de execução dos algoritmos TranonC e seqAnon. À medida que o volume do conjunto de dados é ampliado, ambos os tempo de execução aumentam de forma significativa. Isso acontece devido a complexidade do cálculo dos semi-identificadores conforme visto na Seção 4.2. Contudo, a nossa solução apresentou leve ganho de desempenho quando comparada ao algoritmo seqAnon, uma vez que foi utilizada a técnica de supressão das localizações, ao invés de generalização.

5.3.2.2 Tranon distribuído

Por fim, comparamos os tempos de execução do algoritmo Tranon distribuído com TranonC e seqAnon. Como no experimento anterior, variamos o tamanho dos conjuntos de dados, medindo seus respectivos tempos de execução. A Figura 23a apresenta esse resultado em escala logarítmica.

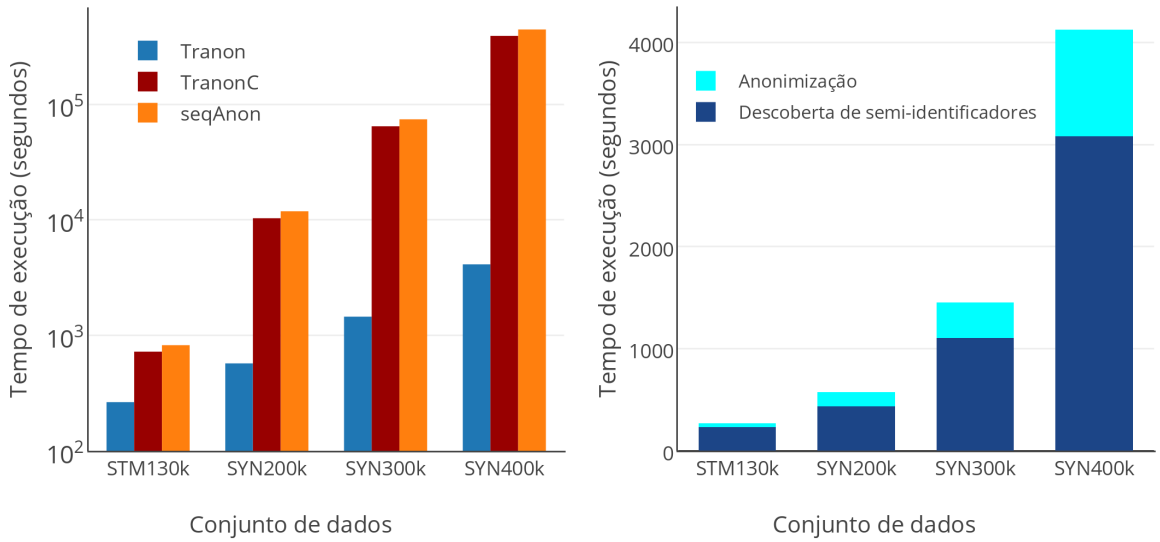


Figura 23 – (a) Tempos de execução dos algoritmos Tranon, TranonC e seqAnon variando o tamanho dos conjuntos de dados, (b) Tempos de execução de cada etapa do algoritmo Tranon.

Quando comparado às estratégias centralizadas, o algoritmo Tranon é aproximadamente $2,7$ vezes mais rápido que o TranonC e $3,1$ vezes mais rápido que o seqAnon, para o conjunto de dados STM130. Para o conjunto SYN200, o algoritmo Tranon distribuído é executado em 572 segundos, enquanto nos algoritmos seqAnon e TranonC esses valores são 3,29 horas e 2,87 horas, respectivamente. Entretanto, quando executados no conjunto de dados SYN400, o algoritmo Tranon é 95 e 108 vezes mais rápido que as abordagens TranonC e seqAnon, respectivamente, comprovando o bom desempenho da solução proposta.

A Figura 23b exibe os tempos de execução de cada etapa do algoritmo Tranon. Observa-se que a ampliação do conjunto de dados faz com que ambos os tempos de descoberta de semi-identificadores e de anonimização aumentem. Por exemplo, no conjunto de dados STM130, a descoberta de semi-identificadores ocorre em 227 segundos, enquanto a anonimização é computada em 37 segundos. Já no conjunto de dados SYN400, esses valores são 3080 e 1043 segundos, respectivamente. Nota-se que o tempo de descoberta de semi-identificadores possui um aumento mais acentuado devido a complexidade dessa descoberta.

5.4 CONCLUSÃO

Neste capítulo foram apresentados os resultados dos experimentos realizados tanto com dados reais quanto com dados sintéticos. O conjunto de dados reais foi coletado da cidade de Montreal, Canadá, e fornecido pela *Société de Transport de Montréal* (STM). O outro conjunto de dados foi gerado utilizando a ferramenta *Brinkhoff's Data Generator* (BRINKHOFF, 2002). Compara-se a nossa abordagem com a solução proposta em (POULIS et al., 2013a), denominada seqAnon. Os experimentos mostraram que o algoritmo Tranon possui menor perda de informação e melhor desempenho, quando comparado ao algoritmo seqAnon.

6 CONSIDERAÇÕES FINAIS

6.1 CONCLUSÃO

Serviços baseados em localização estão cada vez mais presentes em domínios sociais e empresariais, integrando atividades diárias de pessoas e facilitando suas vidas. Tais serviços têm gerado quantidades significativas de dados de trajetória, e, para continuarem contribuindo com benefícios à sociedade, muitas vezes estes dados necessitam estar em domínio público. No entanto, a publicação destes dados pode levar a sérios riscos de violação de privacidade devido à existência dos semi-identificadores, possibilitando que adversários descubram a identidade de indivíduos em um conjunto de dados público. Como forma de evitar esse tipo de descoberta, nesta dissertação apresentamos uma solução de preservação de privacidade para publicação de dados de trajetória que previne ataques de descoberta de identidade, impedindo que um adversário com conhecimento prévio sobre indivíduos não seja capaz de associá-los a registros em um conjunto de dados publicado, e minimizando a perda de informação a qual estes dados estão submetidos no processo de anonimização.

A solução apresentada baseou-se no paradigma *MapReduce*, que permite armazenar e processar grandes conjuntos de dados de forma distribuída. Para garantir a efetividade da solução proposta, adotamos o modelo de privacidade k^m-anonimato, garantindo que um adversário que conhece até m localizações de qualquer trajetória publicada não é capaz de reidentificar um indivíduo com probabilidade maior que $1/k$. Nosso algoritmo de preservação de privacidade, denominado Tranon, é dividido em duas fases. A primeira consiste em encontrar os semi-identificadores, i.e., localizações que permitem associar um indivíduo a seu respectivo registro, em um conjunto de dados. Para isso, implementamos uma fase distribuída de mapeamento, seguido por uma fase de redução, que emite os semi-identificadores de tamanho i, dado um suporte mínimo k. A segunda fase seleciona localizações chaves a partir dos semi-identificadores encontrados com o objetivo de suprimí-las do conjunto de dados e garantir a propriedade k^m-anonimato. Para tal, utilizamos uma estratégia gulosa que visa encontrar o mínimo *Hitting Set* e assim anonimizar (suprimir) a menor quantidade de localizações possíveis.

Ambas as fases do algoritmo Tranon são executadas $\forall i\ 1 \leq i \leq m$, de acordo com princípio apriori. Dessa forma, o cálculo dos semi-identificadores e a anonimização do conjunto de dados para conhecimento adversário de tamanho $i+1$ não leva em consideração localizações removidas anteriormente para conhecimento adversário de tamanho i.

Nós comparamos nossa abordagem com a estratégia seqAnon, proposta anteriormente em (POULIS et al., 2013a). Foram realizados experimentos utilizando dados sintéticos e reais com o objetivo de comprovar a qualidade da solução proposta em termos de eficiência e desempenho. Os experimentos variando os parâmetros k e m mostraram que nossa solução apresenta uma menor perda de informação quando comparada à estratégia seqAnon, sendo assim mais eficiente. Experimentos variando o conjunto de dados também demonstraram que o algoritmo Tranon foi executado siginificativamente mais rápido quando comparado ao algoritmo seqAnon centralizado, conforme esperado.

6.2 TRABALHOS FUTUROS

Como trabalho futuro, pretendemos investigar uma abordagem que generalize as localizações ao invés de suprimí-las, garantindo que o dono dos dados possa escolher a melhor técnica de anonimização a ser utilizada, baseada no contexto da publicação a qual ele planeja realizar. Para isso, deve-se alterar a etapa 2 da solução atual e elaborar um algoritmo que minimize uma determinada função de perda de informação de acordo com a semântica das trajetórias oriundas da estratégia de generalização. Assim, o desafio é fazer com que esse algoritmo seja escalável quando executado em grandes volumes de dados de trajetória, uma vez que estratégias de generalização podem demandar maior esforço computacional na criação de agrupamentos de trajetórias ou de localizações generalizadas.

Outro aspecto a ser investigado é a preservação de privacidade de localizações sensíveis de indivíduos. O objetivo é previnir tanto a descoberta de informação quanto a descoberta de atributo em um conjunto de dados. Pretendemos então modificar o modelo atual de privacidade k^m-anonimato para algum outro que atenda a este requisito, como $(k,l)^m$-anonimato, LKC-privacidade, $(K,C)_L$-privacidade, etc., de tal forma que a perda de informação seja mínima tanto para semi-identificadores quanto para atributos sensíveis.

REFERÊNCIAS

ABUL, O.; BONCHI, F.; NANNI, M. Never walk alone: Uncertainty for anonymity in moving objects databases. In: **Proc. of the 24th ICDE**. [S.l.: s.n.], 2008. p. 376–385.

BONCHI, F.; LAKSHMANAN, L. V. S.; WANG, W. H. Trajectory anonymity in publishing personal mobility data. **SIGKDD Explorations**, p. 30–42, 2011.

BRINKHOFF, T. A framework for generating network-based moving objects. **GeoInformatica**, p. 153–180, 2002.

CHEN, B.; KIFER, D.; LEFEVRE, K.; MACHANAVAJJHALA, A. Privacy-preserving data publishing. **Foundations and Trends in Databases**, p. 1–167, 2009.

CHEN, R.; FUNG, B. C. M.; MOHAMMED, N.; DESAI, B. C.; WANG, K. Privacy-preserving trajectory data publishing by local suppression. **Inf. Sci.**, p. 83–97, 2013.

DEAN, J.; GHEMAWAT, S. Mapreduce: Simplified data processing on large clusters. In: **6th Symposium on Operating System Design and Implementation**. [S.l.: s.n.], 2004. p. 137–150.

FUNG, B. C. M.; WANG, K.; CHEN, R.; YU, P. S. Privacy-preserving data publishing: A survey of recent developments. **ACM Comput. Surv.**, 2010.

GAREY, M. R.; JOHNSON, D. S. **Computers and Intractability: A Guide to the Theory of NP-Completeness**. [S.l.]: W. H. Freeman, 1979.

HE, Y.; NAUGHTON, J. F. Anonymization of set-valued data via top-down, local generalization. **PVLDB**, p. 934–945, 2009.

HU, H.; CHEN, Q.; XU, J. VERDICT: privacy-preserving authentication of range queries in location-based services. In: **29th ICDE**. [S.l.: s.n.], 2013. p. 1312–1315.

IYENGAR, V. S. Transforming data to satisfy privacy constraints. In: **Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada**. [S.l.: s.n.], 2002. p. 279–288.

JR., R. J. B.; AGRAWAL, R. Data privacy through optimal k-anonymization. In: **Proceedings of the 21st International Conference on Data Engineering, ICDE 2005, 5-8 April 2005, Tokyo, Japan**. [S.l.: s.n.], 2005. p. 217–228.

LÄMMEL, R. Google's mapreduce programming model - revisited. **Sci. Comput. Program.**, p. 1–30, 2008.

MACHANAVAJJHALA, A.; KIFER, D.; GEHRKE, J.; VENKITASUBRAMANIAM, M. *L*-diversity: Privacy beyond *k*-anonymity. **TKDD**, 2007.

MEYERSON, A.; WILLIAMS, R. On the complexity of optimal k-anonymity. In: **Proc. of the 23rd ACM SIGACT-SIGMOD-SIGART**. [S.l.: s.n.], 2004. p. 223–228.

MOHAMMED, N.; FUNG, B. C. M.; DEBBABI, M. Walking in the crowd: anonymizing trajectory data for pattern analysis. In: **Proc. of the 18th ACM CIKM**. [S.l.: s.n.], 2009. p. 1441–1444.

POULIS, G.; SKIADOPOULOS, S.; LOUKIDES, G.; GKOULALAS-DIVANIS, A. Distance-based kˆm-anonymization of trajectory data. In: **IEEE 14th International Conference on Mobile Data Management**. [S.l.: s.n.], 2013. p. 57–62.

POULIS, G.; SKIADOPOULOS, S.; LOUKIDES, G.; GKOULALAS-DIVANIS, A. Select-organize-anonymize: A framework for trajectory data anonymization. In: **13th ICDM Workshops**. [S.l.: s.n.], 2013. p. 867–874.

SAMARATI, P. Protecting respondents' identities in microdata release. **IEEE Trans. Knowl. Data Eng.**, p. 1010–1027, 2001.

SILVA, T. L. C. da; MACÊDO, J. A. F. de; CASANOVA, M. A. Discovering frequent mobility patterns on moving object data. In: **Proc. of the 3rd ACM SIGSPATIAL**. [S.l.: s.n.], 2014. p. 60–67.

SOUSA, F. R.; MOREIRA; O, L.; MACÊDO, J.; MACHADO, J. C. Gerenciamento de dados em nuvem: Conceitos, sistemas e desafios. In: **SBBD**. [S.l.: s.n.], 2010. p. 101–130.

SRIKANT, R.; AGRAWAL, R. Mining sequential patterns: Generalizations and performance improvements. In: **5th EDBT**. [S.l.: s.n.], 1996. p. 3–17.

SWEENEY, L. k-anonymity: A model for protecting privacy. **International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems**, p. 557–570, 2002.

TAN, P.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining**. [S.l.]: Addison-Wesley, 2005.

TERROVITIS, M.; MAMOULIS, N. Privacy preservation in the publication of trajectories. In: **9th International Conference on Mobile Data Management**. [S.l.: s.n.], 2008. p. 65–72.

TRUJILLO-RASUA, R.; DOMINGO-FERRER, J. On the privacy offered by (k, δ)-anonymity. **Inf. Syst.**, p. 491–494, 2013.

WHITE, T. **Hadoop - The Definitive Guide: MapReduce for the Cloud**. [S.l.]: O'Reilly, 2009.

WILLISON, D.; EMERSON, C.; SZALA-MENEOK., K.; GIBSON, E.; SCHWARTZ, L.; WEISBAUM, K. Access to medical records for research purposes: Varying perceptions across research ethics boards. **Journal of Medical Ethics 34**, p. 308–314, 2008.

WONG, R. C.; FU, A. W. **Privacy-Preserving Data Publishing: An Overview**. [S.l.]: Morgan & Claypool Publishers, 2010.

YAROVOY, R.; BONCHI, F.; LAKSHMANAN, L. V. S.; WANG, W. H. Anonymizing moving objects: how to hide a MOB in a crowd? In: **12th EDBT**. [S.l.: s.n.], 2009. p. 72–83.