UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

GABRIELLE KARINE CANALLE

Usando relacionamentos entre atributos na Descoberta da Verdade do processo de Fusão de Dados

Recife

2023

GABRIELLE KARINE CANALLE

Usando relacionamentos entre atributos na Descoberta da Verdade do processo de Fusão de Dados

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciências da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Banco de Dados

Orientador (a): Profª Dra. Ana Carolina Brandão Salgado

Recife

2023

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

C213u Canalle, Gabrielle Karine
Usando relacionamentos entre atributos na descoberta da verdade do processo de fusão de dados / Gabrielle Karine Canalle. – 2023.
92 f.: il., fig., tab.

Orientadora: Ana Carolina Brandão Salgado.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.
Inclui referências e apêndices.

1. Banco de dados. 2. Integração de dados. 3. Fusão de dados. I. Salgado, Ana Carolina Brandão (orientadora). II. Título.

025.04 CDD (23. ed.) UFPE - CCEN 2023-95

Gabrielle Karine Canalle

“Usando relacionamentos entre atributos na Descoberta da Verdade do processo de Fusão de Dados”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Banco de Dados.

Aprovado em: 14/03/2023.

_____

Orientadora: Profa. Dra. Ana Carolina Brandão Salgado

BANCA EXAMINADORA

_____

Prof. Dr. Luciano de Andrade Barbosa
Centro de Informática / UFPE

_____

Prof. Dr. Fernando da Fonseca de Souza
Centro de Informática / UFPE

_____

Profa. Dra. Bernadette Farias Lóscio
Centro de Informática / UFPE

_____

Profa. Dra. Damires Yluska Souza Fernandes
Unidade Acadêmica de Informática / IFPB

_____

Prof. Dr. Marcelo Iury de Sousa Oliveira
Centro de Informática / UFPB

Dedico este trabalho aos meus pais, por todo amor, cuidado, criação e apoio para que eu sempre buscasse meus sonhos. Devo tudo a vocês.

AGRADECIMENTOS

Primeiramente, agradeço à Deus, por me dar forças para seguir firme, mesmo após tantos obstáculos. Não foram poucas as vezes que o pensamento de desistir surgiu em minha mente. A jornada de um doutorado é longa e, por muitas vezes, solitária. É uma roda gigante de emoções, pressão, alegria, desespero, motivação. Certos dias você se acha louca, em outros você acha que está fazendo um trabalho muito bom... e tem aqueles dias que você pensa: acho que isso não dá um doutorado. É, não é fácil. Encontrar motivação durante o passar do tempo foi o mais difícil. Principalmente durante o período de pandemia que foi tão complicado. Mas as emoções boas, as pessoas que te apoiam, e o resultado do seu trabalho duro, fazem valer a pena.

Agradeço à minha querida mãe, por me dar forças, me aconselhar, estar comigo, muitas vezes remotamente, me acalmando ou falando uma palavra de conforto. Era só uma fase, como você sempre disse.

Ao meu pai, que nunca mede esforços para me ajudar, me apoiar, e me direcionar em minhas decisões.

Ao meu irmão Gabriel, sem palavras para dizer como sou feliz em ser sua irmã. Te amo muito meu irmão.

Agradeço à minha orientadora Ana Carolina, por ter me acolhido desde minha chegada em Recife para o mestrado. Professora, registro aqui minha admiração por você, e meu orgulho por ter sido sua aluna. Muito obrigada por todos os ensinamentos que, com toda certeza, levarei para a vida.

Aos meus amigos e colegas, sei que muitas vezes perceberam minha ausência, mas sempre foram compreensivos.

Aos membros da banca, pelas valorizas contribuições realizadas.

E a todos, que de alguma maneira tenham contribuído para a concretização deste trabalho.

RESUMO

A Fusão de Dados é uma tarefa primordial quando se deseja integrar dados. Durante o processo de Integração de Dados podem ocorrer conflitos entre os valores de um mesmo objeto do mundo real. Na Fusão de Dados, a etapa responsável por identificar e resolver esses conflitos, descobrindo os valores verdadeiros, é chamada de Descoberta da Verdade. Devido à facilidade de publicação e compartilhamento de dados, muitos valores falsos são disponibilizados na *Web*, e as fontes de dados possuem qualidades variadas. A etapa de Descoberta da verdade é um processo iterativo no qual se utiliza a qualidade das fontes para verificar o grau de confiança dos valores providos por ela, ao mesmo tempo que a qualidade das fontes é calculada considerando-se os valores verdadeiros que ela provê. Deste modo, principalmente em cenários de dados na *Web*, a descoberta da verdade se torna um desafio. Neste sentido, este trabalho propõe considerar relacionamentos entre atributos, os quais serão utilizados para inserir conhecimento adicional na etapa de descoberta da verdade no processo de Fusão de Dados. Deste modo, o processo iterativo de avaliação de confiança dos valores e de confiabilidade das fontes será realizado com maior acurácia, não apenas com base nos dados de entrada, mas também no conhecimento adicional extraído dos relacionamentos. Experimentos foram realizados para avaliar a proposta em comparação com um algoritmo do estado da arte. Os resultados mostraram que a utilização de conhecimento adicional na descoberta da verdade melhora a precisão dos resultados.

Palavras-chaves: integração de dados; fusão de dados; descoberta da verdade.

ABSTRACT

Data Fusion is an essential task for Data Integration process. During a Data Integration process, conflicts may occur between values related to a same real-world object. Identifying and resolving these conflicts is the goal of the Truth Discovery step. Due to the ease of publishing and sharing data, many false values are made available on the Web. Also, data sources have widely varyied qualities. Truth Discovery is an iterative process in which the truth computation and source reliability estimation depend on each other. In this way, mainly in data scenarios on the Web, truth discovery becomes an even greater challenge. In this sense, this work proposes an approach for considering relationships between attributes, which will be used to insert additional knowledge about data in the Truth Discovery step in the Data Fusion process. Thus, the iterative process of evaluating the reliability of the values and the reliability of the sources will be performed with greater accuracy, not only based on input data, but also on additional knowledge extracted from relationships. Experiments were performed to evaluate our proposed approach in comparison with a state-of-the-art algorithm. The results demonstrated that the utilization of additional knowledge in truth discovery improves the precision of the results.

Keywords: data integration; data fusion; truth discovery.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

EM	Expectation Maximization
GAV	Global-as-view
JSON	JavaScript Object Notation
LP	Linear Programming
MAP	Maximum a Posteriori
RBM's	Restricted Boltzmann Machines
SGBD	Sistema Gerenciador de Banco de Dados
SQA	Consistent Query Answering
SQL	Structured Query Language
XML	Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO

Neste capítulo, descrevemos o contexto em que este trabalho está inserido. Apresentamos algumas motivações para realização da pesquisa a definição do problema que será abordado e a hipótese da pesquisa. Também indicamos as contribuições esperadas.

1.1 MOTIVAÇÃO E DEFINIÇÃO DO PROBLEMA

A era do *Big Data* trouxe consigo vários desafios. Entre a quantidade significativa de dados disponibilizados, muitos são contraditórios, redundantes, desatualizados e incompletos. Os dados que descrevem um mesmo objeto podem ser fornecidos por várias fontes, e por isto, podem ser conflitantes. Devido a esse fato, a Fusão de Dados ganha crescente atenção. O objetivo da Fusão de Dados é resolver os conflitos existentes entre os valores fornecidos por diferentes fontes, criando uma representação única para cada entidade.

Muitas pesquisas já foram realizadas na área de Fusão de Dados. As soluções iniciais eram baseadas em regras, como valor médio, valor mais recente e valor máximo. (BLEIHOLDER; NAUMANN, 2006; DONG; NAUMANN, 2009; BLEIHOLDER; NAUMANN, 2008; BERTI-ÉQUILLE; BORGE-HOLTHOEFER, 2015; BERETTA et al., 2018). Dentre essas soluções, a abordagem que mais se destacou foi o voto majoritário, o qual seleciona as respostas providas pela maioria das fontes. No entanto, nenhuma dessas abordagens propostas inicialmente considerava os diferentes níveis de qualidade das fontes de dados que os fornece.

Com a crescente quantidade de dados disponíveis na *Web*, a qualidade dos dados se tornou um problema crítico. São disponibilizados muitos dados falsos, o que introduziu um problema de veracidade dos dados. A veracidade se refere diretamente a problemas de inconsistência e qualidade de dados (BERTI-ÉQUILLE; BORGE-HOLTHOEFER, 2015). Com isso, as soluções tradicionais de Fusão de Dados foram se tornando insuficientes, principalmente devido aos diferentes níveis de qualidade das fontes de dados, o que essas abordagens desconsideravam.

O primeiro trabalho a discutir formalmente o problema de veracidade dos dados no processo de Fusão de Dados foi (YIN; HAN; YU, 2008), que denominou a tarefa como **descoberta da verdade**. Nesse trabalho, foi proposta a primeira solução para decidir valores verdadeiros entre dados conflitantes utilizando a qualidade das fontes de dados. O princípio básico dessas soluções é combinar iterativamente a estimativa de confiabilidade das fontes com a estimativa

de confiança do valor fornecido, seguindo o seguinte princípio: quanto maior a confiabilidade da fonte, maior a probabilidade de ela fornecer a verdade. A informação que é provida por fontes confiáveis, provavelmente é verdadeira (BERTI-ÉQUILLE; BORGE-HOLTHOEFER, 2015). Ao longo dos últimos anos, foram propostas diversas soluções que avaliam a confiabilidade das fontes e levam essa informação em consideração no processo de resolução de conflitos (DONG; BERTI-ÉQUILLE; SRIVASTAVA, 2009b; ZHAO et al., 2012; LI et al., 2016; LI et al., 2017; BROELEMANN; GOTTRON; KASNECI, 2017a; ZHANG et al., 2018).

Entretanto, em cenários de dados na *Web*, é comum a ocorrência do fenômeno *Long-tail*, em que a maioria das fontes fornece valores apenas para alguns atributos de um pequeno número de entidades, enquanto apenas algumas fontes cobrem vários atributos de muitas entidades. Para exemplificar esse contexto, considere como exemplo o cenário de vendas de roupas *online*. Nesse cenário, é comum que haja um pequeno número de produtos mais populares, de marcas muito conhecidas. Esses produtos, provavelmente são oferecidos por muitos vendedores diferentes, e possuem informações detalhadas e abrangentes. No entanto, quando se trata de produtos de marcas menores, oferecidos por apenas alguns vendedores, a disponibilidade de informações pode ser limitada.

Deste modo, principalmente nesse tipo de cenário, avaliar a confiabilidade das fontes e utilizá-la no processo de resolução de conflitos pode ser insuficiente (BROELEMANN; KASNECI, 2018; BERTI-ÉQUILLE; BORGE-HOLTHOEFER, 2015). A eficácia da estimativa de confiabilidade das fontes é fortemente afetada pelo número total de informações providas por cada fonte. Quando uma fonte de dados provê poucos dados, como a maioria das fontes nos cenários *Long-tail*, se torna um desafio estimar sua confiabilidade de forma precisa.

Com isso, foram elencadas as seguintes questões que visam ser respondidas nessa pesquisa:

- **Q1** -Como é possível descobrir os valores verdadeiros de cada atributo entre os valores conflitantes (incerteza e contradição), gerando uma representação única e completa de uma entidade em cenários *Long-tail*?
- **Q2** -Como é possível adaptar a etapa de descoberta da verdade para gerar resultados com maior precisão em cenários *Long-tail*?

Sendo assim, este trabalho propõe uma solução que aborda estes desafios. A proposta se baseia em utilizar relacionamentos entre os atributos na etapa de descoberta da verdade do processo de Fusão de Dados.

Os relacionamentos entre atributos são utilizados para obter conhecimento adicional sobre os dados. Este conhecimento é utilizado no processo de descoberta da verdade, tanto na avaliação de confiança dos valores quanto na estimativa de confiabilidade das fontes. Acreditamos que esses relacionamentos podem auxiliar na descoberta da verdade, principalmente em cenários *Long-tail*.

1.2 DEFINIÇÃO DA HIPÓTESE

Para buscar respostas para essas questões de pesquisa foi estabelecida a seguinte hipótese:

- **Hipótese** - Utilizar relacionamentos entre atributos pode auxiliar no processo de descoberta da verdade e melhorar a precisão do processo principalmente em cenários *Long-tail*.

1.3 OBJETIVO GERAL E OBJETIVOS ESPECÍFICOS

O objetivo geral deste trabalho é melhorar a precisão dos resultados da Descoberta da verdade introduzindo conhecimento adicional no processo por meio da utilização de relacionamentos entre atributos.

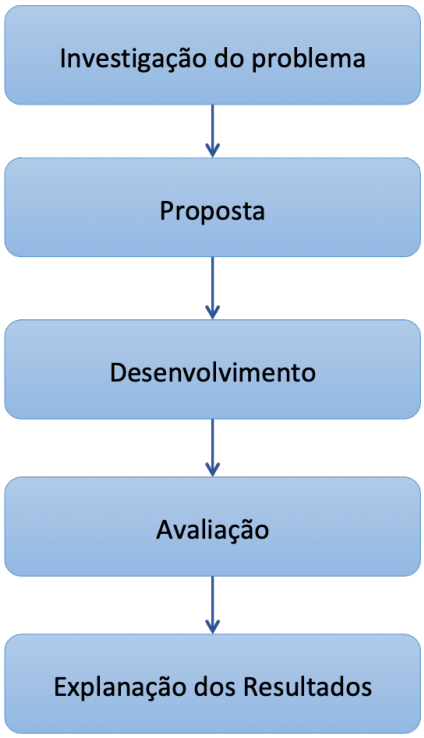
Como objetivos específicos, pode-se elencar:

- Apresentar uma formalização para o problema de Fusão de Dados considerando a utilização de relacionamentos entre atributos no processo;
- Realizar um levantamento do estado da arte, categorizando os trabalhos de acordo com suas principais características para lidar com a descoberta da verdade
- Propor uma abordagem para o processo de descoberta de relacionamentos, que terá como objetivo utilizar relacionamentos entre atributos por meio de regras que serão aplicadas nos dados. As regras são traduzidas para buscas em bases de conhecimento disponíveis ou API relacionadas ao domínio de dados abordado, no intuito de incorporar conhecimento adicional no processo de descoberta da verdade. A abordagem poderá ser aplicada em conjunto com algoritmos do estado da arte que podem ser adaptados de acordo com a solução proposta;
- Realizar a avaliação da abordagem proposta em diferentes cenários de dados.

1.4 METODOLOGIA DE PESQUISA

Quanto ao método de pesquisa, nesta pesquisa será adotada a abordagem quantitativa. No estudo quantitativo, o pesquisador utiliza a literatura como uma estrutura para questões ou hipóteses de pesquisa, e ao final do estudo para comparar os resultados da pesquisa com os resultados existentes na literatura (CRESWELL, 2010). O desenvolvimento da pesquisa se divide em cinco etapas, apontadas na Figura 1. A fase inicial da pesquisa foi de extenso estudo da literatura acerca de Integração de Dados com foco em Fusão de Dados e descoberta da verdade. Foi realizado um levantamento do estado da arte com um comparativo entre as soluções existentes e possíveis problemas em aberto. Com base nesse estudo foi identificado o problema e a hipótese deste trabalho. Esta etapa inicial gerou um *survey*, já publicado (CANALLE; SALGADO; LóSCIO, 2021).

Figura 1 – Etapas do desenvolvimento da pesquisa.



Fonte: Elaborada pela autora (2023).

Na segunda etapa propõe-se uma solução para lidar com descoberta da verdade em cenários *Long-tail* de maneira mais eficaz. Em seguida, na etapa de desenvolvimento, a solução foi desenvolvida na forma de protótipo, possibilitando a execução da descoberta da verdade utilizando a solução proposta. Na etapa de avaliação foram executados experimentos para avaliar a solução proposta, comparando os resultados do algoritmo modificado de descoberta

da verdade com o algoritmo original. Por fim, na última etapa são explanados os resultados obtidos e possíveis direções para trabalhos futuros.

1.5 ORGANIZAÇÃO DO DOCUMENTO

O restante deste documento está organizado da seguinte maneira:

- Capítulo 2 - No capítulo 2, são apresentados os principais conceitos acerca da Fusão de Dados. É apresentada uma classificação dos conflitos, e estratégias para lidar com eles, desde as propostas iniciais até as soluções mais avançadas;
- Capítulo 3 - Neste capítulo, são abordadas as soluções do estado da arte. São apresentadas as classificações para modelos de Fusão de Dados existentes na literatura e, por fim, é definida uma categorização própria. As soluções do estado da arte são apresentadas segundo a categorização proposta. Também são apresentados em uma seção separada os trabalhos que lidam com algum tipo de relacionamento nos dados;
- Capítulo 4 - No capítulo 4, é proposta a solução para utilizar relacionamentos entre atributos no processo de descoberta da verdade;
- Capítulo 5 - No capítulo 5, são apresentadas a experimentação e avaliação da estratégia proposta, listando resultados alcançados;
- Capítulo 6 - No capítulo 6, são apresentadas as conclusões do trabalho, limitações e direcionamentos de possíveis trabalhos futuros.

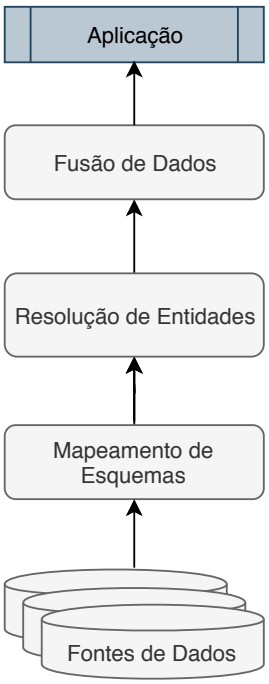
2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os conceitos fundamentais relacionados a esta pesquisa.

2.1 INTEGRAÇÃO DE DADOS

A área de Integração de Dados tem recebido crescente atenção no decorrer dos anos. Desde o surgimento da *Web*, a necessidade de integrar dados tem aumentado e impulsionado o desenvolvimento de diversas pesquisas (GOLSHAN et al., 2017; PAPOTTI; SANTORO, 2018). Com o surgimento da *Web* o número de fontes de dados disponíveis aumentou consideravelmente. Muitas das fontes de dados têm pouca ou nenhuma informação sobre os dados que ela armazena. Além disso, há um grande volume de informações duplicadas e conflitantes são encontrados na *Web*, criando uma maior demanda por soluções de Integração de Dados.

Figura 2 – Etapas do processo de Integração de Dados.



Fonte: adaptado de Dong e Naumann (2009)

A Integração de Dados tem como objetivo combinar dados distribuídos em fontes de dados distintas, heterogêneas e autônomas, provendo aos usuários uma visão unificada desses dados (LENZERINI, 2002; PAPOTTI; SANTORO, 2018). Segundo Dong e Naumann (2009), este processo se divide em três etapas principais, como mostra a Figura 2. A etapa de **Mapeamento de Esquemas** é responsável por encontrar correspondências entre os elementos

semanticamente correspondentes dos esquemas participantes do processo, e com base nessas correspondências gerar mapeamentos entre esses elementos. A partir desses mapeamentos, a etapa de **Resolução de Entidades** tem como objetivo encontrar múltiplas instâncias que se referem a uma mesma entidade do mundo real. Por fim, a **Fusão de Dados** é responsável por combinar as instâncias que se referem à mesma entidade, (identificados pelo processo de Resolução de Entidades) fundindo-as em uma única representação (DONG; NAUMANN, 2009).

Na próxima seção a etapa de Fusão de Dados será detalhada pois é o foco principal deste trabalho.

2.1.1 Fusão de Dados

A Fusão de Dados é uma etapa crucial, já que, no processo de Integração de Dados, as fontes participantes podem fornecer valores conflitantes para os mesmos atributos de uma entidade. Esses conflitos podem ocorrer devido a diferentes motivos, tais como informações desatualizadas, erros de digitação, interpretações inconsistentes da semântica, entre outros. Este problema pode se tornar ainda maior quando fontes de dados copiam dados umas das outras. Por exemplo, uma entidade pessoa pode possuir mais que um número de telefone (podendo conter números diferentes em fontes distintas), endereços desatualizados (devido a mudanças de endereço), ou ainda, erros ortográficos no nome (devido a erros de digitação).

Tabela 1 – Um exemplo de Fusão em um amostra de dados do censo.

	F1	**F2**	**F3**	**F4**	**Voto**	**Ground Truth**
Recife	1.555.000	1.637.834	null	1.537.704	1.537.704	**1.637.834**
São Paulo	12.107.000	12.176.866	null	12.176.866	12.176.866	**12.176.866**
Fortaleza	null	2.643.247	2.609.716	2.554.122	2.654.160	**2.643.247**

Fonte: Elaborada pela autora (2023).

Para exemplificar a Fusão de Dados e tornar mais fácil o entendimento, considere como exemplo a Tabela 1, onde são apresentados dados oriundos de 4 fontes de dados distintas, sobre a população de três cidades brasileiras. Na coluna *Ground Truth* são apresentados os valores corretos. A estratégia básica da Fusão de Dados é aplicar o *voto majoritário* (estratégia *Cry with the wolves*), ou seja, o valor fornecido pelo maior número de fontes de dados é dado como verdadeiro. No entanto, como podemos verificar pelo *Ground Truth* disponível na última coluna da Tabela, nem sempre o voto é uma boa estratégia, pois pode fornecer valores errôneos, como é o caso das cidades de **Recife** e **Fortaleza**. Isso ocorre principalmente em cenários de

dados na *Web*, onde as fontes possuem qualidades muito distintas umas das outras, e são extremamente dinâmicas. Além disso, a quantidade de valores ausentes na *Web* também é alta, o que torna essa estratégia ainda mais falha.

O desafio de descobrir os valores verdadeiros entre dados conflitantes de múltiplas fontes tem atraído muita atenção ao longo de anos, principalmente nas comunidades de Banco de Dados e Inteligência Artificial. E, tem sido abordado sob diferentes nomes: *Data Fusion*, *Fact-Checking*, *Truth Discovery*, *Truth finding*, entre outros. Inicialmente, foram propostas estratégias de Fusão de Dados divididas em três tipos: estratégias para ignorar conflitos, estratégias para evitar conflitos e estratégias para resolver conflitos, sendo a última a mais utilizada, conforme será detalhado na Seção 2.1.3. As outras nomenclaturas utilizadas como sinônimo para o problema de Fusão de Dados surgiram a partir do momento em que se tornou necessário analisar também características relacionadas à veracidade dos dados (Será detalhado na Seção 2.2.1). Atualmente, a maioria dos métodos de Fusão de Dados utiliza informações de qualidade das fontes para ajudar no processo de fusão, como a acurácia e a cobertura das fontes.

2.1.2 Classificação dos Conflitos

Uma questão importante no processo de Integração de Dados é a possibilidade de ocorrerem conflitos entre diferentes fontes de dados. Os conflitos entre as fontes de dados podem ocorrer em três níveis: (i) Nível de esquema: as fontes armazenam os dados em estruturas com esquemas distintos; (ii) Nível de instância: os dados são representados de diferentes maneiras; (iii); Nível de dados: diferentes valores nos dados que descrevem uma mesma entidade do mundo real (MOTRO; ANOKHIN, 2006; BLEIHOLDER; NAUMANN, 2008).

Para solucionar os conflitos de heterogeneidade e realizar a Integração de Dados diversas pesquisas têm sido realizadas. No nível de esquema, Mapeamento e Correspondência de Esquemas (RAHM; BERNSTEIN, 2001), no nível de instância, Resolução de Entidades (ELMAGARMID; IPEIROTIS; VERYKIOS, 2007), e no nível de dados, Fusão de Dados (BLEIHOLDER; NAUMANN, 2008), foco deste estudo.

Os conflitos no nível de dados podem surgir devido à existência de dados incompletos, incorretos ou desatualizados. Para um sistema de Integração de Dados, resolver os conflitos entre as várias fontes, identificando os valores verdadeiros, é uma tarefa crucial.

Os conflitos de dados são classificados, segundo Dong e Naumann (2009), em dois tipos:

incerteza e *contradição*. A incerteza é quando há um conflito entre um valor não-nulo e um ou mais valores nulos, todos utilizados para descrever a mesma propriedade de uma entidade real. Esse conflito é causado por informações faltantes em valores de atributos. O conflito de contradição ocorre quando dois ou mais valores não-nulos que descrevem a mesma propriedade de uma entidade são diferentes. Para exemplificar os tipos de conflitos existentes, considere novamente a Tabela 1, por exemplo, para a cidade de **Recife**, um exemplo de contradição são os valores fornecidos pelas fontes **F1** - 1.555.000, e **F2** - 1.637.834. Um exemplo de incerteza pode ser visto também para a cidade de **Recife**, pelos valores fornecidos pelas fontes **F3** - *null* e **F4** - 1.537.704.

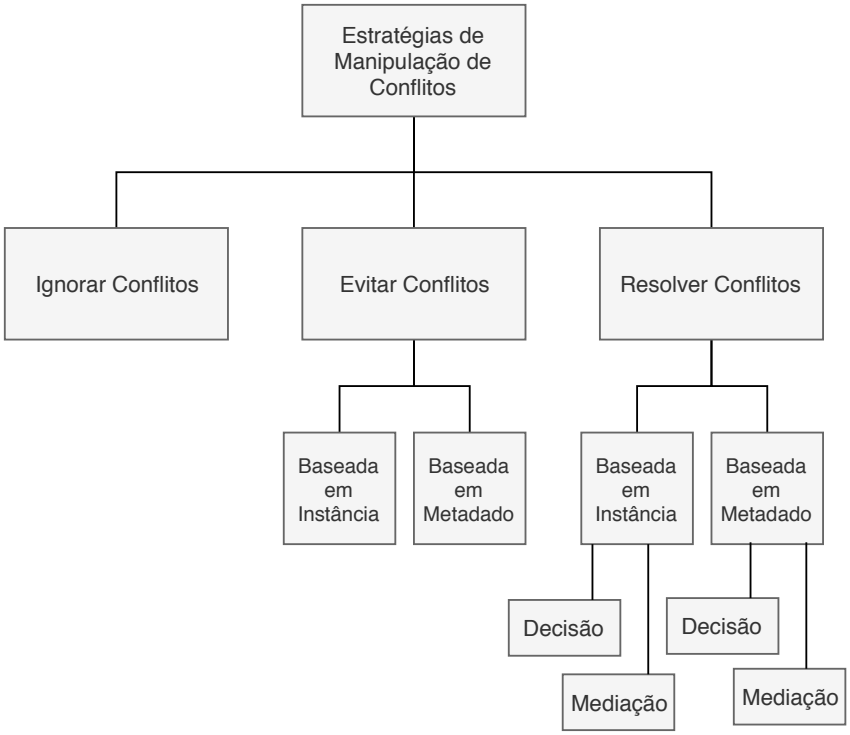
2.1.3 Estratégias de Manipulação de Conflitos

As estratégias de manipulação de conflitos descrevem como lidar com dados inconsistentes ao tomar decisões (ou não) sobre os conflitos como, por exemplo, combinar valores ou inventar um novo valor para criar uma representação única e consistente da instância durante a Fusão de Dados. Bleiholder e Naumann (2006) classificaram essas estratégias em três classes principais: **Ignorar conflitos**, **Evitar conflitos** e **Resolver conflitos**, como mostra a Figura 3. Os autores também mostraram como as estratégias são implementadas utilizando funções de manipulação de conflitos.

Ignorar conflitos - As estratégias de ignorar conflitos não tomam uma decisão sobre o que fazer com os dados conflitantes. Essas estratégias são sempre aplicáveis e fáceis de implementar. No entanto, podem produzir resultados inconsistentes. Como exemplos dessa classe de estratégias pode-se citar a *Pass it on*, e *Consider all possibilities*. A primeira simplesmente passa todos os valores conflitantes ao usuário ou aplicação, e permite que decidam como lidar com possíveis conflitos. A segunda, enumera todos os conflitos, e passa ao usuário todas as combinações possíveis de valores de atributos, criando até combinações que ainda não estão presentes nas fontes.

Evitar conflitos - Estratégias de evitar conflitos reconhecem a existência de conflitos em geral, mas não resolvem conflitos isolados. Tratam os dados conflitantes aplicando uma decisão única igualmente a todos os dados. Essas estratégias são subdivididas em duas classes: baseadas em metadado, que leva em consideração os metadados (como confiabilidade ou atualidade das fontes), ao tomar uma decisão, e baseadas em instância, que consideram apenas os valores

Figura 3 – Classificação das Estratégias de Manipulação de Conflitos.



Fonte: Adaptado de Bleiholder e Naumann (2006)

dos dados conflitantes para tomar a decisão. Como exemplo de estratégias de evitar conflitos baseadas em metadado podemos citar a *Trust your Friends Strategy*, que classifica como corretos os valores de uma fonte preferida, apontada pelo usuário ou encontrada automaticamente, utilizando critérios de qualidade. Um exemplo de estratégia de evitar conflitos baseada em instância é a *Take the Information*, que utiliza as informações não nulas e desconsidera os valores nulos, sendo essa a maneira natural de lidar com conflitos do tipo incerteza.

Resolver conflitos - Diferente das estratégias anteriores, as estratégias de resolver conflitos consideram todos os dados e metadados antes de decidir como resolver um conflito. Essas estratégias também são subdivididas em estratégias de resolver conflitos baseadas em instância e baseadas em metadados, e podem ainda ser classificadas pelo resultado que elas são capazes de produzir: as estratégias de decisão escolhem um valor preferido entre os valores existentes, enquanto as estratégias de mediação podem produzir um valor novo que não existe entre os valores conflitantes, como a média de um conjunto de números conflitantes. Como exemplo de estratégia baseada em instância e de decisão, podemos citar a *Cry with the Wolves*, o qual segue o princípio de acatar a decisão da maioria, ou seja, o valor que ocorre com maior frequência é escolhido como verdade entre os dados conflitantes. Como exemplo de estratégia baseada em instância e de mediação, citamos a *Meet in the Middle*. Esta estratégia não prefere

Quadro 1 – Funções de Manipulação de Conflitos.

Função	Descrição
Count	Conta o número de valores distintos não-nulos
Min / Max	Retorna o valor de entrada mínimo / máximo
Sum / Avg / Median	Calcula a Soma, média, e mediana de todos os valores não-nulos
Random	Escolhe randomicamente um valor entre todos os valores não-nulos
Choose	Retorna o valor fornecido por uma fonte específica
Vote	Retorna o valor que aparece com maior frequência entre os valores presentes
Most recent	Retorna o valor mais recente. Atualidade é avaliada com base em outro atributo ou metadados
Most complete	Retorna o valor da fonte que contém o menor número de valores nulos no atributo em questão
Group	Retorna um conjunto de todos os valores conflitantes. Deixa a resolução de conflitos para o usuário
Coalesce	Retorna o primeiro valor não-nulo que aparece
Shortest / Longest	Escolhe o valor de tamanho mínimo / máximo de acordo com uma medida
Highest quality	Retorna o valor com maior qualidade de informação. Requer o uso de um modelo de qualidade
Fisrt / Last	Retorna o primeiro / último valor, mesmo que seja um valor nulo
Most general / Most specific	Utilizando uma taxonomia ou ontologia, retorna o valor mais geral ou mais específico
Variance / Stddev	Retorna a variância e o desvio padrão dos valores dos dados
Concat	Retorna os valores concatenados. Pode incluir anotações, como os nomes das fontes de dados
Most active	Retorna o valor mais acessado ou utilizado (utilizando estatísticas do SGBD)

Fonte: Adaptado de Bleiholder e Naumann (2006)

um valor ao invés de outro, mas sim tenta inventar um valor o mais próximo possível de todos os valores presentes. Para exemplificar as estratégias baseadas em metadados e de decisão, citamos a *Keep up to Date*, a qual utiliza o valor mais recente. Uma estratégia baseada em metadados e de mediação é utilizar metadados para implementar a estratégia *Meet in the Middle*.

2.1.4 Funções de Manipulação de Conflitos

As estratégias de manipulação de conflitos são implementadas utilizando funções de manipulação de conflitos. O Quadro 1 apresenta algumas das funções de resolução de conflitos mais citadas na literatura. Em Bleiholder e Naumann (2006), as principais funções de manipulação de conflitos existentes da literatura são discutidas detalhando algumas de suas propriedades, como tipo (se é função de coluna única ou múltiplas, dependendo de quantas colunas são utilizadas para decidir o valor resolvido), domínios de entrada aplicáveis (em que domínios as funções podem ser aplicadas, ex. *string*, numérico, categórico), se a função é de mediação ou decisão, entre outras.

Algumas estratégias de manipulação de conflitos têm uma equivalência direta com certas

funções de manipulação de conflitos, e podem ser facilmente implementadas aplicando-se essa função nos dados conflitantes. Como exemplo, a estratégia *Pass it On* pode ser executada utilizando a função *group*. A estratégia *Roll the Dice*, pode ser implementada utilizando a função *random*, a *Take the Information* utilizando a função *coalesce*, a *Meet in the Midle*, utilizando a função *average* e a *Trust your friends* com a função *choose*. As estratégias podem ser implementadas de diferentes maneiras por diversas funções, dependendo do objetivo da decisão sobre os dados conflitantes.

Entre as funções propostas para implementar as estratégias de manipulação de conflitos existentes, a **voto** se tornou a mais utilizada (estratégia *Cry with the wolves*), se tornando a estratégia básica de fusão de dados. Entre os valores conflitantes de diferentes fontes de dados, cada valor tem um voto, e o valor com maior quantidade de votos (ou seja, fornecido pelo maior número de fontes) é considerado como a verdade.

2.1.5 Operadores Relacionais

Operadores relacionais também foram propostos para realizar a fusão de diferentes fontes de dados. Os dados são integrados a partir de tabelas de origem (provavelmente oriundas de fontes distintas) em uma tabela integrada. Os operadores padrão são junção (*join*) e união (*union*). De acordo com (BLEIHOLDER; NAUMANN, 2008), as técnicas baseadas em junção combinam tuplas de várias tabelas enquanto avaliam predicados de algumas de suas colunas. As técnicas baseadas em união criam um esquema comum primeiro, e em seguida, juntam os diferentes conjuntos de tuplas das tabelas de origem.

Junção - A junção combina duas tuplas de duas relações se todos os atributos comuns (mesmo nome) coincidem nos valores de seus atributos. Juntar duas tabelas de fontes distintas pode aumentar o esquema das tabelas individuais pois adiciona atributos de diferentes fontes ao resultado final. O *outer join* estende o resultado da junção padrão, adicionando ao resultado as tuplas que aparecem em apenas uma das duas relações de origem. Os atributos ausentes da outra relação são preenchidos por valores nulos. O *full disjunction* combina duas ou mais relações, onde todas as tuplas são combinadas em uma única tupla.

União - Combina tuplas de duas relações compatíveis (mesmo número de atributos e do mesmo tipo de dados) e remove as tuplas que coincidem em todos os valores de seus atributos. *Outer union* é uma extensão do operador de união e combina duas relações não compatíveis.

O resultado inclui todos os atributos de entrada com exceção dos duplicados, e todas as tuplas de entrada, sem exceções. *Minimum union* faz um *outer join* mas remove as tuplas que estão englobadas em outras tuplas. (ex: tupla t_1 contêm mais valores nulos que a tupla t_2 e todos os valores não nulos das duas tuplas t_1 e t_2 são idênticos. Neste caso, t_1 seria removida). *Merge* é a união de dois *outer joins*, e o *prioritized merge* é utilizado para priorizar valores de uma fonte preferida.

2.2 GRANDES VOLUMES DE DADOS

Desde o surgimento da *Web*, gerar e compartilhar dados se tornaram processos simples, o que tem contribuído para um crescimento acelerado no volume de dados disponíveis em meio digital. Não apenas uma fonte de dados pode conter um enorme volume de dados (ex. Redes sociais), mas também o número de fontes de dados tem aumentado. Esse crescimento tem ocorrido de forma desorganizada, de tal modo que muitos dados contêm valores errôneos, ausentes e duplicados, o que causa dificuldade na utilização e recuperação dos dados (PAPOTTI; SANTORO, 2018)

Além dos desafios enfrentados pela Integração de Dados em cenários tradicionais, nessa era de grandes volumes de dados vários desafios são adicionados ao processo de integração. O termo Integração de Grandes Volumes de Dados surgiu da necessidade de integrar esse volume massivo de dados que está sendo disponibilizado em meio digital. Essa integração em muito se identifica com o processo de Integração de Dados tradicional, mas alguns desafios surgem junto com esse novo cenário.

Segundo Dong e Srivastava (2015), a Integração de Grandes Volumes de dados se difere de Integração de Dados tradicional em muitas dimensões: (i) o número de fontes de dados, mesmo que em um único domínio, tem crescido muito, (ii) muitas das fontes de dados são muito dinâmicas, e uma enorme quantidade de dados é continuamente disponibilizada, (iii) as fontes de dados são extremamente heterogêneas em sua estrutura, com considerável variedade mesmo para entidades similares, e (iv) as fontes de dados são de qualidade amplamente diferentes, com diferenças significativas na cobertura, precisão e atualidade dos dados fornecidos.

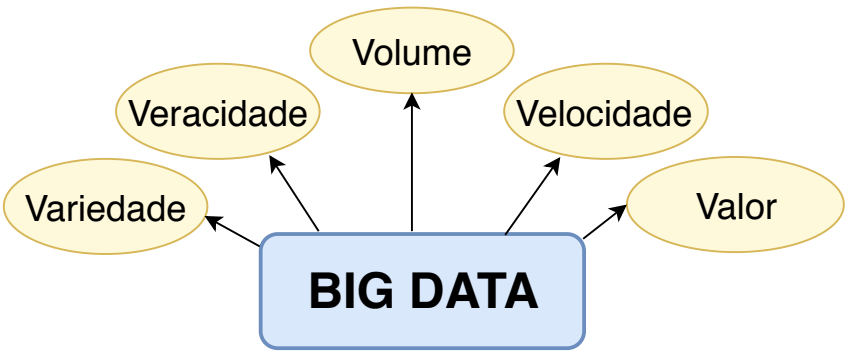
Os desafios atuais da Integração de Dados, que vão além dos cenários tradicionais, possuem uma ligação direta com as características dos cenários de grandes volumes de dados. Desde modo, iremos detalhá-las na próxima seção.

2.2.1 Características de Cenários de Grandes Volumes de Dados

O conceito de grandes volumes de dados, do inglês *Big Data* surgiu em meados dos anos 2000, com o intuito de definir o volume massivo de dados estruturados e não estruturados gerados a cada segundo juntamente com a tarefa de extrair valor desses dados.

Os principais aspectos do *Big Data* podem ser definidos por cinco V's, mostrados na Figura 4. Como o próprio nome, o volume é a mais marcante das características, e o que mais diferencia o tratamento e manipulação dos dados. Nesse ambiente, não somente as fontes de dados contêm um volume massivo de dados, como também, mesmo para um único domínio, o número de fontes de dados cresce exponencialmente. Existem muitos cenários onde uma única fonte de dados pode gerar altos volumes de dados. Típicos exemplos incluem sensores (como GPS), mídias sociais (como *Facebook*, *Twitter*), transações online (como compra e venda), entre outras. O volume de dados disponibilizados continua crescendo, e sistemas convencionais não estão preparados pra tratar uma quantidade tão grande de dados de forma eficiente, o que faz com que novas soluções sejam necessárias para abordar tal desafio.

Figura 4 – O cinco Vs do Big Data.



Fonte: Adaptado de Erl, Khattak e Buhler (2016)

A velocidade em que os dados são gerados também é uma característica que diferencia este cenário. O entendimento convencional de velocidade geralmente considera a rapidez com que os dados chegam e são armazenados, e quão rapidamente eles podem ser recuperados. Contudo, no contexto de grandes volumes de dados a velocidade deve também ser aplicada a dados "em movimento": fluxos constantes de dados em um ritmo que se torna impossível para os sistemas tradicionais conseguirem lidar. A rapidez com que os dados estão sendo produzidos e o quão rápido devem ser tratados para atender a demanda também classificam a característica de velocidade. Além disso, os dados que estão sendo produzidos têm uma vida útil muito curta, o que torna as fontes de dados altamente dinâmicas. Para encontrar *insights*

nesses dados é necessário realizar uma análise quase em tempo real. Deste modo, reagir rápido o suficiente para lidar com a velocidade é um grande desafio.

Quanto mais dados e fontes de dados maior a complexidade para lidar com eles. Os dados podem vir de uma variedade de fontes e em uma variedade de tipos, sendo a variedade outro V do *Big Data*. Com a criação de sensores e dispositivos inteligentes, bem como com as redes sociais, os dados tornaram-se complexos, pois dados semi-estruturados e não estruturados, como texto, arquivos de imagem, vídeo, áudio, dados de sensores, dados Extensible Markup Language (XML), dados JavaScript Object Notation (JSON), entre outros, passaram a ser administrados juntamente com os dados estruturados tradicionais (ZIKOPOULOS et al., 2012). Atualmente, a maior parcela de dados são de dados não-estruturados e semi-estruturados. Deste modo, conseguir lidar com diferentes formatos de dados é um desafio que precisa ser superado.

Além das características já citadas, existe ainda a veracidade. Esta característica se refere a qualidade dos dados e que tem tido sua importância cada vez mais reconhecida (BERTI-ÉQUILLE; BORGE-HOLTHOEFER, 2015). Grande parte dos dados pode conter erros ou serem inválidos, o que acarreta em uma necessidade de processamento. Além disso, as fontes de dados nesse cenário têm qualidades muito diferentes, e possuem diferença significativa na cobertura, precisão e atualidade dos dados fornecidos (DONG; SRIVASTAVA, 2015).

2.2.2 Principais diferenciais na Fusão de Dados em Cenários de Grandes Volumes de Dados

Integração de Dados sempre enfrentou diversos desafios ao longo de anos. Para lidar efetivamente com esses desafios, um progresso considerável foi alcançado nas últimas décadas pela comunidade de Integração de Dados nos tópicos fundamentais (etapas do processo) de alinhamento de esquema, resolução de entidades e fusão de dados, especialmente para dados bem estruturados (DONG; SRIVASTAVA, 2015). No entanto, quando se fala em Integração de grandes volumes de dados, os desafios surgem novamente em todas as etapas do processo.

As principais diferenças da Integração de Dados em cenários de grandes volumes para a Integração de Dados tradicional, são:

- As fontes de dados são extremamente heterogêneas em suas estruturas;
- As fontes de dados possuem qualidades muito distintas (acurácia, cobertura, e atuali-

dade, por exemplo);

- O número de fontes de dados é muito maior que em cenários tradicionais, mesmo que em um mesmo domínio;
- As fontes de dados são dinâmicas (dados são criados e atualizados o tempo todo).

A Fusão de Dados no cenário de integração de dados tradicional já foi amplamente estudada (BLEIHOLDER; NAUMANN, 2008; DONG; SRIVASTAVA, 2013; BERGAMASCHI et al., 2018). No entanto, o cenário de grandes volumes de dados traz consigo novos desafios, diretamente ligados às características destes cenários (e.g, volume, variedade, veracidade, e a velocidade que os dados são produzidos). Neste sentido, as técnicas de Fusão de Dados convencionais podem não ser eficientes, já que não consideram as características dos cenários de grandes volumes de dados. Além disso, há uma diferença expressiva entre a qualidade das fontes de dados o que ocasiona mais um desafio para a Fusão de Dados, a confiabilidade dos dados.

Para realizar a Fusão de Dados em cenário de grandes volumes de Dados são necessários novos métodos e sistemas para possibilitar abordagens eficientes. O principal desafio para Fusão de Dados é lidar com o grande número de dados conflitantes que foi disponibilizado na *Web*. Além disso, ao invés de lidar com um pequeno número de fontes de dados fornecendo dados sobre muitos objetos, neste cenário de grandes volumes de dados outro fenômeno ocorre: um grande número de fontes de dados que fornecem dados sobre apenas um pequeno número de objetos. Por exemplo, diversos sites e fontes fornecem informações sobre uma ou mais celebridades. No entanto, apenas poucas fontes fornecem dados sobre um grande número de celebridades (e.g., Wikipedia) (LI et al., 2014a). Este fenômeno tem sido nomeado como *Long-tail Phenomenon*.

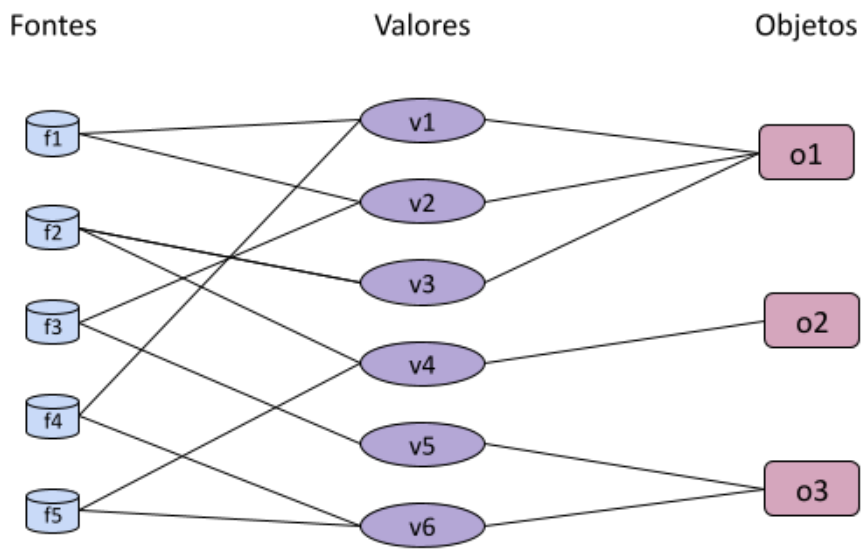
Para lidar com os desafios do *Big Data* é necessário aprimorar a Fusão de Dados, acrescentando novas capacidades ao processo. Quando se fala em veracidade, por exemplo, é necessário analisar a qualidade das fontes de dados participantes do processo por meio de métricas como acurácia, precisão, cobertura e atualidade; verificar se fontes copiam dados umas das outras; se existe correlação entre as fontes de dados, entre outros. Sobre o volume, soluções escaláveis devem ser implementadas, ou ainda soluções que paralelizam o processamento offline. Outra estratégia é realizar a fusão de dados sob demanda online (i.e., *pay-as-you-go approach*), apenas os dados necessários para responder o usuário são fundidos no momento, e ao longo do tempo a quantidade de dados integrados aumenta. Deste modo, a Fusão de Dados se torna um

processo mais completo e criterioso para lidar com um grande volume de dados de qualidades variadas.

2.3 SOLUÇÕES AVANÇADAS DE FUSÃO DE DADOS

Como foi dito na seção anterior, os primeiros métodos propostos para resolver os conflitos existentes nos dados, eram baseados em regras. No entanto, com o rápido crescimento e expansão das fontes na *Web*, a qualidade dos dados varia de uma fonte de dados para outra, ou seja, um vasto volume de dados conflitantes é disponibilizado neste meio. Isso faz com que a Fusão de Dados baseada em regras seja muitas vezes inadequada, já que existe uma necessidade de abordar a característica de veracidade dos dados.

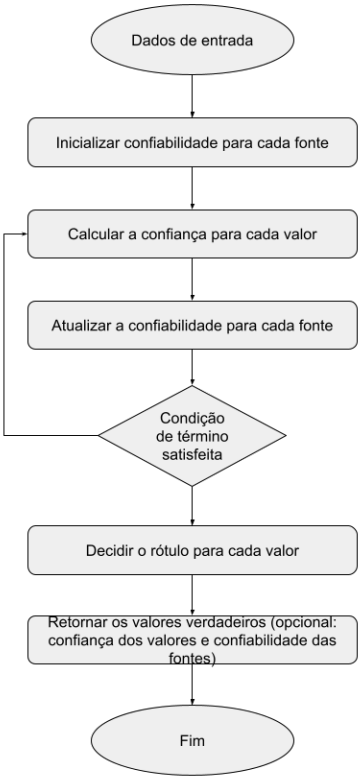
Figura 5 – Processo de Descoberta da Verdade.



Fonte: Adaptado de Yin, Han e Yu (2008)

A área de pesquisa responsável por investigar o problema de veracidade dos dados, é a descoberta da verdade, ou *Truth Discovery* (LI et al., 2015b; BERTI-ÉQUILLE, 2018). Descoberta da verdade tem recebido uma atenção crescente, motivada pela grande diversidade e complexidade de problemas relacionados à veracidade dos dados, mas também por suas aplicações diretas em vários contextos de Extração de Informação, Fusão de Dados, entre outros. Segundo Berti-Équille (2018), o princípio básico da descoberta da verdade é inferir iterativamente a confiabilidade das fontes de dados que fornecem os dados, e estimar a veracidade dos dados a partir da confiabilidade das fontes que os provê. A Figura 5 apresenta como se dá o contexto da descoberta da verdade. Várias fontes de dados proveem valores para um

Figura 6 – Algoritmo básico de descoberta da verdade.



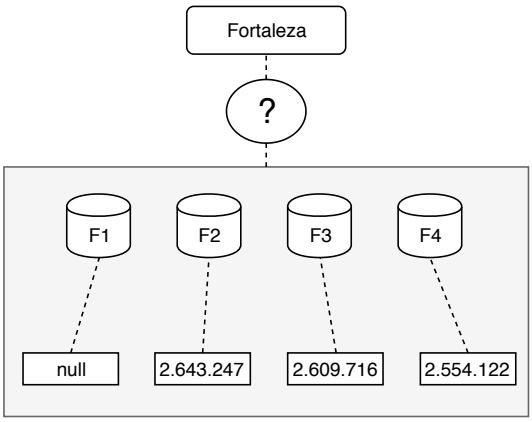
Fonte: Adaptado de Berti-Équille (2018)

dado atributo de diferentes objetos. Entre esses valores provavelmente para um mesmo objeto existem conflitos que são resolvidos pela descoberta da verdade.

O algoritmo central de um método de descoberta da verdade é um algoritmo de votação iterativo, e pode ser visto na Figura 6. Inicialmente, a confiabilidade de cada fonte é inicializada. Depois, para cada objeto e cada valor de atributo, o método calcula a confiança a partir da confiabilidade das fontes que os provê. Em seguida, atualiza a confiabilidade de cada fonte com base na confiança das informações providas por elas. Este procedimento se repete até que uma condição de parada seja atingida (como um número de iterações por exemplo). A saída desse algoritmo geralmente são os valores dados como verdadeiros, os valores de confiança, e os *scores* de qualidade das fontes. A principal diferença entre os métodos é a técnica empregada para realizar o processo de descoberta da verdade (*Bayesian Network*, *Restricted Boltzmann Machines*, *Neural Networks*, entre outras). Outra diferença está na maneira como eles realizam os cálculos de confiança dos valores e as atualizações, e quais métricas são consideradas ao avaliar a qualidade.

Para exemplificar a vantagem de utilizar características de qualidade das fontes, suponha o simples exemplo da Figura 7. Quatro fontes de dados distintas fornecem a informação do total

Figura 7 – Exemplo de descoberta da verdade.

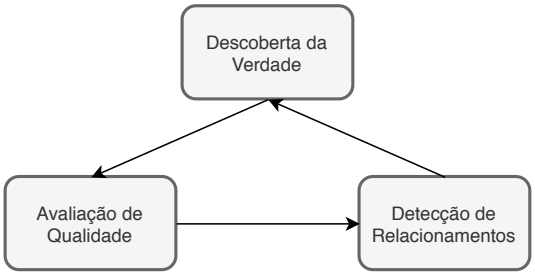


Fonte: Elaborada pela autora (2023).

populacional da cidade de Fortaleza. Neste exemplo, se o *voto majoritário* fosse aplicado nos dados da Figura 7, qualquer um dos valores não-nulos poderia ser dado como verdade, pois ocorre um empate. Cada valor possui 1 voto, e nesse caso a estratégia escolhe aleatoriamente. No entanto, considerando a qualidade da fonte como é feito pelos métodos atuais de descoberta da verdade, a informação dada como verdadeira será aquela oriunda da fonte cujo valor de qualidade é maior. Isso faz com que a probabilidade do método fornecer o valor correto seja mais alto.

No contexto de Fusão de Dados, muitas soluções têm sido propostas com o intuito de abordar o desafio da veracidade, principalmente voltadas a cenários de grandes volumes de dados. De acordo com Dong e Srivastava (2018) essas soluções geralmente contêm alguns ou todos os três componentes ilustrados na Figura 8 e detalhados a seguir.

Figura 8 – Soluções avançadas de Fusão de Dados.



Fonte: Adaptado de Dong e Srivastava (2015)

Descoberta da Verdade - Descobrir a verdade entre os valores conflitantes nos dados. O *voto majoritário* é a abordagem mais simples.

Avaliação de Qualidade - Avaliar a qualidade para cada fonte de dados participante do processo, de acordo com a corretude dos dados providos por elas.

Detecção de Relacionamentos - Detectar relacionamentos existentes entre as fontes de dados. Os relacionamentos podem ser de cópia entre um par de fontes (que pode ser utilizado para descontar votos, por exemplo), ou podem ser de correlação entre um subconjunto de fontes (fontes em um subconjunto podem ser consideradas como uma só na avaliação de qualidade).

2.4 CONSIDERAÇÕES

Neste capítulo, foram apresentados os principais conceitos com relação à Integração de Dados e Fusão de Dados. Foram apontados os tipos de conflitos que podem ocorrer nos dados e as etapas da Integração responsáveis por resolvê-los. Também foram elencadas as características que estão presentes nos cenários de grandes volumes de dados, os principais diferenciais destes cenários para cenários tradicionais, e os novos desafios que surgiram. Por fim, falamos brevemente sobre a arquitetura comumente utilizada pelas soluções de integração de dados avançadas, as quais lidam com cenários de grandes volumes de dados.

No próximo capítulo, será apresentado um levantamento do estado da arte na área de Fusão de Dados. São apresentadas algumas classificações para os métodos de Fusão de Dados, e uma nova classificação é proposta.

3 ESTADO DA ARTE

O problema de resolução de conflitos foi mencionado pela primeira vez na literatura na área de Integração de Bancos de Dados Relacionais, se referindo ao problema de um atributo de uma mesma entidade ter valores contraditórios em diferentes fontes de dados. Contudo, não foi dada muita importância ao problema, já que a maioria das estratégias propostas para lidar com dados conflitantes evitavam ou simplesmente ignoravam conflitos de dados (BLEIHOLDER; NAUMANN, 2008).

Resolução de conflitos de dados é um problema amplamente estudado no cenário tradicional de Integração de Dados, chamado de Fusão de Dados (MIRZA; SIDDIQI, 2016). Fusão de Dados como parte do processo de Integração de Dados tem sido estudada extensivamente ao longo de anos. Diversos sistemas de Integração e Fusão de Dados foram propostos na literatura (BILKE et al., 2005; FUXMAN; FAZLI; MILLER, 2005; MOTRO; ANOKHIN, 2006). Esses sistemas eram classificados de acordo com a estratégia que utilizavam para abordar os conflitos de dados. As estratégias de manipulação de conflitos podem ser classificadas em três classes: ignorar conflitos, evitar conflitos e resolver conflitos.

As estratégias para evitar e ignorar conflitos logo se tornaram insatisfatórias devido a limitações. Os conflitos não eram resolvidos, o que muitas vezes poderia gerar resultados inconsistentes. Desta forma, as estratégias de resolução de conflitos tornaram-se promissoras, transformando-se no foco principal das pesquisas na área, já que fornecem meios para decisões individuais de fusão para cada conflito de dado.

Os primeiros sistemas e métodos propostos para a Fusão de Dados eram geralmente baseados em regras. Funções como o valor médio, valor máximo, valor mínimo, valor mais recente ou a votação eram aplicadas para resolver conflitos. No decorrer dos anos, as abordagens de Fusão de Dados passaram a utilizar diferentes características dos dados e das fontes para ajudar na Fusão de Dados, como qualidade das fontes (BROELEMANN; GOTTRON; KASNECI, 2017a; LI et al., 2016; XIAO et al., 2016; ZHAO et al., 2012), cópia entre fontes de dados (FANG et al., 2017; DONG; BERTI-ÉQUILLE; SRIVASTAVA, 2009a; DONG; BERTI-ÉQUILLE; SRIVASTAVA, 2009b), dificuldade de objetos (WANG et al., 2017; GALLAND et al., 2010), relações entre objetos (NAKHAEI; AHMADI, 2017; PASTERNACK; ROTH, 2010) e popularidade de objetos (FANG, 2017). Além disso, novas técnicas também foram empregadas para resolver o problema de Fusão de Dados, como Inferência Bayesiana (FANG, 2017), Redes Neurais (BROELEMANN; KASNECI, 2018),

Quadro 2 – Classificações para modelos de Fusão de Dados.

Trabalho	Classificação
Bleiholder e Naumann (2008)	▪ Baseados em Resolver de Conflitos ▪ Baseados em Evitar Conflitos ▪ Baseados em Ignorar Conflitos
Li et al. (2015a)	▪ Voting ▪ Web link-based ▪ IR-based ▪ Bayesian-based
Dong et al. (2014)	▪ Votação ▪ Baseados em Relações ▪ Baseados em Qualidade – Baseados em links/ligações da Web – Baseados em Recuperação da Informação – Bayesianos – Modelos Gráficos
Berti-Équille e Borge-Holthoefer (2015)	▪ Baseados em Acordo/Concordância ▪ Baseados em estimativa de Maximum a Posteriori ▪ Analíticos ▪ Baseados em Inferência Bayesiana
Li et al. (2015b)	▪ Iterativos ▪ Baseados em Otimização ▪ Baseados em Modelo gráfico probabilístico

Fonte: Elaborado pela autora (2023).

Máquina de Boltzmann Restritas (BROELEMANN; GOTTRON; KASNECI, 2017b), e técnicas de *Bootstrapping* (XIAO et al., 2016). Conforme novas características foram sendo exploradas e novas técnicas utilizadas no processo de Fusão de Dados, também surgiram novas classificações para os métodos.

Com base nas classificações existentes na literatura e nos trabalhos recentes, criamos uma classificação um pouco diferente. Um resumo das classificações mais utilizadas na literatura pode ser visto no Quadro 2. Dividimos os métodos de Fusão de Dados em quatro categorias: Baseado em Regras, Aprendizado de Máquina, Baseado em Otimização e Baseado em Probabilidade. A diferença principal para outras categorizações é a inclusão da categoria de Aprendizado de Máquina, que foi motivada devido ao crescente aumento nas propostas de métodos de Fusão de Dados que utilizam essa técnica. Outra diferença é a categoria Baseada em Probabilidade, que é mais geral pois aborda todos os trabalhos que utilizam probabilidade para realizar a Fusão de Dados independente da técnica utilizada (*Markov, Bayesian,*

Quadro 3 – Definições das categorias das classificações apresentadas por diferentes autores para métodos de Fusão de Dados.

Categorias	**Descrição**
Baseado em resolver conflitos	Utilizam uma variedade de estratégias que seguem diferentes implementações para resolver os conflitos dos dados.
Baseado em evitar conflitos	Manipulam os dados para evitar a ocorrência de conflitos.
Baseado em ignorar conflitos	Lidam com os dados conflitantes ignorando os conflitos existentes.
Votação/*Baseline*	É a estratégia padrão. Entre valores conflitantes, cada valor tem um voto de cada fonte de dados. O valor com a maior contagem de votos é escolhido como correto entre os valores conflitantes.
Baseados em links da Web/Baseados em ligações da Web	Métodos inspirados pela medição de autoridades de páginas da Web com base em links (por exemplo, utilizando PageRank, algoritmo de avaliação de relevância usado pelo Google para posicionar websites em resultados de pesquisas.)
Baseados em Recuperação da Informação	Mede a confiabilidade da fonte e a similaridade entre os valores fornecidos e os valores reais. São utilizadas métricas de similaridade amplamente aceitas na área de Recuperação de Informação, como Cosine similarity
Baseados em Inferência Bayesiana	Baseados em análise bayesiana. Confiam na modelagem bayesiana para calcular a precisão da fonte e o valor de confiança.
Baseado em modelo gráfico probabilístico	Aplica modelos gráficos probabilísticos para obter conjuntamente a confiabilidade da fonte e o valor de corretude
Impactados por cópia	O cálculo da contagem de votos desconta os votos de valores copiados de outras fontes
Baseado em acordo/concordância	São baseados na contagem do número de fontes que concordam/discordam com cada item de dados
Baseado em estimativa de Maximum a Posteriori (MAP)	Utilizam Expectation Maximization (EM) ou Gibbs sampling para calcular as variáveis latentes ótimas (ou seja, a verdade e a confiabilidade da fonte), com base nas observações disponíveis
Analíticos	Utilizam matrix diagonalization para resolver a descoberta da verdade, que é reformulada como um problema de otimização
Baseados em Relações	Também consideram as relações entre as fontes
Iterativos	Nesses métodos as etapas de cálculo e estimativa de confiabilidade da fonte são conduzidas iterativamente até a convergência
Baseados em otimização	Esses métodos consideram a descoberta da verdade como um problema de otimização para inferir a confiabilidade da fonte e as informações confiáveis, e atualizar as verdades e os pesos de confiabilidade das fontes de forma iterativa até a convergência. Assim, esses métodos são semelhantes aos métodos iterativos.
Baseados em aprendizado de máquina	Métodos que usam alguma técnica de Aprendizado de máquina, algoritmos projetados para reconhecer padrões e identificar correções nos dados, além de fazer previsões ou tomar decisões. A máquina aprende a partir dos dados disponíveis e ajusta seus modelos e parâmetros para otimizar o desempenho das tarefas
Baseados em probabilidade	Métodos que utilizam alguma técnica de probabilidade, teorias, métodos estatísticos para quantificar incerteza e variabilidade dos dados.

Fonte: Elaborado pela autora (2023).

Graphical-model).

No Quadro 3, as definições de cada categoria apresentada nas classificações listadas no Quadro 2 são definidas. Algumas categorias se repetem em mais de uma classificação com

nomes semelhantes, e podem ser vistas juntas.

Nas próximas seções apresentamos os métodos de Fusão de Dados de acordo com a categorização proposta neste trabalho. Embora cada método seja classificado em apenas uma categoria, alguns trabalhos podem abranger mais de uma. No entanto, os métodos foram classificados pela estratégia primária utilizada para realizar a Fusão de Dados.

3.1 ABORDAGENS BASEADAS EM REGRAS

Para resolver os conflitos existentes nos dados, os métodos baseados em regras foram os primeiros a serem propostos na literatura. Geralmente, esses métodos utilizam a média ou mediana (para dados numéricos), ou empregam uma votação majoritária (para dados categóricos) para predizer os valores verdadeiros. A vantagem dessas abordagens é que o resultado geralmente é mais passível de depuração e compreensão em geral.

A classificação dos métodos baseados em regras é feita de acordo com a estratégia de manipulação de conflitos que é utilizada. Como explicado detalhadamente no Capítulo 2, as estratégias de manipulação de conflitos se dividem em três classes principais: **evitar conflitos**, **ignorar conflitos** e **resolver conflitos**. Em resumo, as Estratégias de Ignorar Conflitos não tomam uma decisão sobre o que fazer com os dados conflitantes. As Estratégias de Evitar Conflitos reconhecem a existência de conflitos em geral, mas não resolvem os conflitos isoladamente, ou seja, os conflitos são tratados em conjunto aplicando uma decisão única igualmente a todos os dados como, por exemplo, escolher os dados originados de uma fonte "preferida", entre dados conflitantes. Ambas estratégias têm suas limitações e podem produzir resultados inconsistentes, já que não resolvem os conflitos em si de maneira isolada. Diferentemente, as Estratégias de Resolver Conflitos fornecem meios para decisões individuais de fusão para cada conflito de dado.

As estratégias de manipulação de conflitos são executadas utilizando funções de manipulação de conflitos. Várias estratégias de manipulação de conflitos foram propostas (FUXMAN; FAZLI; MILLER, 2005; MOTRO; ANOKHIN; ACAR, 2004; SCHALLEHN, 2004; BLEIHOLDER; NAUMANN, 2006). No Quadro 4, são apresentadas algumas das estratégias mais citadas na literatura, a classificação segundo a forma de manipular conflitos, possíveis funções que podem ser utilizadas para implementar cada estratégia, e uma breve descrição.

Quadro 4 – Estratégias de resolução de conflitos baseadas em regras.

Estratégia	Classificação	Funções	Descrição
Pass it on	Ignorar Conflitos	Group, Concat	Aloca os conflitos para o usuário ou aplicação realizarem a decisão
Take the information	Evitar Conflitos	Coalesce, Longest	Dá preferência a valores do que a valores nulos
Trust your friends	Evitar Conflitos	Choose, Choose depending, Highest quality, First, Most complete	Pega o valor de uma fonte preferida
Cry with the wolves	Resolver Conflitos	Vote	Pega o valor que ocorre com mais frequência
Roll the dice	Resolver Conflitos	Random	Seleciona um valor randômico
Meet in the middle	Resolver Conflitos	Average, Median, Most general	Pega um valor médio
Keep up to date	Resolver Conflitos	Most recent, First	Seleciona o valor mais recente

Fonte: adaptado de Bleiholder e Naumann (2006)

3.1.1 Sistemas de Integração de Dados

Diferentes Sistemas de Integração de Dados foram propostos por pesquisadores da área de banco de dados, com o intuito de fornecer uma visão unificada de várias fontes de dados heterogêneas. Esses sistemas obrigatoriamente possuem funcionalidades para fundir os dados e, portanto, lidar com dados conflitantes. No entanto, o problema de conflitos nos dados e como resolvê-los é mencionado apenas por alguns deles, como *Hippo*, *Ariadne* e *Hermes*. Um levantamento completo desses sistemas é feito em Bleiholder e Naumann (2008), comparando características como: estratégia de manipulação de conflitos, tipos de conflitos tratados, modelo de dados suportado, modelo de integração implementado, modelo de acesso aos dados, entre outras.

Dentre esses sistemas, alguns que se destacaram por suas habilidades de Integração de Dados foram:

- **Ignorar Conflitos**
  - Nimble (DRAPER; HALEVY; WELD, 2001) - sistema comercial de integração utilizado por várias grandes empresas para limpar e integrar dados de diferentes fontes heterogêneas. É um dos primeiros sistemas de integração que utiliza XML como modelo de dados. Por ser um sistema comercial, não são publicadas muitas informações sobre os detalhes de implementação. Deste modo, não há informação sobre como os conflitos são manipulados no sistema.

  - Pegasus (AHMED et al., 1991) - sistema multi-banco de dados que permite ao usuário acessar diferentes fontes de dados distribuídas e heterogêneas. Serve como referência para muitos dos sistemas que foram desenvolvidos a partir de então. Utiliza o modelo de dados orientado a objetos para representar os dados. O problema de ter representações duplicadas para um mesmo objeto e, portanto, conflitos nos dados, é reconhecido mas não resolvido. Se houver objetos duplicados eles existirão apenas na visão do administrador do sistema, que é responsável por resolver conflitos manualmente, já que o sistema não dá suporte para essa resolução.
  - Carnot (COLLET; HUHNS; SHEN, 1991) - sistema de integração que consiste em agentes de software independentes, cada um com sua própria área de conhecimento. Utiliza como modelo de dados global uma base de conhecimento. As consultas podem ser emitidas usando *Structured Query Language (SQL)* em um dos esquemas locais. A partir disso, os agentes são responsáveis por mapear as consultas aos conceitos da ontologia, e depois às próprias fontes de dados. Devido à estrutura modular, as consultas podem tanto ser executadas nos esquemas locais como no esquema global. Os mapeamentos são feitos à mão por especialistas e, ao combinar resultados, as duplicatas não são detectadas e os conflitos de dados são ignorados.

- **Evitar Conflitos**

  - Hippo (CHOMICKI; MARCINKOWSKI; STAWORKO, 2004a; CHOMICKI; MARCINKOWSKI; STAWORKO, 2004b) - sistema segue o paradigma *Consistent Query Answering (SQA)*, devolvendo aos usuários apenas respostas consistentes, sem existência de conflitos nos dados.
  - SIMS e Ariadne (S*ervices and Information Management for decision Systems*) (ARENS; KNOBLOCK; SHEN, 1996; KNOBLOCK et al., 2001) - o sistema SIMS fornece acesso inteligente a fontes de dados distribuídas e heterogêneas (banco de dados, bases de conhecimento, arquivo simples), retirando do usuário a necessidade de saber a localização das fontes, linguagens de consulta, entre outros. A abordagem padrão do sistema é utilizar um esquema global que relacione as informações entre as diferentes fontes locais, onde o usuário realiza consultas sob esse esquema global. O sistema consiste de duas partes: 1) planejador/executor de consultas que determina como a consulta irá ser processada com eficiência; 2) *wrappers* que fazem a

mediação entre o sistema e as fontes de dados para que possam ser consultadas como se fossem bancos de dados SQL. Um *wrapper* pega a consulta formulada na linguagem do SIMS, traduz na linguagem de consulta apropriada para a fonte de dados, e envia os resultados da consulta de volta para o SIMS. SIMS fornece um abordagem alternativa, em que para cada tipo de aplicação um modelo de domínio é criado. Nesse modelo, o domínio é representado utilizando linguagem de representação de conhecimento, descrevendo objetos, atributos e seus relacionamentos. Ariadne estende o sistema SIMS para fornecer acesso também a fontes de dados *Web*. Em ambos não é descrito como os conflitos de dados são tratados. Apenas para IDs de objetos, no Ariadne é utilizada a estratégia *Trust your friends*.

- ConQuer (FUXMAN; FAZLI; MILLER, 2005) - utiliza uma técnica de reescrita de consultas para obter apenas respostas consistentes. A reescrita funciona com qualquer Sistema Gerenciador de Banco de Dados (SGBD) relacional e não precisa de nenhum processamento adicional. Os conflitos de dados são evitados retornando apenas respostas consistentes (i.e., Estratégia *No Gossiping*).

- **Resolver Conflitos**

  - HumMer (BILKE et al., 2005) - sistema permite a integração semiautomática de várias fontes de dados remotas e heterogêneas. O sistema considera toda a variedade de conflitos (de esquema, de instância e de dados). A Fusão de Dados é realizada em agrupamentos criados pela etapa de resolução de entidades, na qual cada agrupamento contém representações do mesmo objeto. Na fusão, são aplicadas funções de agregação padrão e avançadas, como mínimo, máximo, voto, para cada agrupamento.

  - FusionPlex (MOTRO; ANOKHIN, 2006) - sistema capaz de integrar informações de diferentes fontes de dados heterogêneas, e possui recursos de reconhecimento de conflitos e resolução, com base em informações de qualidade das fontes de dados. O sistema agrupa tuplas que correspondem ao mesmo objeto de acordo com uma chave global e, em seguida, utiliza metadados de qualidade (atualidade, acurácia, disponibilidade, entre outros), para escolher somente tuplas de alta qualidade a serem consideradas na Fusão de Dados. Em seguida, são aplicadas funções de manipulação de conflitos (mínimo, máximo, média) para alcançar uma representação única para cada objeto.

– Hermes (*HEterogeneous Reasoning and MEdiator System*) (SUBRAHMANIAN et al., 1995) - o sistema permite a combinação de diferentes fontes por meio de um mediador. São abordados conflitos esquemáticos e de dados. Para resolver conflitos esquemáticos, os mesmos conceitos em fontes distintas são mapeados para um conceito global comum (abordagem *Global-as-view (GAV)*). Ambos os tipos de conflitos são reconhecidos e resolvidos utilizando informações de confiabilidade e registro de data e hora, por exemplo (estratégias *Trust your friends* e *Keep up to date*). O usuário pode acessar o sistema utilizando diferentes linguagens de consulta, como texto livre, utilizando uma linguagem de consulta gráfica, ou uma linguagem baseada em lógica.

3.1.2 Outros trabalhos

Além de Sistemas de Integração de Dados, outros trabalhos também foram propostos para Fusão de Dados baseada em regras. Em Cecchin, Ciferri e Hara (2010) é proposto um modelo para Fusão de Dados XML. A motivação do trabalho é que como os sistemas de integração e limpeza de dados geralmente precisam de um esforço manual em algum momento do processo, esse trabalho visa minimizar esse esforço, automatizando o máximo possível de tarefas. O modelo é baseado em definir manualmente uma política composta de um conjunto de regras. Depois, quando são identificados conflitos nos processos, eles são tratados sem intervenção do usuário. Uma ferramenta é resultado do trabalho, chamada *XFusion*, criada com base no modelo proposto, a qual oferece suporte à integração de dados XML e processos de limpeza.

Em Hara, Ciferri e Ciferri (2013), foi proposta uma extensão ao modelo de fusão de dados proposto em Cecchin, Ciferri e Hara (2010). Este novo modelo suporta atualizações incrementais baseadas em informações de proveniência. Para isso, é mantido um repositório de operações que reflete as decisões do usuário e a proveniência dos dados, juntamente com uma base de regras. Quando o usuário decide aplicar uma dada estratégia a um conflito, isso é mapeado para uma sequência de operações básicas armazenadas no repositório de operações. O repositório de operações, junto com a base de regras, permite atualizações incrementais no banco de dados local e minimiza a quantidade de intervenção do usuário em futuros processos de fusão.

O sistema é baseado em três módulos: fusão, validação e atualização. Os dados de cada banco de dados local são carregados no banco de dados separadamente pelo módulo de atu-

alização. O módulo de atualização é responsável por gerar um documento que representa os conflitos de elementos (de uma nova fonte ou de atualizações de uma fonte já existente). Este documento é a entrada para o processo de fusão. No módulo de fusão, um banco de dados mediador é gerado, resultante da fusão de dados oriundos de várias fontes. Além do documento gerado pelo módulo anterior, também é entrada da fusão um conjunto de regras definidas pelo usuário. Ao realizar a fusão, as regras são armazenadas no repositório de regras, e as operações de fusão são armazenadas no repositório de operações.

No módulo de validação é determinado se a execução das operações no repositório terá o mesmo efeito se executadas em novas versões das fontes. Quando uma fonte não apresenta atualizações, todas as operações são válidas e não há necessidade de reexecução. No entanto, se alguma atualização ocorre nas fontes, podem existir operações inválidas no repositório, o que exige que o objeto passe por um novo processo de fusão. Este módulo também tem como objetivo detectar remoções ou inserções realizadas nas fontes. Deste modo, ao filtrar os objetos que permaneceram inalterados em novas versões e reaplicar regras de fusão definidas anteriormente, é possível minimizar a necessidade de intervenção manual do usuário em futuros processos de fusão.

3.2 ABORDAGENS PROBABILÍSTICAS

Os trabalhos que se enquadram nesta categoria utilizam modelos probabilísticos para calcular conjuntamente a confiabilidade da fonte e a corretude dos valores. Diversos trabalhos foram propostos utilizando essa abordagem, dentre eles, alguns tiveram mais destaque, e serão resumidamente apresentados a seguir.

Truthfinder (YIN; HAN; YU, 2008) - O primeiro trabalho a introduzir formalmente o problema de veracidade na Fusão de Dados foi Yin, Han e Yu (2008). O problema da veracidade foi formulado da seguinte maneira: dada uma grande quantidade de informações conflitantes, sobre diversos objetos, fornecidas por diferentes fontes (*websites*), como descobrir os valores verdadeiros dos atributos de cada objeto. Para isso, é proposto um algoritmo iterativo nomeado *Truthfinder*, cuja principal ideia é que as informações provavelmente serão verdadeiras se forem fornecidas por fontes confiáveis, e uma fonte é confiável se a maioria das informações fornecidas por ela forem verdadeiras. A entrada do algoritmo são valores sobre atributos de um dado tipo de objetos. Os valores são fornecidos por diversos *websites*. Geralmente, existem

muitos valores conflitantes para cada objeto, e o objetivo do algoritmo é identificar os valores verdadeiros entre os conflitos de dados.

Dong, Berti-Équille e Srivastava (2009a) - É apresentado um estudo de como melhorar a descoberta da verdade utilizando detecção de dependência entre fontes e análise da precisão das fontes. A técnica proposta considera não apenas se duas fontes compartilham os mesmos valores para um dado objeto, mas também se os valores compartilhados são verdadeiros ou falsos. Compartilhar os mesmos valores verdadeiros não significa necessariamente que as fontes sejam dependentes. No entanto, se os mesmos valores falsos são compartilhados, geralmente as fontes são dependentes. Para avaliar a dependência, são apresentados dois tipos de fontes de dados: fontes independentes, que fornecem os valores de forma independente, e fonte copiadora, que copia parte (ou todos) dos dados de outras fontes. Para computar a probabilidade de duas fontes serem dependentes foi utilizado o modelo Bayesiano. Existe uma interdependência entre descobrir os valores verdadeiros utilizando a dependência entre fontes, e descobrir a dependência entre as fontes utilizando os valores verdadeiros. O método proposto faz isso de forma iterativa.

Dong, Berti-Équille e Srivastava (2009b) - Nesse trabalho os autores propõem avaliar se existe relação de cópia entre fontes de dados dinâmicas (onde as informações mudam o tempo todo). Primeiramente, os autores utilizam *Hidden Markov Model*, que considera o histórico de atualização das fontes, ou seja, se duas fontes compartilham um histórico de atualizações similar, é mais provável que exista uma relação de cópia. Além disso, para avaliar a qualidade das fontes, são propostas várias medidas de qualidade, como cobertura, exatidão e atualidade. Por fim, é proposto um modelo Bayesiano para decidir o *lifespan* (quando o valor real de um atributo é alterado e qual é o novo valor) de cada objeto. Esse modelo considera a qualidade da fonte e a existência de cópia, portanto é menos afetado por atualizações erradas, dados obsoletos ou dados copiados.

Cosine, 2-Estimates e 3-Estimates (GALLAND et al., 2010) - Nesse trabalho são apresentados três algoritmos para identificar os valores verdadeiros relatados por um conjunto de visões, bem como a estimativa de qualidade das fontes. Primeiro, é introduzido um modelo probabilístico de dados que leva em consideração a incerteza associada aos valores relatados e a cobertura das fontes. Todos os algoritmos calculam os valores de verdade e qualidade iterativamente. *Cosine* é baseado na medida de similaridade *cosine*, muito popular em Recuperação da Informação. *2-Estimates* faz o mesmo, mas quando a fonte fornece um valor, é conside-

rado que a fonte vota contra os outros valores. *3-Estimates* refina o *2-Estimates*, considerando também a confiabilidade em cada valor, ou seja, a probabilidade deste valor estar correto. Este valor é calculado iterativamente junto com a confiabilidade da fonte e a contagem dos votos.

SOLARIS (LIU et al., 2011) - É apresentado o SOLARIS, *framework* para Fusão de Dados online. O sistema ao invés de retornar todas as respostas de uma vez, começa retornando as respostas da primeira fonte de dados, e em seguida atualiza a saída à medida que detecta mais fontes. Para cada resposta retornada, é apresentada a probabilidade da resposta ser correta com base nos dados recuperados e na qualidade da fonte de dados. Quando o sistema ganha confiança suficiente de que os dados que não foram processados dificilmente alterariam as respostas, eles terminam, sem necessariamente investigar todas as fontes de dados. Por este motivo, o SOLARIS pode reduzir o tempo de processamento das consultas. O SOLARIS requer conhecimento prévio da precisão das fontes e da existência de fontes dependentes.

LTM (ZHAO et al., 2012) - Os autores propõem um modelo gráfico probabilístico (***L**atent **T**ruth **M**odel*) que pode inferir automaticamente os valores verdadeiros e a qualidade das fontes. Este trabalho se diferencia das abordagens anteriores pois é o primeiro trabalho proposto para avaliar os valores verdadeiros para atributos com vários valores (por exemplo, os autores de um livro). Os dados de entrada estão no formato de triplas (entidade, atributo, fonte). Além disso, LTM permite que conhecimentos prévios sobre a distribuição das verdades ou a qualidade das fontes sejam incorporados ao método.

LCA (PASTERNACK; ROTH, 2013) - A abordagem LCA (***L**atent **C**redibility **A**nalysis*) é proposta para avaliar a credibilidade das fontes e descobrir as verdades utilizando modelos probabilísticos e algoritmo *EM* para calcular a probabilidade de um valor ser verdadeiro. A verdade é modelada como uma variável latente e a credibilidade da fonte é capturada por um conjunto de parâmetros. Foram propostas quatro variações do LCA: *SimpleLCA*, *GuessLCA*, *MistakeLCA* e *LieLCA*. A diferença entre as variações é na maneira que a confiabilidade da fonte é considerada. Dentre as variações o que mais se destaca é o *GuessLCA* devido ao seu desempenho consistente e à escalabilidade linear de acordo com o tamanho do problema.

Pochampally et al. (2014) - Nesse trabalho, são propostas técnicas de descoberta da verdade utilizando correlação entre fontes de dados no processo. O principal destaque é que nesse trabalho a correlação de fontes não é avaliada apenas com base em cópia, mas também em outros sentidos, como fontes que fornecem dados de domínios complementares (correlação negativa), extratores que podem se concentrar em diferentes tipos de informação (correlação

negativa), ou ainda extratores que aplicam regras em comum na extração (correlação positiva, sem cópia). Esses critérios de correlação são modelados e aplicados no processo de descoberta da verdade.

IATD (ZHANG et al., 2016) - Nesse trabalho é proposto um método probabilístico não supervisionado baseado no modelo Bayesiano. Os autores acreditam que as alegações feitas por uma fonte podem ser influenciadas por outras. Para modelar as influências entre as fontes, o IATD (***I****nfluence-****a****ware* ***T****ruth* ***D****iscovery*) introduz o conceito de "confiabilidade da reivindicação", que junta a confiabilidade da fonte, que faz a reivindicação, e a confiabilidade de seus influenciadores. Considerando as correlações da fonte como um conhecimento prévio para derivação de influência, a confiabilidade de uma fonte pode ser estimada com mais precisão. O IATD é dividido em dois estágios: no primeiro é especificada a confiabilidade individual das fontes, bem como a confiabilidade *influence-aware*, dado o influenciador definido para cada alegação. Então, a segunda etapa visa gerar afirmações heterogêneas, dada a "confiabilidade da reivindicação" de cada afirmação. O modelo pode manipular tipos numéricos e categóricos de dados.

SmartMTD (FANG et al., 2017; FANG, 2017) - Essa abordagem baseada em grafos utiliza modelos de Cadeia de Markov com inferência bayesiana para descobrir a verdade de objetos com múltiplos valores. SmartMTD incorpora dois conceitos: relações entre fontes e popularidade de objetos, com o intuito de melhorar o processo de descoberta da verdade. A relação entre fontes é modelada em duas medidas: relações de suporte e relações de cópia. São construídos *supportive graphs*, que modelam a relação de "apoio"entre as fontes em suas afirmações positivas e negativas, derivando a confiabilidade em dois aspectos: precisão positiva e negativa das fontes. A relação de cópia entre fontes é capturada com *malicious agreement graphs*, considerando que as fontes que compartilham os mesmos valores falsos têm maior probabilidade de serem dependentes. A popularidade do objeto é considerada também no cálculo de confiabilidade da fonte. É proposta uma técnica para quantificar a popularidade de objetos com base na ocorrência e na cobertura das fontes.

Hybrid (LI et al., 2017) - Nesse trabalho os autores propõem um modelo chamado *Hybrid* que tem como proposta descobrir os valores verdadeiros para objetos com múltiplos valores. *Hybrid* toma duas decisões: quantos valores o objeto possui e quais são as verdades. O modelo também funciona para atributos de valor único, já que descobre a quantidade de valores que o atributo possui. Sob uma sequência de valores verdadeiros identificados anteriormente, com

base no modelo bayesiano, é calculada a probabilidade de um valor ser a próxima verdade, e a probabilidade de não existir mais valores verdadeiros.

FTS (ZHANG et al., 2018) - Nesse trabalho o problema de descoberta da verdade é modelado como um modelo gráfico probabilístico. Os autores propõem o uso de três indexes para medir a qualidade das fontes: ***F****alse rate*, ***T****rue rate*, e ***S****ilent rate*. Em comparação com o estado da arte que não considera como a qualidade da fonte é afetada quando a mesma fornece valores nulos, esse modelo faz uso total de todas as informações providas pelas fontes (verdadeiras e falsas) e dos valores nulos para melhorar a precisão da descoberta de verdade. Para a tarefa de descoberta da verdade, os autores fazem uso do método *Hub Authority*, inicialmente proposto por Kleinberg (1999), com diversas variações propostas ao longo do tempo. O index *silent rate* é o principal diferencial desse trabalho, pois engloba a utilização de dados nulos fornecidos pelas fontes para medir o valor de qualidade.

(JARADAT et al., 2022) - Esse trabalho propõe uma abordagem de fusão de dados probabilística que lida com atributos de valores únicos e multi valorados, extraindo os valores verdadeiros para dados incertos e conflitantes. Os autores apontam que a abordagem foi desenvolvida para lidar com um ambiente de fusão de dados dinâmico e sob demanda. A abordagem foi implementada dentro de um sistema de integração de dados e testada em diferentes cenários de dados. Os resultados foram satisfatórios, mostrando que a abordagem lida de forma eficiente com cenários de dados dinâmicos. Como limitação do trabalho, os autores citaram que essa abordagem desconsidera a correlação entre fontes de dados.

3.3 ABORDAGENS BASEADAS EM OTIMIZAÇÃO

SSTF (YIN; TAN, 2011) - Nesse trabalho, é proposta uma abordagem para a descoberta da verdade de forma semi-supervisionada. A abordagem é classificada como semi-supervisionada, pois uma grande quantidade de fatos não rotulados também participa do processo de aprendizagem. SSTF (***S****emi-****S****upervised* ***T****ruth* ***D****iscovery)* tem como objetivo atribuir um *score* de confiança, que varia entre -1 e 1, a cada valor entre os valores conflitantes, de modo que os valores verdadeiros obtenham pontuação mais alta que os valores falsos. Para isso, o modelo utiliza um pequeno conjunto de dados rotulados como verdadeiros (*ground truth*) para ajudar a distinguir valores verdadeiros dos falsos, assim como identificar fontes de dados confiáveis. O processo é realizado iterativamente.

CATD (LI et al., 2014a) - Esse trabalho propõe um método nomeado CATD (***C**onfidence-**A**ware **T**ruth **D**iscovery*) para resolver os conflitos de dados e encontrar os valores verdadeiros entre informações providas por múltiplas fontes. O método proposto pode lidar com o desafio trazido pelo *Long-tail phenomenon* (i.e., a maioria das fontes fornece poucas informações sobre um ou dois objetos, e apenas algumas fontes cobrem uma maior quantidade de informações). Para avaliar a confiabilidade da fonte, os autores acreditam que apenas fazer uma avaliação pontual não é suficiente para cenários que contenham muitas fontes que fornecem poucas informações, já que a eficácia desta estimativa é fortemente afetada pelo número total de informações providas por cada fonte. Por isso, é proposto um estimador baseado no intervalo de confiança da confiabilidade da fonte. Esse estimador pode avaliar com sucesso a confiabilidade da fonte e descontar o efeito de fontes que fornecem poucas informações. A confiabilidade é avaliada com base na variância da distribuição de erros (i.e., diferenças entre os valores fornecidos pela fonte e as verdades), que reflete o grau de confiabilidade dessa fonte: se uma fonte não é confiável, os erros cometidos ocorrem com frequência, portanto a variação da distribuição de erro é grande.

Li et al. (2015c) - Esse trabalho propõe resolver o problema de descoberta da verdade em cenários de dados dinâmicos em que as informações são coletadas continuamente. Nesses cenários existem alguns desafios, como: i) a informação verdadeira dos objetos evolui ao longo do tempo, ou seja, a informação verdadeira pode ser alterada em diferentes instantes de tempo; ii) a confiabilidade das fontes muda ao longo do tempo. Para abordar esses desafios, o método proposto atualiza a confiança dos valores, e a confiabilidade das fontes de maneira incremental. A solução proposta é modelada como um problema de otimização e utiliza *MAP estimation* para calcular os pesos das fontes. Para captar as relações temporais entre a informação de confiança identificada e a confiabilidade da fonte, são incorporados dois fatores: fator de suavização e fator de decaimento no método proposto. O fator de suavização é usado para lidar com os valores verdadeiros em evolução, ou seja, que podem ser alterados ao longo do tempo, e o fator de decaimento é usado para lidar com a confiabilidade das fontes nesses cenários dinâmicos.

CRH (LI et al., 2014b; LI et al., 2016) - O *framework* CRH (***C**onflict **R**esolution on **H**eterogeneous Data*) foi proposto para inferir as verdades a partir de múltiplos dados conflitantes de variados tipos, oriundos de fontes de dados distintas. Nesse trabalho, os autores propõem tratar o problema de heterogeneidade dos dados, disponibilizando diferentes tipos de

funções (i.e. função que mede a distância entre o valor verdadeiro e o valor fornecido) utilizadas para capturar as características de diferentes tipos de dados. O problema de resolução de conflitos é modelado como um problema de otimização, e é proposto um procedimento iterativo em duas etapas: descoberta da verdade e avaliação de confiabilidade da fonte.

DTD (WANG et al., 2017) - O DTD (***D**istributed **T**ruth **D**iscovery Framework*) é proposto para um novo cenário de descoberta da verdade, ou seja, descoberta da verdade distribuída. Em suma, as informações sobre os objetos, fornecidas por fontes diferentes, são geralmente distribuídas em vários servidores locais. Sob essa configuração distribuída, os métodos tradicionais de descoberta de verdade não podem ser aplicados diretamente, pois geralmente exigem todas as informações reunidas em um servidor central. O DTD incorpora uma estimativa de incerteza, inferida a partir das informações fornecidas para cada objeto. O *framework* consiste de dois componentes: cálculo da verdade local, calcula as verdades locais dos objetos em cada servidor local; e cálculo da verdade central, infere as verdades finais no servidor central com as saídas de todos os servidores locais. Uma abordagem chamada UbTD (*Uncertainty-based Batch Truth Discovery*) é proposta na etapa de avaliação da verdade local, para modelar as diferenças entre objetos como os valores de incerteza usados para estimar as verdades em servidores locais. Diferenças entre objetos, nesse trabalho, são baseadas em dois fatores principais: i) os fatores internos, ou seja, as características dos próprios objetos, como o nível de dificuldade; ii)) os fatores externos, como o número de informações sobre cada objeto e a qualidade das fontes que as proveem. Na prática, as diferenças entre os objetos são inevitáveis, especialmente nos ambientes distribuídos, e podem influenciar diretamente na avaliação da verdade. A etapa central de estimativa da verdade tem como objetivo inferir as verdades finais dos objetos considerando a qualidade dos servidores locais e as verdades estimadas.

3.4 ABORDAGENS BASEADAS EM APRENDIZADO DE MÁQUINA

ETCIBoot (XIAO et al., 2016) - ***E**stimating **T**ruth and **C**onfidence **I**nterval via **Boot**strapping* é proposto para descobrir automaticamente intervalos de confiança em tarefas de descoberta da verdade. A maioria dos métodos de descoberta da verdade se concentra em fornecer um valor pontual para a verdade de cada objeto, mas em muitas aplicações do mundo real, um intervalo de confiança na estimativa da verdade é mais desejável. Um intervalo de confiança estimado da verdade pode beneficiar qualquer cenário de descoberta de verdade, fornecendo

informações adicionais na saída. Para isso, é proposto um procedimento iterativo de duas etapas: i) atualizar a confiança dos valores com base no valor de confiabilidade da fonte; e ii) atualizar os pesos de confiabilidade das fontes com base nos valores verdadeiros. Na primeira etapa, ao invés de dar um valor pontual, é adotado o procedimento para obter intervalos de confiança. A maioria dos métodos de descoberta da verdade existentes aplica média ponderada ou votação usando as informações de todas as fontes. Em contraste, o ETCIBoot primeiro faz o *bootstrap* de múltiplos conjuntos de fontes e então em cada conjunto de fontes obtém uma estimativa de verdade. O estimador de verdade final é definido como a média dessas estimativas.

SLiMFast (REKATSINAS et al., 2017) - Esse trabalho propõe um *framework* que expressa a Fusão de Dados como um problema de aprendizado estatístico utilizando modelos probabilísticos discriminativos. A tarefa é composta por duas etapas: i) realizar o aprendizado estatístico para calcular os parâmetros do modelo, usado para estimar a precisão das fontes de dados; e ii) realizar inferências probabilísticas para prever os valores reais dos objetos. O SLiMFast é a primeira abordagem de Fusão de Dados a combinar conflitos entre fontes de dados com recursos específicos do domínio como informação adicional para estimar a precisão das fontes. O usuário tem a possibilidade de especificar um conjunto de características que são consideradas informativas para estimar a confiabilidade das fontes.

LTD-RBM (BROELEMANN; GOTTRON; KASNECI, 2017a; BROELEMANN; GOTTRON; KASNECI, 2017b) - ***L**atent **T**ruth **D**iscovery based on **R**estricted **B**oltzmann **M**achines* é uma abordagem de descoberta da verdade baseada em redes neurais. Esse algoritmo utiliza *Restricted Boltzmann Machines (RBM's)* para realizar a inferência das verdades e confiabilidade das fontes. RBM's são redes de duas camadas, uma escondida e uma visível, e unidades binárias. Podem ser utilizadas para aprendizado não supervisionado. Nesse caso, a motivação para usar RBM's para a descoberta da verdade é sua capacidade de aprender fatores desconhecidos, o principal fator oculto é o valor verdadeiro desconhecido, que precisa ser descoberto pela tarefa de descoberta da verdade. Para realizar o cálculo de probabilidades, as RBM's precisam aprender sobre um conjunto de exemplos de treinamento, que pode ser feito de forma não supervisionada. Nesse trabalho é utilizado o método de aprendizagem *Contrastive Divergence*.

GRBM (BROELEMANN; KASNECI, 2018) - Esse trabalho propõe uma extensão de Broelemann, Gottron e Kasneci (2017a), Broelemann, Gottron e Kasneci (2017b), e diferente da maioria dos métodos existentes na literatura incorporam características arbitrárias para resol-

ver o problema de descoberta de verdade. Propõe-se uma extensão do método LTD-RBM adicionando vetores de características. As características dependem das fontes de dados, das informações fornecidas, ou de ambas. Exemplos de características são: a área de especialidade de um fonte de dados, o tópico de uma informação fornecida por uma fonte, a dificuldade dessa informação, e a certeza de uma fonte sobre a informação fornecida. Utilizar vetores de características modifica a função do cálculo de confiabilidade das fontes. Esta abordagem é baseada no treinamento não supervisionado de redes *feed-forward*, utilizando *Contrastive Divergence*, enquanto o pré-treinamento é feito de forma supervisionada.

3.5 TRABALHOS QUE EXPLORAM RELACIONAMENTOS

Com o intuito de aprimorar a eficiência dos processos de Fusão de Dados, diversos autores têm proposto métodos que exploram diferentes tipos de relacionamentos. Entre esses relacionamentos, destacam-se os aqueles entre fontes, entre objetos, ou ainda relacionamentos entre atributos. No presente trabalho, estamos interessados principalmente em relacionamentos entre atributos. No entanto, serão apresentados alguns estudos relevantes que exploram relacionamentos entre objetos.

Pasternack e Roth (2010) - Nesse trabalho, a principal motivação dos autores é que os algoritmos de descoberta da verdade muitas vezes são incapazes de considerar o conhecimento prévio do usuário no processo. Por isto, os autores propõem um *framework* para incorporar conhecimento prévio em qualquer algoritmo de descoberta da verdade, expressando tanto o "senso comum", quanto fatos específicos já conhecidos pelo usuário. O conhecimento dado em lógica de primeira ordem é traduzido em um programa linear. É feita a suposição de que para cada atributo de entidade existe apenas um valor verdadeiro (*single-truth)*. Também são propostos quatro novos algoritmos de descoberta da verdade: *Sums*, *Average-Log*, *Investment* e *PooledInvestment*.

- *Sums* - Adaptação do *Hubs and Authorities* (KLEINBERG, 1999) para o problema de descoberta da verdade. As fontes são vistas como *hubs* e os valores providos pelas fontes são como *authorities*.
- *Average-Log* - Avalia a confiabilidade da fonte como uma média da crença em seus valores fornecidos.

- *Investment* - As fontes "investem"sua confiabilidade de maneira uniforme entre seus valores fornecidos, e em seguida coletam sua credibilidade a partir do valor de confiança das informações providas por elas.

- *PooledInvestment* - É uma variante do *Investment*, a diferença é que a confiança dos valores fornecidos é dada como uma função linear, definida a partir da soma da confiabilidade investida das fontes fornecedoras.

Para aplicar conhecimento a priori aos algoritmos de descoberta da verdade, o conhecimento é traduzido em um programa linear. É realizado um procedimento iterativo até a convergência ou outro critério de parada, das seguintes etapas: i) calcular a confiabilidade das fontes; ii) calcular a confiança dos valores; iii) obter os valores corretos com *Linear Programming (LP)*.

O primeiro passo para traduzir o conhecimento em um programa linear é formulá-lo em lógica proposicional. Depois, as cláusulas proposicionais são convertidas em forma normal conjuntiva. Para exemplificar as cláusulas proposicionais, a partir do conhecimento *"Tom is older than John and a person has exactly one age"*, considerando os seguintes valores fornecidos: *Age(Tom, 30)*, *Age(Tom, 40)*, *Age(John, 25)*, *Age(John, 35)*, as cláusulas proposicionais seriam:

- $Age(Tom, 30) \rightarrow Age(John, 25) \bigwedge (Age(Tom, 30) \bigoplus Age(Tom, 40) \bigwedge (Age(John, 25) \bigoplus Age(John, 35))$

Os experimentos foram conduzidos em diferentes domínios. Por exemplo, no domínio populacional, o senso comum utilizado como conhecimento foi: "a população cresce com o tempo". Também foi considerado conhecimento prévio do tamanho de algumas cidades, sendo o senso comum: “as cidades maiores são mais populosas”. Foram selecionadas randomicamente 2500 pares (a, b), onde a cidade a é mais populosa que a cidade b em um determinado ano t. No domínio biográfico, com dados como data de nascimento, de falecimento, filhos, entre outros, os conhecimentos utilizados foram: "ninguém morre antes de nascer", "as pessoas são inférteis antes dos 7 anos", "ninguém vive depois dos 125 anos", "todos os cônjuges têm vidas sobrepostas", "nenhuma criança nasce mais de um ano depois da morte de um pai (ou mãe)", "ninguém tem mais de dois pais (pai e mãe)", e "ninguém nasce ou morre depois de 2008 (ano de coleta de conjunto de dados)". Com isso, os autores concluíram que o conhecimento prévio

tanto de senso comum, quanto conhecimentos específicos, melhoram a acurácia dos métodos, e diminuem o tempo de convergência do algoritmo.

HLCR (NAKHAEI; AHMADI, 2017) - Nesse trabalho, é proposta uma abordagem de resolução de conflitos de alto nível (***H**igh **L**evel **C**onflict **R**esolution*), baseada em modelo gráfico para resolução dos conflitos, e utiliza relações entre objetos para inferir a verdade. A ideia básica por trás deste artigo é que as informações existentes nos relacionamentos entre objetos podem ajudar a resolver conflitos e descobrir a verdade, porque objetos relacionados possuem alguns atributos que são comuns entre eles. Por exemplo, duas pessoas que são colegas de classe na universidade têm o mesmo nível de educação. Ou dois livros publicados por um determinado editor têm o mesmo domínio ou domínio relacionado.

Resumidamente, a abordagem tem dois objetivos principais: encontrar o relacionamento entre os objetos e estimar valores verdadeiros utilizando a relação entre os objetos. Para o primeiro, os autores introduzem os conceitos de identificador de atributo - elementos que descrevem os atributos e são extraídos do próprio atributo ou de outros (e.g., como palavras-chave que podem ser extraídas do atributo título de um livro). Dois objetos que possuem mais que um limiar de identificadores em comum são potencialmente relacionados. Para o segundo objetivo, um Grafo de Resolução de Conflito (CRG - *Conflict Resolution Graph*) não direcionado foi proposto junto à abordagem. Os nós são pares de objetos-atributos e a relação entre objetos é estimada por identificadores de atributos.

Os experimentos foram realizados com dados reais e sintéticos. Os autores concluíram que a abordagem proposta supera as técnicas existentes pois utiliza informação adicional que existe entre os objetos. No entanto, quando mais de 50% das fontes são confiáveis, o desempenho da abordagem é similar a dos métodos comparados, ou seja, na ausência de fontes confiáveis o HLCR tem melhores resultados que os outros métodos.

Pradhan, Aref e Prabhakar (2018) - Nesse trabalho, os autores abordam o problema de integrar relacionamentos de entidades no processo de fusão de dados para melhorar a eficácia dos modelos. São explorados três tipos de relacionamentos:

- Subsunção/Sobreposições - um valor fornecido por uma fonte pode ser parte de um ou mais valores fornecidos (e.g., pop e rock são gêneros musicais, e são generalizações do gênero Pop/Rock). O gênero Pop/Rock implica ou apoia os gêneros pop e rock.
- Equivalência - entidades do mundo real podem ser referidas de forma diferente por diferentes fontes e contextos (e.g., Hip Hop e Rap são referidos como sinônimos em

algumas culturas e contextos). Deste modo, qualquer fonte que provê Hip Hop como um gênero musical, concorda com Rap. A relação entre tais reivindicações induz uma implicação bidirecional, isto é, ambas as reivindicações implicam uma na outra.

- Exclusão Mútua - Em algumas situações, quando um valor é dado como verdade isso requer que todos os outros valores sejam dados como falsos (e.g., uma música pode ser do gênero Alt R&B ou clássico, mas não ambos. Se um é considerado verdadeiro, o outro deve ser declarado falso).

A partir desses relacionamentos, foram considerados dois temas: implicação e exclusão mútua. Implicação resume as relações de subsunção, sobreposições e equivalência e indica valores que podem estar corretos ou incorretos ao mesmo tempo. A exclusão mútua determina um conjunto de valores que não podem ser simultaneamente corretos. O modelo é representado no formato de um grafo direcionado. Os vértices representam o conjunto de valores distintos, e as arestas representam o relacionamento entre os valores dos vértices correspondentes.

Figura 9 – Informações sobre gêneros musicais de quatro músicas providas por quatro *websites*. Os valores corretos estão marcados com (*).

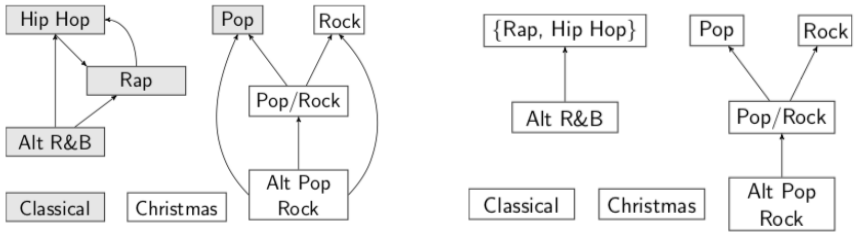
ID	Data Item	S_1	S_2	S_3	S_4	S_5
O_1	**Silent Night**		*Christmas*	*Pop**	*Pop/Rock**	
O_2	**Feel It Still**	*Pop**	{*Alt Pop Rock**, *Rap*}	*Rock**	*Pop/Rock**	*Pop**
O_3	**Perfect**		*Pop**	*Classical*	*Pop/Rock**	*Classical*
O_4	**Unforgettable**	*Rap**	{*Pop*, *Alt R&B**}	*Classical*	*Hip Hop**	

Fonte: Pradhan, Aref e Prabhakar (2018)

Para exemplificar suponha os dados apresentados na Figura 9. Os *websites* fornecem informações conflitantes para um mesmo atributo de um objeto, e.g., Fonte S2 fornece o valor *Christmas* como gênero para a música S*ilent Night*, enquanto a Fonte S3 fornece o valor Pop, e S4 fornece Pop/Rock. Nesse contexto, diferentes relacionamentos podem ser identificados: i) valores podem ser hierarquicamente relacionados (e.g., pop/rock é um subgênero dos gêneros pop e rock; Alt R&B possui suas origens no Hip Hop); ii) valores diferentes podem ter o mesmo significado (e.g., Hip Hop e Rap são considerados equivalentes); iii) valores podem ser mutualmente exclusivos (e.g., A música *Unforgettable* não pode ser simultaneamente dos gêneros clássico e Hip Hop). Para este exemplo, os relacionamentos foram obtidos a partir da DBpedia e *AllMusic*, um guia musical online. Dado o conhecimento de como os diferentes gêneros musicais se relacionam, os modelos de fusão podem se beneficiar deste conhecimento e reavaliar as probabilidades dos valores serem verdades ou não.

Para representar os relacionamentos dos dados e possibilitar a integração do conhecimento com diferentes modelos de fusão, é proposto modelar os relacionamentos em um grafo não direcionado. Como este grafo pode possuir um grande número de vértices e arestas redundantes, o grafo é processado em duas etapas para obter uma representação concisa: i) remover os vértices redundantes; ii) reduzir o número de arestas ao menos possível, mantendo a mesma alcançabilidade. Na Figura 10 podemos visualizar o grafo direcionado e o grafo modificado.

Figura 10 – A esquerda, o grafo direcionado dos relacionamentos de entidades entre os valores para um atributo específico. A direita, o grafo modificado.



(a) Grafo direcionado (b) Grafo direcionado modificado

Fonte: Pradhan, Aref e Prabhakar (2018)

Depois, o objetivo é integrar o grafo de relacionamentos com modelos existentes de fusão de dados. Na etapa de calcular a qualidade das fontes, utilizando os relacionamentos, uma fonte além dos valores diretamente fornecidos por ela, também apoia implicitamente os valores que são apoiados pelos valores fornecidos por ela. Por exemplo, considerando a fonte S2 na Figura 9, e utilizando o grafo modificado da Figura 10, o cálculo de qualidade da fonte será reavaliado conforme apresentado na Figura 11. Comparando a Figura 9 com a Figura 11, pode-se observar que dos 11 valores fornecidos e apoiados pela fonte S2, 8 estão corretos, resultando em uma precisão de 0.73. Antes de reavaliar a qualidade da fonte, sem considerar os valores apoiados, apenas 3 valores eram dados como corretos, tendo um valor de precisão de 0.5. A cobertura também aumenta de 0.27 para 0.73. Na segunda etapa, a qualidade da fonte é utilizada para calcular a exatidão dos valores. A exatidão do valor é calculada com base não apenas na fonte que o provê, mas também nas fontes que os apoiam implicitamente.

Figura 11 – Novos dados contendo os valores suportados pelos valores fornecidos pela fonte S2.

ID	$V_i(S_2)$	$\overrightarrow{V_i}(S_2)$	**Correct**
$\mathbf{O_1}$	Christmas	Christmas	Pop, Pop/Rock
$\mathbf{O_2}$	Alt Pop Rock, Rap	Alt Pop Rock, Pop/Rock, Pop, Rock, Rap	Alt Pop Rock, Pop/Rock, Pop, Rock
$\mathbf{O_3}$	Pop	Pop	Pop/Rock, Pop
$\mathbf{O_4}$	Pop, Alt R&B	Pop, Alt R&B, Hip Hop, Rap	Alt R&B, Hip Hop, Rap

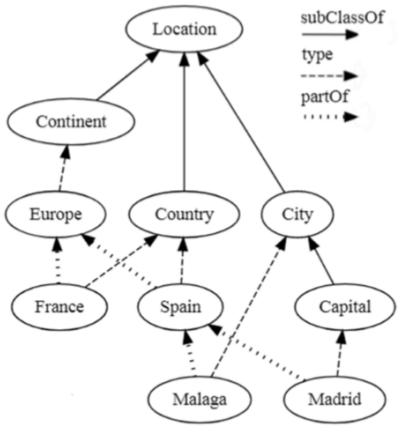
Fonte: Pradhan, Aref e Prabhakar (2018)

Nos experimentos a eficácia da utilização do conhecimento sobre relacionamentos foi avaliada. Para isso, o *dataset* Restaurantes foi utilizado. O atributo localização do restaurante foi dividido em diferentes granularidades, separado em valores distintos. Os relacionamentos foram extraídos da *Wikipedia* e *Google maps*. Os autores demonstraram a aplicabilidade da abordagem proposta em diferentes modelos de fusão existentes, e que comparado a outros métodos, o algoritmo proposto alcança uma melhora significativa nos resultados. No entanto, a eficácia da abordagem proposta depende da completude do conhecimento extraído.

Beretta et al. (2016), Beretta et al. (2018) - Nesse trabalho, é estudado como adaptar modelos de descoberta da verdade, cujo objetivo é selecionar os valores verdadeiros para predicados funcionais, incorporando conhecimento a priori em forma de *partial order* (e.g. relacionamento de subsunção em uma ontologia). A principal ideia é que esse conhecimento pode ser considerado para melhorar as estimativas de confiança dos valores e de confiabilidade das fontes.

Para um melhor entendimento, considere a Figura 12. Supondo que o objetivo da tarefa de descoberta da verdade, nesse caso, é identificar o local de nascimento do pintor *"Pablo Picasso"*, diferentes fontes forneceram valores. Esses valores podem ser vistos na Figura 12(b). Esses valores podem ser parcialmente ordenados por sua granularidade, como pode ser visto na Figura 12(a). A partir daí, sabe-se que *Malaga* e *Granada* são cidades espanholas; que nem todos os valores são conflitantes; que *Malaga* e *Granada* não podem ser ambas verdadeiras. Essas considerações são possíveis devido ao conhecimento prévio dos relacionamentos de dependência entre os valores fornecidos pelas fontes.

Figura 12 – Exemplo de ordem parcial que pode existir entre valores, e relação com valores fornecidos pelas fontes.



(a) Ordem parcial entre valores.

Source	Object	Predicate	Value
A	Pablo Picasso	bornIn	Spain
B	Pablo Picasso	bornIn	Granada
C	Pablo Picasso	bornIn	Europe
D	Pablo Picasso	bornIn	Màlaga

(b) Valores fornecidos por diferentes fontes

Fonte: Pradhan, Aref e Prabhakar (2018)

Esse trabalho é o primeiro a propor o uso de ordem parcial (como conhecimento a priori) em abordagens de descoberta da verdade. Aqui, os autores assumem que as dependências entre valores são conhecidas a priori na forma de ordem parcial, modeladas como uma ontologia. Então, é proposta uma abordagem de pós-processamento capaz de selecionar os valores verdadeiros dadas as estimativas de confiança retornadas por qualquer modelo de descoberta da verdade que considere valores estruturados. Para isso, o algoritmo *Sums* (PASTERNACK; ROTH, 2010) é adaptado para considerar conhecimento a priori. Este processo se divide em três passos: i) selecionar os melhores valores verdadeiros candidatos - permite recuperar o valor verdadeiro mais específico possível e todos os seus ancestrais utilizando informações disponíveis, como índices de confiança e ordenação parcial de valores, ii) ordenar os valores selecionados, com base em critérios predefinidos, e iii) filtrar os top k valores. Em tarefas de descoberta da verdade, geralmente o objetivo final k deve ser igual a 1. No entanto, em casos que há incerteza, pode ser útil retornar um conjunto de valores, mesmo se o predicado for funcional (i.e., na lógica formal um predicado p é considerado funcional se para qualquer sujeito há apenas um único valor $v \in V$ para o qual $p(sujeito, v)$ é verdadeiro).

Os experimentos para validação da proposta foram realizados com *datasets* sintéticos. Cada conjunto de dados sintético contém um conjunto de declarações referentes a um predicado específico, um conjunto de fontes e o subconjunto de declarações fornecidas por cada fonte. Diversas configurações do algoritmo foram testadas em diferentes cenários. Os resultados confirmam que o uso de ordenação parcial de valores ajuda a melhorar o cálculo de confiabilidade da fonte e, consequentemente, a descoberta da verdade. Especificamente, os melhores resultados são obtidos com a configuração do algoritmo que seleciona um conjunto de alternativas e as classifica por meio da confiabilidade das fontes que fornecem esses valores. Os resultados mostraram um comportamento semelhante em ambos conjuntos de dados, obtidos por duas ontologias diferentes (*DBpedia* e *Gene Ontology*).

3.6 CONSIDERAÇÕES

Os primeiros trabalhos de Fusão de Dados e Resolução de Conflitos em sua maioria eram baseados em regras. Inicialmente as soluções não resolviam conflitos, apenas evitavam ou ignoravam, o que logo se tornou ineficaz. Com cada vez mais dados sendo disponibilizados, e mais conflitos acontecendo, foram propostas outras soluções para resolução de conflitos. Funções como valor médio, mínimo, valor mais recente, ou voto, eram aplicados nos dados

para decidir o valor de um atributo entre valores conflitantes.

Novas soluções foram propostas ao longo dos anos com o intuito de melhorar a eficiência do processo. Com isso, pesquisadores propunham explorar diferentes características, como qualidade da fonte de dados, dependência entre fontes de dados, dificuldade e popularidade de objetos, entre outros. Também foram empregadas diferentes técnicas para isso, como Inferência Bayesiana, *Bootstrapping*, redes neurais, entre outros.

Entre novas características exploradas pelas soluções de Fusão de Dados, alguns autores como Pasternack e Roth (2010), Nakhaei e Ahmadi (2017), Pradhan, Aref e Prabhakar (2018), Beretta et al. (2018), propuseram explorar relacionamentos entre os dados. Em Pasternack e Roth (2010) relacionamentos são explorados ao incorporar conhecimento prévio em algoritmos de descoberta da verdade. Em Nakhaei e Ahmadi (2017), os autores exploram relacionamentos entre objetos, onde os objetos que possuem identificadores em comum são relacionados. No trabalho de Pradhan, Aref e Prabhakar (2018), as afirmações feitas pelas fontes de dados são modelados em grafos, para identificar relacionamentos entre elas. O principal tipo de relacionamento utilizado é de Subsunção/Sobreposição, encontrados em valores hierárquicos. Por fim, em Beretta et al. (2018) é proposto utilizar conhecimento a priori para melhorar a estimativa de confiabilidade das fontes. Nesse trabalho o tipo de relacionamento considerado também é de valores hierárquicos.

Diferente dessas soluções, este trabalho se concentra em explorar relacionamentos entre atributos de um determinado domínio de dados, a fim de introduzir conhecimento adicional no processo de descoberta da verdade. Essa abordagem busca utilizar as relações existentes entre os atributos para obter uma fusão de dados mais precisa.

No próximo capítulo, será apresentada a proposta desta tese de forma detalhada.

4 USANDO RELACIONAMENTOS ENTRE ATRIBUTOS NA DESCOBERTA DA VERDADE DO PROCESSO DE FUSÃO DE DADOS

Neste trabalho, o interesse é na Fusão de Dados em cenários de dados na *Web*, nos quais um grande número de fontes de dados pode fornecer valores para apenas alguns atributos das entidades, enquanto apenas um pequeno número de fontes cobre a maioria dos atributos. Este fenômeno vem sendo nomeado na literatura como *Long-tail* (BERTI-ÉQUILLE, 2018).

Mais especificamente, o foco deste trabalho é na descoberta da verdade. A descoberta da verdade é muitas vezes declarada como sinônimo de Fusão de Dados por diferentes autores. No entanto, no contexto deste trabalho, é vista como parte do processo de Fusão de Dados e visa integrar as informações oriundas de múltiplas fontes de maneira consistente, supondo que as outras etapas do processo de integração já tenham sido realizadas (i.e., Alinhamento de Esquemas e Resolução de Entidades). Este capítulo apresenta a proposta deste trabalho com base neste contexto.

A Seção 4.1 apresenta algumas definições preliminares relacionadas ao contexto da abordagem proposta. Na Seção 4.2 é definido o problema que será investigado, e a hipótese de pesquisa. A Seção 4.3 apresenta a visão geral da abordagem proposta, a arquitetura (4.3.1), e um exemplo ilustrativo da utilização da solução proposta (4.3.4).

4.1 DEFINIÇÕES PRELIMINARES

Nesta seção, são apresentadas algumas notações e definições de conceitos fundamentais utilizados para a especificação da estratégia proposta.

Fonte de Dados. Uma Fonte de Dados f fornece dados sobre entidades do mundo real. Um conjunto de Fontes de Dados é representado por $F = \{f_1, f_2, ..., f_n\}$.

Entidade. Uma entidade e é a representação de um conceito do mundo real, como uma música, um livro, ou uma pessoa. As entidades são expressas por um conjunto de atributos $A = \{a_1, a_2, ...a_m\}$. Cada entidade e possui um ou mais atributos do conjunto A que a identifica unicamente.

Domínio. Em um domínio de dados D existem vários conjuntos de Instâncias I referentes a diferentes entidades e, fornecidos por um conjunto de fontes de dados F.

Instância. Uma instância i_n é um valor associado a uma entidade e_j, fornecida por uma fonte de dados f_i, denotada por $f_i.i_n$. Uma instância i_n é definida por um conjunto de pares

$\{(a_1, v_1), (a_2, v_2), ..., (a_j, v_k)\}$, tal que $a_i \in A$, e v_i é o valor de a_i para a entidade e_j na fonte f_i.

Conjunto de Instâncias. Seja $I = \{(i_1, f_1), (i_2, f_2), ...(i_i, f_k)\}$ um conjunto de instâncias oriundo de diferentes fontes de dados, no qual todas as instâncias representam diferentes entidades e de um mesmo domínio D.

Regra de Domínio. Seja r uma regra de domínio que se aplica quando os valores de um atributo a_x determinam os valores de um atributo a_y. Uma regra de domínio é expressa como $r : a_x \rightarrow a_y$. Um conjunto de regras de um domínio D é representado por $R = \{r_1, r_2, ..., r_z\}$

Afirmação. Uma afirmação $afirm$ é um par de valores (v_r, v_t) para uma determinada Regra de domínio r. Uma afirmação é expressa por $afirm : v_r \rightarrow v_t$. Uma dada regra r quando aplicada nos dados gera um conjunto de afirmações $H = \{afirm_1, afirm_2, ..., afirm_p\}$.

4.2 DEFINIÇÃO DO PROBLEMA

A necessidade de avaliar a veracidade dos dados tem aumentado significativamente nos últimos anos, principalmente devido à facilidade de publicação da informação pelos usuários. Dentre os dados disponibilizados, é comum encontrar valores incorretos, inventados, e principalmente, conflitantes, o que aumenta a incerteza e imprecisão dessas informações. A qualidade dos dados varia de acordo com a qualidade das fontes que os fornece. Pela necessidade de avaliar a veracidade dos dados, a descoberta da verdade tem recebido muita atenção em diversas áreas, incluindo a Fusão de Dados.

No entanto, em alguns cenários, torna-se um desafio realizar a descoberta da verdade. Em cenários de dados na *Web*, por exemplo, é comum ocorrer um fenômeno onde a maioria das fontes fornece valores para apenas alguns atributos das entidades, enquanto que um pequeno número de fontes fornece valores para muitos atributos. Ao extrair esses dados, em uma consulta, muitos atributos de uma entidade podem estar ausentes. Este problema tem recebido atenção de pesquisadores da área de Fusão de Dados e descoberta da verdade (BERTI-ÉQUILLE, 2018; LI et al., 2014a; XIAO et al., 2016), e é comumente chamado de *Long-tail Phenomenon*.

O principal desafio se dá pois o estado da arte do processo de descoberta da verdade se baseia basicamente na confiabilidade das fontes, e em cenários onde a maioria das fontes possui uma baixa cobertura (ou seja, fornece poucos valores), o cálculo de confiabilidade das fontes pode ser comprometido, acarretando em valores de confiabilidade imprecisos. Por este

motivo, calcular a qualidade, e utilizá-la no processo de descoberta da verdade somente com base nos dados fornecidos por elas pode ser ineficaz. Deste modo, é necessário aperfeiçoar a maneira como a descoberta da verdade é realizada, e isso é um desafio a ser superado.

Deste modo, este trabalho tem como objetivo propor uma solução que aborde este desafio. Em um determinado domínio D, dado um conjunto de Fontes de Dados $F = \{f_1, f_2, ..., f_n\}$, que provê conjuntos de instâncias para diferentes entidades e, assumindo que o processo de Resolução de Entidades já foi realizado (VIEIRA; LÓSCIO; SALGADO, 2019), como é possível identificar os valores corretos para o conjunto de atributos $A = \{a_1, a_2, ...a_m\}$ nos conjuntos de instâncias de diferentes entidades e?

Para isso, se propõe considerar relacionamentos entre atributos e utilizá-los no processo de descoberta da verdade. Existem atributos que possuem uma relação direta. Por exemplo, o atributo cep possui uma relação com os atributos rua, $cidade$, $estado$, pois sabendo o valor de cep de um endereço é possível identificar a rua, a $cidade$ e o $estado$ aos quais ele pertence. Neste caso, se *cep* $=$ *"50640-290"* então *rua* $=$ *"Rua Tapiassu"*, *cidade* $=$ *"Recife"*, *estado* $=$ *"Pernambuco"*. Acreditamos que capturar esses relacionamentos pode auxiliar na descoberta da verdade, principalmente em cenários *Long-tail*, pois permite uma avaliação de confiabilidade das fontes e confiança dos valores mais eficiente, já que o cálculo de confiabilidade da fonte e confiança do valor serão realizados não apenas com base nos dados de entrada, mas também no conhecimento adicional obtido. Deste modo pode-se auxiliar no processo de descoberta da verdade penalizando ou reforçando os valores de confiabilidade das fontes e confiança dos valores, com base nesse conhecimento adicional.

4.2.1 Exemplo Motivacional

Para um melhor entendimento, considere como exemplo os dados de entrada contidos na Figura 13, que possui instâncias referentes a um restaurante da cidade de Recife, oriundas de cinco fontes de dados distintas. Suponha que os dados foram alocados em uma tabela, cujas colunas são os atributos fornecidos por todas as fontes, ilustrado no Quadro 5. Como se pode observar, existem diversos conflitos nos dados, tanto de contradição (dois valores não nulos), como de incerteza (valor não nulo com valor nulo). Neste cenário, poderia ser aplicada a votação majoritária, que é o *baseline* da descoberta da verdade. No entanto, em casos de empate, como para o atributo **Fone**, o valor encontrado pode não ser o que realmente reflete o mundo real. Além disso, no cenário atual com grande facilidade de publicação e

compartilhamento de dados, muitas fontes de dados não proveem valores confiáveis. Por isso, torna-se indispensável avaliar a qualidade das fontes de dados de forma satisfatória durante a descoberta da verdade.

Figura 13 – Conjunto de Dados de entrada.

Dados Recife ((Nome, Alphaiate), (Cidade, Recife), (Cep, 50740-201), (Rua, Rua Arthur Muniz), (Fone, 3419-7588))
TripAdvisor ((Nome, Alphaiate), (Cidade, Recife))
Facebook ((Nome, Alphaiate), (Cidade, Recife), (Cep, 69103-944))
GuiaMais ((Nome, Alphaiate), (Cidade, Recife), (Rua, Avenida Boa Viagem), (Area, 81), (Fone, 3465-7588))
ChefsClub ((Nome, Alphaiate), (Cidade, São Paulo), (Cep, 32023-332), (Rua, Praça La Coruna), (Area, 81), (Fone, 3467-5461))

Quadro 5 – Um exemplo motivacional.

Fonte	Nome	Cidade	Estado	Cep	Rua	Area	Fone
Dados Recife	Alphaiate	Recife	PE	50740-201	Rua Arthur Muniz		
TripAdvisor	Alphaiate	Recife		69103-944			
Facebook	Alphaiate	Recife			Avenida Boa Viagem	81	3419-7588
GuiaMais	Alphaiate				R. Arthur Muniz		
ChefsClub	Alphaiate	São Paulo		32023-332	Praça La Corunã	81	3467-5461

Fonte: Elaborada pela autora (2023).

Porém, em cenários com muitos valores nulos, o cálculo de confiabilidade das fontes pode ser impreciso, o que compromete a eficiência da descoberta da verdade. Por exemplo, a fonte **GuiaMais** fornece apenas valores para os atributos **Nome** e **Rua**. Já a fonte **ChefsClub** só não fornece valor para o atributo **Estado**. Deste modo, a cobertura da fonte **ChefsClub** é bem maior do que a cobertura da fonte **GuiaMais**, o que não significa que a fonte **GuiaMais** é menos confiável, mas que o valor de confiabilidade dessa fonte, obtido no cálculo de confiabilidade pode não reproduzir sua confiabilidade real, já que é baseado em uma cobertura muito baixa, diferente da confiabilidade da fonte **ChefsClub**, que tem uma alta cobertura. Ou seja, fontes com baixa cobertura são prejudicadas no processo de avaliação de confiabilidade. Como a cobertura da maioria das fontes em cenários *Long-tail* é baixa, avaliar a confiabilidade apenas com base nos dados de entrada pode não ser suficiente.

Sabendo desses desafios, como é possível descobrir os valores verdadeiros de cada atributo entre os dados conflitantes (incerteza e contradição), gerando uma representação única e completa de uma entidade? Ou então, como se pode adaptar a estimativa de confiabilidade das fontes para estes cenários, e torná-la mais consistente?

Sabe-se que os dados podem possuir relacionamentos entre os atributos. Por exemplo, no Quadro 5, os atributos *Cep*, *Rua*, *Cidade*, *Estado* possuem uma relação em que tendo conhecimento do Cep, pode-se saber a Rua, Cidade e Estado relacionados. Outro exemplo acontece com o atributo *Cidade* e *Area*, onde sabendo o código de Area é possível definir a Cidade relacionada a ele. Ainda tendo conhecimento da Cidade também pode-se definir o Estado. Deste modo, utilizar relacionamentos entre atributos pode ajudar no processo de Descoberta da Verdade?

4.3 SOLUÇÃO PROPOSTA

Para abordar o desafio apresentado na Seção 4.2, a proposta desta tese é uma solução baseada em inserir conhecimento adicional na Descoberta da Verdade por meio da utilização de relacionamentos entre atributos. Pretende-se deixar claro que neste trabalho não há interesse em propor uma solução nova para o processo de Descoberta da Verdade, e sim uma abordagem adaptável que se propõe à melhorar os resultados dos algoritmos do estado da arte.

A seguir, serão apresentadas a arquitetura da abordagem, a descrição do processo, e um exemplo ilustrativo.

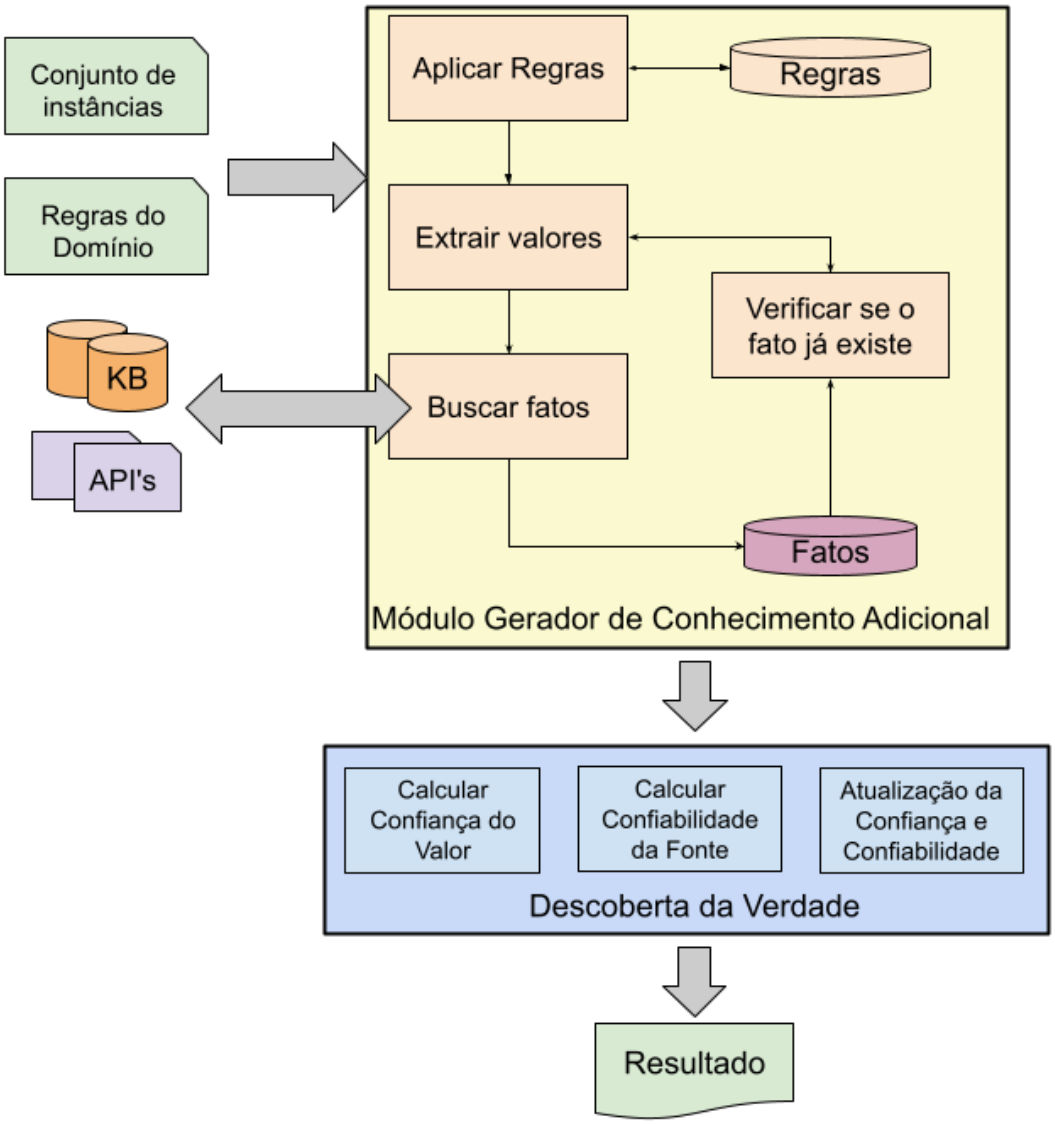
4.3.1 Arquitetura da Abordagem Proposta

A Figura 14 apresenta a arquitetura da abordagem proposta. Esta arquitetura é composta por dois módulos: i) módulo gerador de conhecimento adicional; ii) módulo de descoberta da verdade. Cada módulo é explicado com mais detalhes a seguir.

Módulo Gerador de Conhecimento Adicional Neste módulo, as regras são definidas para o domínio específico dos dados. As regras são armazenadas em um repositório, e posteriormente aplicadas sobre os dados de entrada, para extrair os valores, gerando afirmações. A partir dessa etapa, são realizadas buscas na *Web* para confirmar ou não essas afirmações. Quando uma afirmação após a busca é dada como verdadeira, dizemos que a afirmação é válida, e ela então é armazenada no repositório de fatos. Caso contrário, a afirmação é desconsiderada.

Módulo de Descoberta da Verdade - No módulo de Descoberta da Verdade, os modelos existentes podem ser implementados/adaptados (LI et al., 2014a; FANG, 2017; BROELEMANN; KASNECI, 2018; ZHANG et al., 2018). Em sua maioria, esses modelos são iterativos em duas

Figura 14 – Arquitetura da abordagem proposta.



Fonte: Elaborada pela autora (2023).

etapas: avaliação da confiança do valor e avaliação de confiabilidade das fontes. Após o módulo de descoberta de relacionamentos ser executado e as consultas na *Web* obterem as afirmações corretas, o conhecimento armazenado no repositório de fatos é utilizado para reforçar ou penalizar os valores de confiança e confiabilidade.

4.3.2 Módulo gerador de Conhecimento adicional

O módulo gerador de conhecimento adicional está ilustrado no Algoritmo 1. Na linha 1, o algoritmo recebe como entrada um conjunto de instâncias de entidades de um mesmo domínio, oriundas de múltiplas fontes, e o conjunto de atributos dessas entidades. Também são dadas

como entrada as regras geradas por um especialista de domínio. Na linha 6, para cada regra do conjunto de regras, se aplica no conjunto de instâncias gerando afirmações. Essas afirmações são traduzidas em consultas, e buscadas na *Web*. As afirmações válidas são dadas como fatos e armazenados no repositório de fatos para posterior utilização. Quando um novo conjunto de dados do mesmo domínio for a entrada do processo, pode-se verificar que já existem regras deste domínio no repositório de regras. Essas regras já foram aplicadas em dados e já geraram fatos, que também podem ser reutilizados. Conforme mais processos forem sendo realizados, e mais fatos forem sendo armazenados, o resultado da descoberta da verdade possivelmente será mais correto, pois quanto mais, regras criadas, e fatos armazenados, maior o conhecimento adicional sobre os dados.

Algoritmo 1 Algoritmo para geração dos fatos

1: **function** ExtraiFatos(I, $Regras$)
   ▷ I corresponde ao conjunto de instâncias das entidade do domínio D;
   ▷ $Regras$ corresponde ao conjunto de regras de dominio
2: $AFIRM' \leftarrow \emptyset$ ▷ inicializa o conjunto de novas afirmações
3: $Q \leftarrow \emptyset$ ▷ inicializa a query de busca
4: **for all** $r \in Regras$ **do** ▷ Percorre cada regra r do conjunto de regras
5: $AFIRM \leftarrow aplicaRegras(r, I)$ ▷ aplica a regra nos dados e gera as afirmações
6: $Q \leftarrow traduzAfirmacoes(AFIRM)$ ▷ traduz as afirmações em linguagem de consulta
7: $AFIRM' \leftarrow realizaBusca(Q)$ ▷ realiza a busca das afirmações na *Web* e retorna um subconjunto de afirmações válidas
8: $Fatos \leftarrow armazenaRepositorioFatos(AFIRM', r)$
9: **end for**
10: **end function**

Caso nos dados existam novos atributos que não passaram pelo processo anterior, pode-se verificar a existência de novas regras com esses novos atributos, não sendo necessário repetir o processo para todos os atributos do domínio.

4.3.3 Módulo de Descoberta da Verdade

No processo de descoberta da verdade, pode-se adaptar outros algoritmos do estado da arte que implementem um processo iterativo para inferir confiabilidade das fontes e confiança dos valores (YIN; HAN; YU, 2008; LI et al., 2014a; FANG, 2017; ZHANG et al., 2018). O núcleo da descoberta da verdade é um algoritmo iterativo. Primeiro a confiabilidade de cada fonte é inicializada. Para cada atributo da entidade, o método calcula a confiança de cada valor provido

a partir da confiabilidade das fontes que os forneceu. Em seguida, atualiza a confiabilidade de cada fonte a partir do valor de confiança dos valores fornecidos por elas. O procedimento se repete até que uma condição de parada seja satisfeita. Geralmente, o critério de parada é um número máximo de iterações, ou um limiar de variabilidade sobre os resultados (ou seja, se a confiança do valor ou a confiabilidade da fonte não variar de uma iteração para outra).

Neste trabalho, está sendo utilizado o algoritmo ***TruthFinder*** (YIN; HAN; YU, 2008), um dos mais utilizados da literatura. O *TruthFinder* é um modelo computacional que possui como base quatro heurísticas:

- Heurística 1. Geralmente há apenas um valor verdadeiro para uma propriedade de um objeto
- Heurística 2. Este valor verdadeiro parece ser o mesmo ou semelhante em diferentes fontes
- Heurística 3. Os valores falsos em fontes diferentes são menos prováveis de serem os mesmos ou semelhantes
- Heurística 4. Em um determinado domínio, uma fonte que fornece, em sua maioria, valores verdadeiros para muitos objetos, provavelmente fornecerá valores verdadeiros para outros objetos.

Baseado nessas heurísticas, o algoritmo prevê que um valor fornecido por muitas fontes confiáveis, provavelmente é verdadeiro. Se um valor estiver em conflito com valores fornecidos por fontes confiáveis, provavelmente ele é falso. Por outro lado, uma fonte é confiável se fornecer valores verdadeiros. Assim fica claro que tanto a confiabilidade das fontes quanto a confiança dos valores são determinadas uma pela outra, tornando o processo iterativo.

O *Truthfinder* é baseado em calcular de forma iterativa a confiança dos valores e a confiabilidade das fontes. A confiança de um valor v é a probabilidade de v ser verdade. A confiabilidade de uma fonte de dados f é o valor de confiança esperado para os fatos providos por ela. O algoritmo se divide nesses dois passos: i) cálculo de confiança dos valores; ii) cálculo de confiabilidade das fontes. Nesse processo, ambos os valores são dependentes um do outro. Os cálculos serão resumidamente apresentados a seguir. Mais detalhes podem ser encontrados em (YIN; HAN; YU, 2008).

A confiança do valor é calculada em 3 partes : primeiro, dado um valor $v \in I(f)$, sendo $I(f)$ um conjunto de instâncias contendo valores fornecidos por uma fonte f, o cálculo de confiança do valor é dada pela Equação 4.1.

$$\sigma(v) = \frac{\sum r(f)}{f \in F(v)} \tag{4.1}$$

Sendo $r(f)$, o valor de confiabilidade de uma fonte de dados f.

Na segunda parte, a confiança do valor é ajustada considerando os valores que são similares a ele. Nesta etapa, obtém-se o valor para $\sigma*(v)$ que diz respeito ao valor de confiança ajustado considerando a influência dos outros valores. Por fim, na Equação 4.2 a confiança é ajustada novamente considerando as dependências entre fontes de dados (casos onde uma fonte copia de outra).

$$s(v) = \frac{1}{1 + e^{-\gamma\sigma*(v)}} \tag{4.2}$$

Para o cálculo de confiabilidade de uma fonte de dados f, se considera a confiança dos valores providos por f. O cálculo pode ser visto na Equação 4.3.

$$t(w) = \frac{\sum_{f \in F(w)} s^{mod}(f)}{\mid F(w) \mid} \tag{4.3}$$

O processo itera atualizando confiança dos valores e confiabilidade das fontes até atingir estabilidade. Essa estabilidade é medida pela mudança na confiabilidade das fontes. Se de um ciclo de iteração para outro, a mudança for suficientemente baixa, o algoritmo encerra.

Na presente proposta é adicionado um novo ajuste no cálculo de confiança dos valores, considerando a base de fatos. O cálculo pode ser visto na Equação 4.4, onde c é o valor de ponderação da confiança, e y é o valor de ponderação dos fatos considerados.

$$s^{mod} = s(v) * c + y \tag{4.4}$$

Deste modo, além dos dados em si para calcular a confiança dos valores será utilizado o conhecimento adicional contido no repositório de fatos. Por consequência, na atualização de confiabilidade das fontes os fatos também serão considerados. O processo de descoberta da verdade está ilustrado no Algoritmo 2.

Na etapa de avaliação de confiança dos valores, verifica-se se existem fatos que podem ajudar no processo. Sendo assim, os fatos são utilizados para atualizar o cálculo de confiança do valor e de confiabilidade das fontes, reforçando ou penalizando esses valores. A saída do

Algoritmo 2 Algoritmo de Descoberta da Verdade

1: **function** DESCOBREVERDADE(I, F)
   ▷ I corresponde ao conjunto de instâncias;
   ▷ F corresponde ao conjunto de fontes de dados;
2: $C \leftarrow \emptyset$ ▷ inicializa a confiança do valor
3: $Confs \leftarrow \emptyset$ ▷ inicializa os valores de confiabilidade das fontes
4: $Verdades \leftarrow \emptyset$ ▷ conjunto de valores corretos
5: $Fatos \leftarrow ExtraiFatos(I, Regras)$ ▷ aplica as regras nos dados e armazena os fatos no repositório
6: $W_f \leftarrow 0.9$ ▷ inicializa o peso de confiabilidade das fontes
7: **for all** $(i_j, f_i) \in I$ **do**
8: **for all** $(a_i, v_i) \in i_j$ **do**
9: $C \leftarrow AvaliaConfiancaValor(W_f, Fatos.a_i, v_i)$ ▷ avalia a confiança do valor utilizando os fatos armazenados no repositório e o valor de confiabilidade da fonte que o forneceu
10: $W_f \leftarrow AtualizaConfiabilidadeFonte(C, f_i, Fatos)$ ▷ atualiza os valores de confiabilidade da fonte que forneceu o valor com base na confiança dos valores e nos fatos
11: $Confs \leftarrow armazenaValorConfiabilidade(W_f, f_i)$ ▷ armazena os valores de confiabilidade das fontes
12: $Rotulo \leftarrow ClassificaRotuloValor(C, v_i)$ ▷ valor recebe um rótulo de verdadeiro ou falso de acordo com o valor de confiança
13: **if** $Rotulo = true$ **then**
14: $Verdades \leftarrow (a_i, v_i)$ ▷ se o rótulo for verdadeiro, adiciona o valor no conjunto de verdades
15: **end if**
16: **end for**
17: **end for**
18: **return** $Verdades$ ▷ retorna o conjunto de valores identificados como verdade
19: **end function**

processo de descoberta da verdade é para cada entidade, um valor verdadeiro único para os atributos que passaram pelo processo. Também pode-se retornar como saída, se desejado, o valor de confiabilidade das fontes, e o valor de confiança dos valores obtidos no processo.

4.3.4 Exemplo Ilustrativo

Tome como exemplo novamente o Quadro 5. Os passos a seguir ilustram a aplicação da abordagem proposta.

Passo 1. O primeiro passo do processo é criar uma representação completa dessas instâncias. O resultado do primeiro passo pode ser visto no Quadro 5. Com esta representação criada, podemos facilmente visualizar a existência de conflitos, tanto de contradição (quando o conflito

ocorre com dois valores não nulos), quanto de incerteza (quando o conflito ocorre entre um valor não nulo e um valor nulo).

Passo 2. O passo 2 é aplicar as regras de domínio nos dados gerando afirmações. Para este exemplo, dentre outras regras, teríamos:

- $r_1 : [Cep] \rightarrow [Estado]$
- $r_2 : [Cep] \rightarrow [Cidade]$
- $r_3 : [Area] \rightarrow [Cidade]$
- $r_4 : [Cidade] \rightarrow [Estado]$

Aplicando a regra r_1 nos dados, por exemplo, temos a seguinte afirmação:

- $afirm_1 : [50740 - 201] \rightarrow [PE]$

Para a regra $r_3 : [Area] \rightarrow [Cidade]$ são geradas as seguintes afirmações:

- $afirm_2 : [81] \rightarrow [Recife]$
- $afirm_3 : [81] \rightarrow [S\tilde{a}oPaulo]$

Passo 3. Para cada regra, uma consulta é gerada para verificar se as afirmações são fatos ou não. Neste exemplo, utilizando as afirmações $afirm_2$ e $afirm_3$, se teriam as seguintes consultas:

Consulta SPARQL 1

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?areaCode
WHERE
{
  ?city dbo:areaCode ?areaCode .
  VALUES ?city {dbr:Recife}
}
```

Consulta SPARQL 2

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?areaCode
WHERE
{
  ?city dbo:areaCode ?areaCode .
  VALUES ?city {dbr:São Paulo}
}
```

Buscando em uma base de conhecimento, por exemplo, na DBPedia[1], se encontra os valores apresentados na Figura 15.

Figura 15 – Resultados das consultas SPARQL na base de conhecimentos.

Fonte: Elaborada pela autora (2023).

Tendo como resultado da busca que o código de área para a cidade de Recife é 81, pode-se validar a afirmação $afirm_2$ e armazená-la no repositório de fatos. A afirmação $afim_3$ é dada como inválida e descartada. No caso do exemplo, a afirmação $afirm_2$ foi armazenada como um fato no repositório de fatos. Utilizando esse fato para ajudar no processo de descoberta da verdade para o atributo **Area**, tem-se que a fonte *Facebook* forneceu valores corretos para os atributos **Cidade** e **Area**. Enquanto a fonte *ChefsClub* forneceu um valor incorreto, já que confirmamos que o código 81 não é da cidade de São Paulo e sim de Recife. Na etapa de cálculo de confiança do valor, o valor fornecido pela fonte **Facebook** recebe uma confirmação. Já o valor fornecido pela fonte **ChefsClub** recebe uma penalização, tendo seu valor de confiança diminuído. O mesmo acontece com as fontes, onde a fonte **Chefsclub** recebe penalização em

[1] https://wiki.dbpedia.org

sua confiabilidade, e a fonte **Facebook** recebe uma confirmação. A saída do processo é uma instância única e completa, que representa a entidade de maneira mais confiável possível.

Quadro 6 – Conjunto de dados do mesmo domínio para exemplificar reuso dos fatos armazenados.

Fonte	Nome	Cidade	Cep	Rua	Area	Fone	Especialidade
Dados Recife	Guaiamum		50740-201			3441-1509	
TripAdvisor	Guaiamum Gigante	Recife		Rua Oliveira	81		Frutos do Mar
Facebook	Guaiamum	Recife		Rua José Góes			
GuiaMais	Guaiamum						Restaurante
ChefsClub	Ilha do Guaiamum		50740-201	R. Maria Carolina		3466-2122	Frutos do Mar

Fonte: Elaborada pela autora (2023).

Para exemplificar a reutilização das regras geradas a partir dos relacionamentos e dos fatos armazenadas nos repositórios, tome como exemplo o conjunto de dados contido no Quadro 6. Neste exemplo, o primeiro passo seria verificar no repositório de regras se existem regras para esse domínio de dados. Deste modo evita-se refazer todo o processo. Pode-se analisar se no domínio existe algum atributo diferente, e verificar a possibilidade de existir alguma regra relacionada a ele. A partir das regras existentes desse domínio, e de novas regras criadas para novos atributos (se houver), são geradas afirmações com base nos valores desse conjunto de dados. O processo de tradução das afirmações e a busca são realizados, e as afirmações válidas são armazenadas no repositório de fatos. Também é verificado se no repositório de fatos já existem fatos criados para algum valor contido no conjunto de dados, em que o atributo esteja do lado esquerdo da afirmação. Por exemplo: se no repositório de fatos já existe o fato $[Recife] \rightarrow [81]$, pode-se utilizá-lo para auxiliar nesse novo processo de descoberta da verdade.

4.4 CONSIDERAÇÕES

Neste capítulo, foi apresentada a proposta e principal contribuição deste trabalho: uma abordagem para utilizar conhecimento adicional na descoberta da verdade do processo de fusão de dados. Inicialmente, foi dada uma visão geral, bem como as definições preliminares necessárias para o entendimento do problema. Em seguida, foi formulado o problema e a hipótese, juntamente com um exemplo motivacional.

Depois, foi apresentada a solução proposta, a arquitetura, e seus módulos. E por fim, foi

apresentado um exemplo ilustrativo para melhor entendimento. O próximo capítulo apresenta os experimentos realizados para a avaliação da proposta.

5 EXPERIMENTOS

Neste Capítulo serão apresentadas as implementações e experimentos realizados ao longo deste trabalho. O objetivo deste capítulo é avaliar a abordagem proposta no capítulo 4.

5.1 EXPERIMENTO 1

Ao longo de estudos sobre o estado da arte em relacionamentos entre atributos, acreditamos que o tipo de relacionamento que utilizamos na nesta solução seria uma dependência funcional. Por este motivo, foi realizado um levantamento dos algoritmos do estado da arte de descoberta de dependência funcional e suas extensões, tais como dependência funcional aproximada, e dependência funcional condicional, para realizar um experimento e analisar o comportamento desses algoritmos no cenário abordado nesta pesquisa. A intuição inicial era que, para o cenário de Fusão de Dados, a dependência funcional aproximada seria mais indicada, pois os conflitos existentes nos dados poderiam violar facilmente as dependências funcionais.

Deste modo, descobrir dependências funcionais em dados conflitantes pode não ser muito eficaz. Essa suposição também seria avaliada por meio de experimentos. Era pretendido avaliar os algoritmos principalmente em cenários onde ocorrem *Long-tail phenomenon*. A partir dessa avaliação, o algoritmo identificado como mais apropriado para o contexto deste trabalho seria selecionado para utilização e/ou possíveis adaptações.

Um dos trabalhos mais citados na literatura é o de Huhtala et al. (1999), nomeado **TANE**, que tem o objetivo de descobrir dependências funcionais e CTANE, que descobre dependências funcionais aproximadas sobre conjuntos de dados. O TANE/CTANE é implementado em linguagem Python e seu código está disponível[1].

Foi realizado este primeiro experimento utilizando o algoritmo TANE/CTANE em diferentes *datasets*, para responder a seguinte pergunta: Como o algoritmo se comporta em cenários tradicionais? E em cenários com maior quantidade de valores nulos? Primeiramente, o TANE foi testado com o *dataset* adult[2]. Esse conjunto de dados possui mais de 48.000 instâncias e 14 atributos. Foram encontradas 78 dependências funcionais. No *dataset* abalone[3], que possui 4.177 instâncias e 8 atributos, foram encontradas 137 dependências funcionais. Para

[1] https://www.cs.helsinki.fi/research/fdk/datamining/tane/
[2] https://archive.ics.uci.edu/ml/datasets/adult
[3] https://archive.ics.uci.edu/ml/datasets/abalone

exemplificar, o resultado desse *dataset* pode ser visto no Apêndice A. Os resultados para o CTANE foram bem similares.

Para entender melhor porque tantas dependências foram encontradas, foi criado um *dataset* manualmente, contendo 7 atributos e um pequeno número de instâncias extraídas da *Web*. Para testar o algoritmo com valores nulos, foram elencados 3 cenários: i) *dataset* completo; ii) *dataset* com poucos valores nulos; iii) *dataset* com muitos valores nulos. No cenário completo, o TANE retornou 22 dependências funcionais. No cenário com poucos nulos, 23, e no cenário com muitos valores nulos, retornou 21. Para um *dataset* pequeno, ainda é uma grande quantidade de dependências. Com o CTANE, as dependências eram praticamente similares ao TANE. Analisando os resultados, independente de dependência funcional ou aproximada, o retorno não foi satisfatório para o objetivo esperado. Os algoritmos criam muitas dependências que não têm possibilidade de serem buscadas na *Web*. Muita dependência gerada não faz sentido para o objetivo deste trabalho, visto que elas são geradas a partir dos valores dos dados em si. Deste modo, foi verificado que, para o objetivo de encontrar relacionamentos entre atributos para auxiliar na descoberta da verdade, as dependências funcional e aproximada não são interessantes de serem aplicadas.

5.2 EXPERIMENTO 2

Este experimento teve como objetivo avaliar a proposta apresentada no Capítulo 4. Foi realizada a implementação da proposta e os testes com dados. A seguir serão apresentados os detalhes.

5.2.1 Ferramentas

Foi utilizada linguagem Python para a codificação, e as seguintes bibliotecas:

- **SPARQLWrapper** - para executar as consultas SPARQL e retornar os resultados em formatos mais manipuláveis
- **Pandas** - para manipulação dos dados
- **Strsimpy** and **FuzzyWuzzy** - para verificar a similaridade dos dados
- **Numpy** - para operações nos dados, processamento e análise

Também foi utilizado o ambiente **Jupyter Notebook** para testar os códigos. Neste ambiente, é possível criar blocos de texto e blocos de código. Nos blocos de texto, pode-se explicar passo a passo o projeto. Também possibilita explorar os conjuntos de dados, fazer tratamento e limpeza de dados e gerar gráficos de forma simples.

5.2.2 Codificação

Os parâmetros de entrada do algoritmo de descoberta da verdade foram inicializados seguindo as especificações do trabalho original ((YIN; HAN; YU, 2008)). O *core* do algoritmo bem como os parâmetros de inicialização podem ser encontrados no Apêndice B.

5.2.3 Conjuntos de Dados

Os dados utilizados neste trabalho são oriundos da plataforma de dados abertos do governo de Pernambuco. O *dataset*[4] possui 130 entidades de restaurantes da cidade de Recife. As entidades são compostas por sete atributos: id, $nome$, $endereço$, $telefone$, $especialidade$, $site$, $e-mail$. O segundo *dataset*[5] possui 505 entidades referentes a parques e praças da cidade de Recife. As entidades são compostas por treze atributos: id, $nome$, $tipo$, $endereco$, $codigo_logradouro$, lei, $nome_oficial$, $area$, $perimetro$, $codigoBairro$, $bairro$, $latitude$, $longitude$, dos quais foram utilizados apenas id, $nome$, $codigoBairro$, $bairro$.

Na etapa de pré-processamento dos dados, foram necessárias algumas modificações. No *dataset* de restaurantes, o atributo endereço foi dividido em: rua, $número$, $bairro$, $cidade$. Também foi adicionado o atributo cep em ambos os *datasets*. Um atributo $fonte$ foi adicionado ainda, para conter o id da fonte que provê cada instância. Os *datasets* pré-processados foram considerados como *gold standard*.

5.2.4 Resultados

O processo de descoberta da verdade é realizado por atributo. Deste modo, para este experimento focamos nos dados de endereço. Para realizar os experimentos, foram simulados diferentes cenários de dados com diversas fontes de dados provendo valores para as entidades

[4] http://dados.recife.pe.gov.br/dataset/bares-e-restaurantes
[5] http://dados.recife.pe.gov.br/dataset/parques-e-pracas

dos *datasets*. Deste modo, uma fonte de dados provê valores para várias entidades, gerando um conjunto de dados com múltiplas instâncias de uma mesma entidade, e contendo valores conflitantes (nulos e errôneos).

Quadro 7 – Completude dos atributos - cenário 1 - *dataset* Restaurantes.

Atributos	Completude
id	1
fonte	1
nome	1
rua	0,98
numero	0,92
bairro	0,98
cidade	0,88
cep	0,93

Fonte: Elaborada pela autora (2023).

Quadro 8 – Completude dos atributos - cenário 2 - *dataset* Parques e Praças.

Atributos	Completude
id	1
fonte	1
nome	0,93
codigoBairro	1
bairro	1
cep	0,92

Fonte: Elaborada pela autora (2023).

Cenário 1. Neste primeiro cenário a completude dos atributos é alta em ambos os *datasets*, como podemos ver nos Quadros 7 e 8. A completude de um atributo diz respeito a quantidade de instâncias onde esse atributo possui um valor fornecido.

Como regra neste cenário, em ambos os *datasets* temos que $R_1 : Cep- > bairro$ e $R_2 : Cep- > rua$. Deste modo, por meio do algoritmo gerador de conhecimento adicional, as regras são aplicadas nos dados e base de fatos é criada para cada *dataset*. Posteriormente, a base de fatos é utilizada na descoberta da verdade. Alguns fatos que compõem as bases de fatos, geradas por meio do consumo da api viacep[6], podem ser vistos na Figura 17.

Na base de fatos, conforme explicado no Capítulo 4, são armazenados os atributos participantes da regra, e seus respectivos valores que são fato. Com isso, na descoberta da verdade

[6] https://viacep.com.br

Figura 16 – Recorte das bases de fatos geradas a partir do *dataset* Restaurantes(a) e do *dataset* Parques e praças(b).

```
RegraEsq,RegraDir,ValorEsq,ValorDir
0,Cep,Bairro,52050-225,Graças
1,Cep,Bairro,51021-370,Boa Viagem
2,Cep,Bairro,51020-180,Boa Viagem
3,Cep,Bairro,51110-131,Pina
4,Cep,Bairro,51011-550,Boa Viagem
5,Cep,Bairro,51020-190,Boa Viagem
6,Cep,Bairro,50040-050,Santo Amaro
7,Cep,Bairro,51020-031,Boa Viagem
8,Cep,Bairro,51010-000,Pina
9,Cep,Bairro,51021-000,Boa Viagem
10,Cep,Bairro, 50110-900,Santo Amaro
```

(a) *Dataset* Restaurantes.

```
id,RegraEsq,RegraDir,ValorEsq,ValorDir
0,Cep,bairro,52060-240,Parnamirim
1,Cep,bairro,52060-141,Parnamirim
2,Cep,bairro,52060-412,Santana
3,Cep,bairro,52060-411,Parnamirim
4,Cep,bairro,50050-210,Soledade
5,Cep,bairro,50110-005,Santo Amaro
6,Cep,bairro,51010-170,Pina
7,Cep,bairro,52071-525,Monteiro
8,Cep,bairro,50030-280,Recife
9,Cep,bairro,50070-200,Paissandu
10,Cep,bairro,50030-320,Recife
```

(b) *Dataset* Parques e Praças.

Fonte: Elaborada pela autora (2023).

de um dado atributo, é verificado se existe alguma regra cujo atributo esteja ao lado direito da regra. Se sim, isso quer dizer que, para este atributo, existem fatos que podem ser utilizados como auxílio na descoberta da verdade

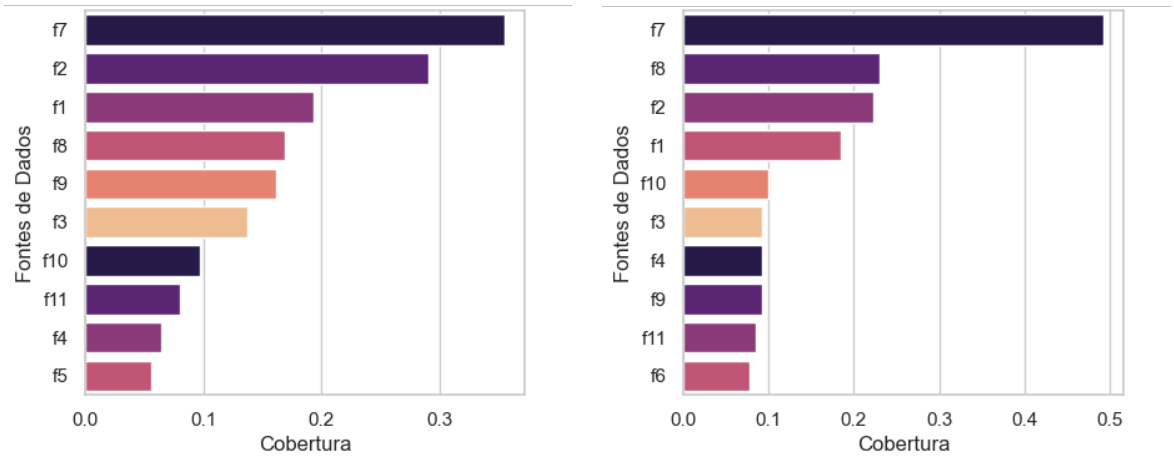
No *dataset* Restaurantes, a média de cobertura das fontes para o atributo bairro é de $0,12$, e para o atributo rua é de $0,13$. Já para o *dataset* Parques e Praças, a média de cobertura das fontes para o atributo bairro é de $0,12$. A cobertura média dos atributos é calculada com base na presença ou ausência desse valor na fonte de dados, em relação ao conjunto total de valores esperados para uma determinado atributo. Sendo assim, podemos concluir que a maioria das fontes provê valores dos atributos bairro e rua para um pequeno número de entidades, enquanto poucas fontes cobrem várias entidades.

Nas Figura 18, pode-se verificar as fontes de dados que possuem maior cobertura para os atributos bairro e rua no *dataset* Restaurantes. Ou seja, as fontes que mais proveem valores para os atributos bairro e rua de diferentes entidades. Já na figura 18, pode-se verificar as fontes que mais proveem valores para o atributo bairro do *dataset* Parques e Praças.

Inicialmente foi realizada a descoberta da verdade no *dataset Restaurante* para o atributo bairro, e rua utilizando o algoritmo *Truthfinder* (YIN; HAN; YU, 2008) original. Em seguida, o a descoberta da verdade foi realizada com o algoritmo modificado (considerando as informações adicionais da base de fatos, criada a partir do módulo gerador de conhecimento adicional). O mesmo processo foi realizado para o atributo bairro do *dataset* Praças e Parques. Os resultados podem ser vistos no Quadro 9.

Podemos notar a partir da comparação dos resultados entre o algoritmo original e o algo-

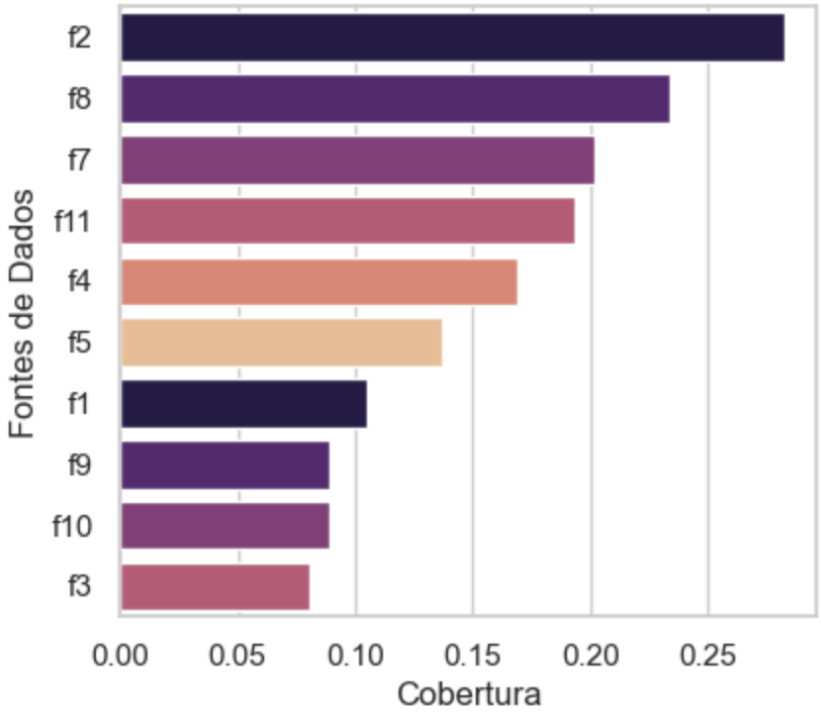
Figura 17 – 10 Fontes de dados com maior cobertura no *dataset* Restaurantes para o atributo Rua (a) e para o atributo Bairro (b).



(a) Atributo Rua (b) Atributo Bairro

Fonte: Elaborada pela autora (2023).

Figura 18 – 10 Fontes de dados com maior cobertura para o atributo bairro - Parques e Praças.



Fonte: Elaborada pela autora (2023).

ritmo modificado utilizando a base de fatos, que a precisão do algoritmo modificado em todos os experimentos se manteve mais alta. A precisão é uma métrica que mede a taxa de acertos de um algoritmo, e é calculada comparando os resultados obtidos com o *gold stadard*. Quanto maior a precisão melhor o desempenho do algoritmo (LYU et al., 2021).

Ainda analisando os resultados, no atributo bairro do *dataset* Restaurantes, enquanto o

Quadro 9 – Comparação dos resultados da Descoberta da Verdade entre o algoritmo original o Algoritmo Modificado - Cenário 1.

Algoritmo	***Precisão***
Dataset Restaurantes - Bairro	
TD original	72,1%
TD modificado	97,6%
Dataset Restaurantes - Rua	
TD original	86,7%
TD modificado	90,7%
Dataset Parques e Praças	
TD original	75,0%
TD modificado	88,2%

Fonte: Elaborada pela autora (2023).

algoritmo original teve uma precisão de 72,1%, o algoritmo modificado, teve uma precisão de 97,6%, um aumento de 25,5% no resultado da descoberta da verdade. Para o atributo rua, a precisão do algoritmo original foi de 86,7%, enquanto a do algoritmo modificado foi de 90,7%. Um aumento menor com relação ao atributo bairro, mas ainda assim o algoritmo modificado obteve um resultado superior. Além disso, no *dataset* Parques e Praças, o algoritmo original teve 75,0% de precisão, e o modificado teve 88,2%, um aumento de 13% na precisão do resultado.

Ou seja, no Cenário 1, em ambos os *datasets* os fatos tiveram grande impacto no processo, e beneficiaram os resultados do algoritmo, melhorando a precisão do resultado final em ambos os casos. Com isso podemos afirmar que utilizar o conhecimento adicional da descoberta da verdade melhora os resultados do processo, especialmente em cenários de dados *long-tail* com alta completude nos atributos.

Cenário 2. Neste cenário a completude dos atributos é menor em ambos os *datasets*, como pode-se ver nos Quadros 10 e 11. Como o número de valores faltantes neste cenário é maior, a cobertura das fontes também se difere do Cenário 1.

Na Figura 19, pode-se verificar a cobertura das fontes para os atributos rua e bairro do *dataset* Restaurantes no cenário 2. É possível perceber que as fontes que mais proveem valores para os dois atributos se diferem. Enquanto para o atributo bairro a fonte que cobre mais entidades é a $fonte1$, para o atributo rua são as $fontes7$ e 10. Na Figura 20 é possível visualizar a cobertura das fontes para o atributo bairro do *dataset* Parques e Praças.

No cenário 2, como pode-se observar no Quadro 12, para os resultados do atributo bairro do

Quadro 10 – Completude dos atributos *dataset* Restaurantes - Cenário 2.

Atributos	Completude
id	1
fonte	1
nome	1
rua	0.73
numero	0.54
bairro	0.72
cidade	0.31
cep	0.52

Fonte: Elaborada pela autora (2023).

Quadro 11 – Completude dos atributos - cenário 2 - *dataset* Parques e Praças.

Atributos	Completude
id	1
fonte	1
nome	0,83
codigoBairro	0,71
bairro	0,76
cep	0,62

Fonte: Elaborada pela autora (2023).

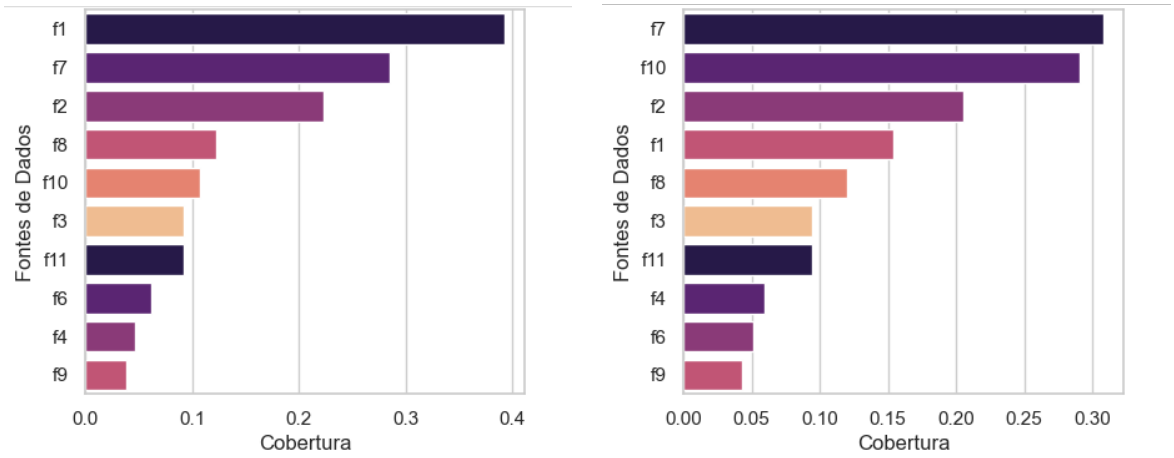
Quadro 12 – Comparação dos resultados da Descoberta da Verdade entre o algoritmo original o Algoritmo Modificado - Cenário 2.

Algoritmo	***Precisão***
Dataset Restaurantes - Bairro	
TD original	69,3%
TD modificado	93,8%
Dataset Restaurantes - Rua	
TD original	85,1%
TD modificado	91,8%
Dataset Parques e Praças	
TD original	73,1%
TD modificado	92,5%

Fonte: Elaborada pela autora (2023).

dataset Restaurantes, tanto o algoritmo original quanto com o algoritmo modificado tiveram uma queda na precisão. No entanto, em cenários com maior quantidade de dados ausentes

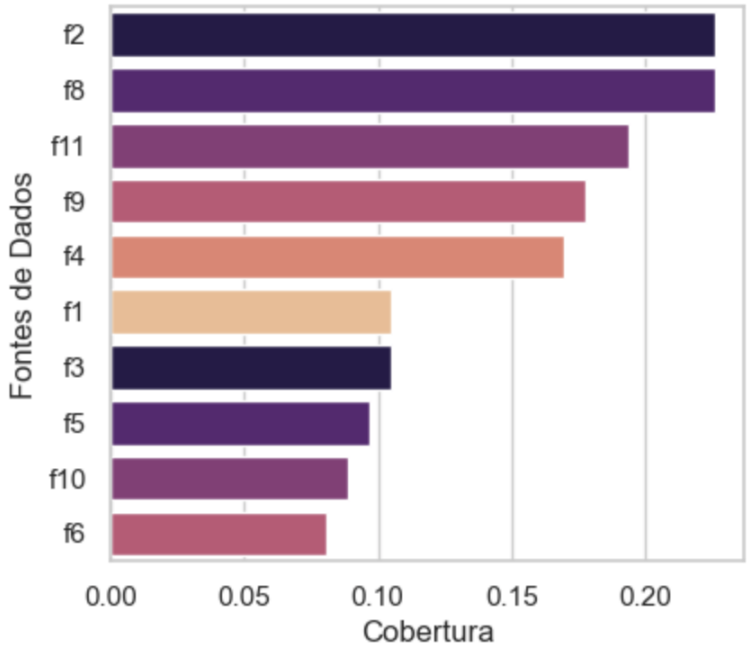
Figura 19 – 10 Fontes de dados com maior cobertura no *dataset* Restaurantes para o atributo Bairro (a) e para o atributo Rua (b).



(a) Atributo Rua (b) Atributo Bairro

Fonte: Elaborada pela autora (2023).

Figura 20 – 10 Fontes de dados com maior cobertura no *dataset* Parques e Praças para o atributo bairro - Cenário 2



Fonte: Elaborada pela autora (2023).

como é o caso, o algoritmo modificado teve um desempenho muito superior ao algoritmo original. Esse fato pode ser comprovado analisando o resultado no mesmo *dataset*, mas para o atributo rua, em que o algoritmo original se manteve com precisão inferior. Além disso, para o *dataset* Parques e Praças, o resultado foi bem semelhante. O algoritmo modificado teve uma precisão de 92,5% contra 73,1% do atributo original. Pode-se notar um aumento significativo, de 19,4% na precisão do resultado.

Tanto no cenário 1 como no cenário 2, o resultado da descoberta da verdade com o algoritmo modificado se manteve superior. Esse dado segue corroborando com a hipótese levantada neste trabalho. Assim, com esse experimento pode-se notar o melhor desempenho do algoritmo modificado que utiliza os fatos como informação adicional na descoberta da verdade.

5.3 CONSIDERAÇÕES

Neste capítulo foram apresentados os experimentos realizados para avaliação da nossa proposta. Inicialmente foi explicado o Experimento 1, como se deu o desenvolvimento e os resultados. Depois, apresentamos o Experimento 2, as ferramentas utilizadas em seu desenvolvimento, como foi realizada a codificação, qual conjunto de dados foi utilizado e manipulado, e os resultados.

O experimento 2 foi conduzido utilizando *datasets* distintos, criando-se dois cenários de dados. Ambos os cenários se enquadram no fenômeno *long-tail*, conforme comprovado pela média de cobertura das fontes de dados. No cenário 1, foi considerada uma alta completude dos atributos, enquanto no cenário 2 a completude foi menor. O objetivo era compreender o desempenho da solução proposta em diferentes cenários de completude de dados. Os resultados revelaram que, em ambos os cenários, o algoritmo modificado utilizando a solução proposta neste trabalho se manteve superior em níveis de precisão.

Este resultado nos faz acreditar que a utilização da base de fatos como conhecimento adicional é benéfica para o processo de descoberta da verdade. Portanto, podemos confirmar a hipótese levantada neste trabalho.

No próximo capítulo serão apresentadas as conclusões referentes a este trabalho.

6 CONCLUSÕES

Neste trabalho, foi proposta a utilização de relacionamentos entre atributos para gerar uma base de fatos a ser considerada como conhecimento adicional na etapa de Descoberta da Verdade no processo de Fusão de Dados.

Inicialmente, foi realizado um extenso levantamento do estado da arte, categorizando os trabalhos de fusão de dados e descoberta da verdade. Esse levantamento culminou em um *survey*, publicado no *Journal of Intelligent Information Systems - 2021* (CANALLE; SALGADO; LóSCIO, 2021).

Em seguida, foi apresentada a especificação detalhada da proposta, incluindo definições principais, a formalização do problema, e a arquitetura da abordagem. Além disso, foi apresentado um exemplo ilustrativo para facilitar a compreensão. Esta proposta gerou um artigo que foi apresentado no *Workshop de Teses e Dissertações* do *Simpósio Brasileiro de Banco de Dados - 2020*.

Experimentos foram executados para avaliação da proposta e consequentemente, comprovação da hipótese levantada no Capítulo 1. No experimento principal os resultados do algoritmo que implementa a proposta desta tese foram comparados ao resultado do algoritmo original. Foi constatado que os resultados obtidos com o algoritmo modificado apresentaram uma maior precisão em relação ao algoritmo original.

Com isso, foi possível comprovar a hipótese formulada. Os resultados obtidos nos experimentos fornecem evidências suficientes de que a utilização de relacionamentos entre atributos na etapa de Descoberta da Verdade pode melhorar a precisão dos resultados. Essa abordagem pode proporcionar benefícios em diferentes domínios de aplicação.

6.1 LIMITAÇÕES

Para este trabalho, podemos elencar as seguintes limitações:

- Um bom resultado no processo de Descoberta da Verdade vai depender da qualidade dos fatos que serão utilizados no processo. Deste modo, é necessário definir fatos confiáveis para gerar a base de fatos, ou o processo pode, inclusive, ser atrapalhado por fatos errados (ou desatualizados);
- Existem poucos conjuntos de dados disponíveis para a área de descoberta da verdade,

com *gold standard*. O desafio principal, é encontrar conjuntos de dados em que hajam relacionamentos entre os atributos, necessários para avaliar a abordagem proposta neste trabalho. Por este motivo se deu a limitação de testes;

- A necessidade de um especialista de domínio;
- Falta de avaliação da solução proposta em cenários de grandes volumes de dados, analisando a performance da abordagem.

6.2 TRABALHOS FUTUROS

Como trabalhos futuros podemos citar:

- Realização de experimentos em outros domínios de dados;
- Comparação com outros algoritmos do estado da arte, de preferência algoritmos elencados na seção 3.5, que tem como característica, a descoberta da verdade por meio do uso de relacionamentos;
- Estudos sobre como realizar o gerenciamento da base de fatos.

REFERÊNCIAS

AHMED, R.; SMEDT, P. D.; DU, W.; KENT, W.; KETABCHI, M. A.; LITWIN, W.; RAFII, A.; SHAN, M.-C. The pegasus heterogeneous multidatabase system. *IEEE Computer*, v. 24, n. 12, p. 19–27, 1991. Disponível em: <http://dblp.uni-trier.de/db/journals/computer/computer24.html#AhmedSDKKLRS91>.

ARENS, Y.; KNOBLOCK, C. A.; SHEN, W.-M. Query reformulation for dynamic information integration. *J. Intell. Inf. Syst.*, v. 6, n. 2/3, p. 99–130, 1996. Disponível em: <http://dblp.uni-trier.de/db/journals/jiis/jiis6.html#ArensKS96>.

BERETTA, V.; HARISPE, S.; RANWEZ, S.; MOUGENOT, I. How can ontologies give you clue for truth-discovery? an exploratory study. In: AKERKAR, R.; PLANTIé, M.; RANWEZ, S.; HARISPE, S.; LAURENT, A.; BELLOT, P.; MONTMAIN, J.; TROUSSET, F. (Ed.). *WIMS*. ACM, 2016. p. 15:1–15:12. ISBN 978-1-4503-4056-4. Disponível em: <http://dblp.uni-trier.de/db/conf/wims/wims2016.html#BerettaHRM16>.

BERETTA, V.; HARISPE, S.; RANWEZ, S.; MOUGENOT, I. Truth selection for truth discovery models exploiting ordering relationship among values. *Knowl.-Based Syst.*, v. 159, p. 298–308, 2018. Disponível em: <http://dblp.uni-trier.de/db/journals/kbs/kbs159.html#BerettaHRM18>.

BERGAMASCHI, S.; BENEVENTANO, D.; MANDREOLI, F.; MARTOGLIA, R.; GUERRA, F.; ORSINI, M.; PO, L.; VINCINI, M.; SIMONINI, G.; ZHU, S.; GAGLIARDELLI, L.; MAGNOTTA, L. From data integration to big data integration. In: _____. *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years*. Cham: Springer International Publishing, 2018. p. 43–59. ISBN 978-3-319-61893-7. Disponível em: <https://doi.org/10.1007/978-3-319-61893-7_3>.

BERTI-ÉQUILLE, L. Truth discovery. In: _____. *Encyclopedia of Big Data Technologies*. Cham: Springer International Publishing, 2018. p. 1–8. ISBN 978-3-319-63962-8. Disponível em: <https://doi.org/10.1007/978-3-319-63962-8_23-1>.

BERTI-ÉQUILLE, L.; BORGE-HOLTHOEFER, J. *Veracity of Data: From Truth Discovery Computation Algorithms to Models of Misinformation Dynamics*. [S.l.]: Morgan & Claypool Publishers, 2015. (Synthesis Lectures on Data Management).

BILKE, A.; BLEIHOLDER, J.; BÖHM, C.; DRABA, K.; NAUMANN, F.; WEIS, M. Automatic data fusion with HumMer. In: *VLDB, Demo Abstract Band*. [s.n.], 2005. ISBN 1-59593-154-6. Disponível em: <http://www.informatik.hu-berlin.de/mac/publications/VLDB2005.pdf>.

BLEIHOLDER, J.; NAUMANN, F. Conflict handling strategies in an integrated information system. In: *IJCAI Workshop on Information on the Web (IIWeb)*. [S.l.: s.n.], 2006.

BLEIHOLDER, J.; NAUMANN, F. Data fusion. *ACM Computational Surveys*, ACM, New York, NY, USA, v. 41, n. 1, p. 1–41, 2008. ISSN 0360-0300.

BROELEMANN, K.; GOTTRON, T.; KASNECI, G. Ltd-rbm: Robust and fast latent truth discovery using restricted boltzmann machines. In: *ICDE*. IEEE Computer Society, 2017. p. 143–146. ISBN 978-1-5090-6543-1. Disponível em: <http://dblp.uni-trier.de/db/conf/icde/icde2017.html#BroelemannGK17>.

BROELEMANN, K.; GOTTRON, T.; KASNECI, G. Restricted boltzmann machines for robust and fast latent truth discovery. *CoRR*, abs/1801.00283, 2017. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1801.html#abs-1801-00283>.

BROELEMANN, K.; KASNECI, G. Combining restricted boltzmann machines with neural networks for latent truth discovery. *CoRR*, abs/1807.10680, 2018. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1807.html#abs-1807-10680>.

CANALLE, G.; SALGADO, A. C.; LóSCIO, B. A survey on data fusion: what for? in what form? what is next? *Journal of Intelligent Information Systems*, v. 57, p. 1–26, 08 2021.

CECCHIN, F.; CIFERRI, C. D. de A.; HARA, C. S. Xml data fusion. In: *DaWak*. Springer, 2010. (Lecture Notes in Computer Science, v. 6263), p. 297–308. Disponível em: <http://dblp.uni-trier.de/db/conf/dawak/dawak2010.html#CecchinCH10>.

CHOMICKI, J.; MARCINKOWSKI, J.; STAWORKO, S. Computing consistent query answers using conflict hypergraphs. In: *CIKM*. ACM, 2004. p. 417–426. Disponível em: <http://dblp.uni-trier.de/db/conf/cikm/cikm2004.html#ChomickiMS04>.

CHOMICKI, J.; MARCINKOWSKI, J.; STAWORKO, S. Hippo: A system for computing consistent answers to a class of sql queries. In: *EDBT*. Springer, 2004. (Lecture Notes in Computer Science, v. 2992), p. 841–844. Disponível em: <http://dblp.uni-trier.de/db/conf/edbt/edbt2004.html#ChomickiMS04>.

COLLET, C.; HUHNS, M. N.; SHEN, W.-M. Resource integration using a large knowledge base in carnot. *IEEE Computer*, v. 24, n. 12, p. 55–62, 1991. Disponível em: <http://dblp.uni-trier.de/db/journals/computer/computer24.html#ColletHS91>.

CRESWELL, J. W. Projeto de pesquisa métodos qualitativo, quantitativo e misto. In: *Projeto de pesquisa métodos qualitativo, quantitativo e misto*. [S.l.]: Artmed, 2010.

DONG, X. L.; BERTI-ÉQUILLE, L.; SRIVASTAVA, D. Integrating conflicting data: The role of source dependence. *PVLDB*, v. 2, n. 1, p. 550–561, 2009. Disponível em: <http://dblp.uni-trier.de/db/journals/pvldb/pvldb2.html#DongBS09>.

DONG, X. L.; BERTI-ÉQUILLE, L.; SRIVASTAVA, D. Truth discovery and copying detection in a dynamic world. *PVLDB*, v. 2, n. 1, p. 562–573, 2009. Disponível em: <http://dblp.uni-trier.de/db/journals/pvldb/pvldb2.html#DongBS09a>.

DONG, X. L.; GABRILOVICH, E.; HEITZ, G.; HORN, W.; MURPHY, K.; SUN, S.; ZHANG, W. From data fusion to knowledge fusion. *PVLDB*, v. 7, n. 10, p. 881–892, 2014.

DONG, X. L.; NAUMANN, F. Data fusion - resolving data conflicts for integration. *PVLDB*, v. 2, n. 2, p. 1654–1655, 2009. Disponível em: <http://dblp.uni-trier.de/db/journals/pvldb/pvldb2.html#DongN09>.

DONG, X. L.; SRIVASTAVA, D. Big data integration. In: *2013 IEEE 29th International Conference on Data Engineering (ICDE)*. [S.l.: s.n.], 2013. p. 1245–1248.

DONG, X. L.; SRIVASTAVA, D. *Big Data Integration*. Morgan & Claypool Publishers, 2015. 1-198 p. (Synthesis Lectures on Data Management). Disponível em: <http://dx.doi.org/10.2200/S00578ED1V01Y201404DTM040>.

DONG, X. L.; SRIVASTAVA, D. Data fusion. In: LIU, L.; ÖZSU, M. T. (Ed.). *Encyclopedia of Database Systems (2nd ed.)*. Springer, 2018. Disponível em: <http://dblp.uni-trier.de/db/reference/db/d2.html#DongS18>.

DRAPER, D.; HALEVY, A. Y.; WELD, D. S. The nimble integration engine. In: MEHROTRA, S.; SELLIS, T. K. (Ed.). *SIGMOD Conference*. ACM, 2001. p. 567–568. Disponível em: <http://dblp.uni-trier.de/db/conf/sigmod/sigmod2001.html#DraperHW01>.

ELMAGARMID, A. K.; IPEIROTIS, P. G.; VERYKIOS, V. S. Duplicate record detection: A survey. *IEEE TKDE*, IEEE Computer Society, Los Alamitos, CA, USA, v. 19, n. 1, p. 1–16, 2007. ISSN 1041-4347.

ERL, T.; KHATTAK, W.; BUHLER, P. *Big Data Fundamentals*. [S.l.]: Prentice Hall: Upper Saddle River, NJ, USA, 2016.

FANG, X. S. Truth discovery from conflicting multi-valued objects. In: BARRETT, R.; CUMMINGS, R.; AGICHTEIN, E.; GABRILOVICH, E. (Ed.). *WWW (Companion Volume)*. ACM, 2017. p. 711–715. Disponível em: <http://dblp.uni-trier.de/db/conf/www/www2017c.html#Fang17>.

FANG, X. S.; SHENG, Q. Z.; WANG, X.; NGU, A. H. H. Smartmtd: A graph-based approach for effective multi-truth discovery. *CoRR*, abs/1708.02018, 2017. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1708.html#abs-1708-02018>.

FUXMAN, A.; FAZLI, E.; MILLER, R. J. Conquer: efficient management of inconsistent databases. In: *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. New York, NY, USA: ACM, 2005. p. 155–166. Disponível em: <http://www.cs.toronto.edu/~afuxman/publications/sigmod05.pdf>.

GALLAND, A.; ABITEBOUL, S.; MARIAN, A.; SENELLART, P. Corroborating information from disagreeing views. In: *WSDM*. ACM, 2010. p. 131–140. Disponível em: <http://dblp.uni-trier.de/db/conf/wsdm/wsdm2010.html#GallandAMS10>.

GOLSHAN, B.; HALEVY, A. Y.; MIHAILA, G. A.; TAN, W.-C. Data integration: After the teenage years. In: SALLINGER, E.; BUSSCHE, J. V. den; GEERTS, F. (Ed.). *PODS*. ACM, 2017. p. 101–106. ISBN 978-1-4503-4198-1. Disponível em: <http://dblp.uni-trier.de/db/conf/pods/pods2017.html#GolshanHMT17>.

HARA, C. S.; CIFERRI, C. D. de A.; CIFERRI, R. R. Incremental data fusion based on provenance information. In: *In Search of Elegance in the Theory and Practice of Computation*. Springer, 2013. (Lecture Notes in Computer Science, v. 8000), p. 339–365. Disponível em: <http://dblp.uni-trier.de/db/conf/birthday/buneman2013.html#HaraCC13>.

HUHTALA, Y.; KäRKKäINEN, J.; PORKKA, P.; TOIVONEN, H. Tane: An efficient algorithm for discovering functional and approximate dependencies. *Comput. J.*, v. 42, n. 2, p. 100–111, 1999. Disponível em: <http://dblp.uni-trier.de/db/journals/cj/cj42.html#HuhtalaKPT99>.

JARADAT, A.; SAFIEDDINE, F.; DERAMAN, A.; ALI, O.; AL-AHMAD, A.; ALZOUBI, Y. I. A probabilistic data fusion modeling approach for extracting true values from uncertain and conflicting attributes. *Big Data and Cognitive Computing*, v. 6, n. 4, 2022. ISSN 2504-2289. Disponível em: <https://www.mdpi.com/2504-2289/6/4/114>.

KLEINBERG, J. M. Authoritative sources in a hyperlinked environment. *JACM*, v. 46, n. 5, p. 604–632, set. 1999.

KNOBLOCK, C. A.; MINTON, S.; AMBITE, J. L.; ASHISH, N.; MUSLEA, I.; PHILPOT, A.; TEJADA, S. The ariadne approach to web-based information integration. In: *International Journal of Cooperative Information Systems*. [S.l.: s.n.], 2001. v. 10(1-2), p. 145–169.

LENZERINI, M. Data integration: A theoretical perspective. In: *Proc. of the 21st ACM SIGMOD-SIGART Symposium on Principles of Database*. [s.n.], 2002. Disponível em: <http://portal.acm.org/citation.cfm?id=543644>.

LI, F.; DONG, X. L.; LANGEN, A.; LI, Y. Discovering multiple truths with a hybrid model. *CoRR*, abs/1705.04915, 2017. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1705.html#LiDLL17>.

LI, Q.; LI, Y.; GAO, J.; SU, L.; ZHAO, B.; DEMIRBAS, M.; FAN, W.; HAN, J. A confidence-aware approach for truth discovery on long-tail data. *PVLDB*, v. 8, n. 4, p. 425–436, 2014. Disponível em: <http://dblp.uni-trier.de/db/journals/pvldb/pvldb8.html#LiLGSZDFH14>.

LI, Q.; LI, Y.; GAO, J.; ZHAO, B.; FAN, W.; HAN, J. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In: *SIGMOD Conference*. ACM, 2014. p. 1187–1198. Disponível em: <http://dblp.uni-trier.de/db/conf/sigmod/sigmod2014.html#LiLGZFH14>.

LI, X.; DONG, X. L.; LYONS, K.; MENG, W.; SRIVASTAVA, D. Truth finding on the deep web: Is the problem solved? *CoRR*, abs/1503.00303, 2015. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1503.html#LiDLMS15>.

LI, Y.; GAO, J.; MENG, C.; LI, Q.; SU, L.; ZHAO, B.; FAN, W.; HAN, J. A survey on truth discovery. *SIGKDD Explorations*, v. 17, n. 2, p. 1–16, 2015. Disponível em: <http://dblp.uni-trier.de/db/journals/sigkdd/sigkdd17.html#LiGMLSZFH15>.

LI, Y.; LI, Q.; GAO, J.; SU, L.; ZHAO, B.; FAN, W.; HAN, J. On the discovery of evolving truth. In: *KDD*. ACM, 2015. p. 675–684. Disponível em: <http://dblp.uni-trier.de/db/conf/kdd/kdd2015.html#LiLGSZFH15>.

LI, Y.; LI, Q.; GAO, J.; SU, L.; ZHAO, B.; FAN, W.; HAN, J. Conflicts to harmony: A framework for resolving conflicts in heterogeneous data by truth discovery. *IEEE TKDE*, v. 28, n. 8, p. 1986–1999, 2016. Disponível em: <http://dblp.uni-trier.de/db/journals/tkde/tkde28.html#LiLGSZFH16>.

LIU, X.; DONG, X. L.; OOI, B. C.; SRIVASTAVA, D. Online data fusion. *PVLDB*, v. 4, n. 11, p. 932–943, 2011. Disponível em: <http://dblp.uni-trier.de/db/journals/pvldb/pvldb4.html#LiuDOS11>.

LYU, S.; OUYANG, W.; WANG, Y.; SHEN, H.; CHENG, X. Truth discovery by claim and source embedding. *IEEE Transactions on Knowledge and Data Engineering*, v. 33, n. 3, p. 1264–1275, 2021.

MIRZA, A.; SIDDIQI, I. Data level conflicts resolution for multi-sources heterogeneous databases. In: *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*. [S.l.: s.n.], 2016. p. 36–40.

MOTRO, A.; ANOKHIN, P. Fusionplex: resolution of data inconsistencies in the integration of heterogeneous information sources. *Information Fusion*, v. 7, n. 2, p. 176–196, 2006. Disponível em: <http://dblp.uni-trier.de/db/journals/inffus/inffus7.html#MotroA06>.

MOTRO, A.; ANOKHIN, P.; ACAR, A. C. Utility-based resolution of data inconsistencies. In: *IQIS*. ACM, 2004. p. 35–43. Disponível em: <http://dblp.uni-trier.de/db/conf/iqis/iqis2004.html#MotroAA04>.

NAKHAEI, Z.; AHMADI, A. Toward high level data fusion for conflict resolution. In: *ICMLC*. IEEE, 2017. p. 91–97. Disponível em: <http://dblp.uni-trier.de/db/conf/icmlc/icmlc2017.html#NakhaeiA17>.

PAPOTTI, P.; SANTORO, D. Data integration. In: _____. *Encyclopedia of Big Data Technologies*. Cham: Springer International Publishing, 2018. p. 1–6. ISBN 978-3-319-63962-8. Disponível em: <https://doi.org/10.1007/978-3-319-63962-8_6-1>.

PASTERNACK, J.; ROTH, D. Knowing what to believe (when you already know something). In: *COLING*. Tsinghua University Press, 2010. p. 877–885. Disponível em: <http://dblp.uni-trier.de/db/conf/coling/coling2010.html#PasternackR10>.

PASTERNACK, J.; ROTH, D. Latent credibility analysis. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. *Proceedings of the 22nd international conference on World Wide Web*. 2013. p. 1009–1020. Disponível em: <http://www2013.org/proceedings/p1009.pdf>.

POCHAMPALLY, R.; SARMA, A. D.; DONG, X. L.; MELIOU, A.; SRIVASTAVA, D. Fusing data with correlations. In: *SIGMOD Conference*. ACM, 2014. p. 433–444. Disponível em: <http://dblp.uni-trier.de/db/conf/sigmod/sigmod2014.html#PochampallySDMS14>.

PRADHAN, R.; AREF, W. G.; PRABHAKAR, S. Leveraging data relationships to resolve conflicts from disparate data sources. In: HARTMANN, S.; MA, H.; HAMEURLAIN, A.; PERNUL, G.; WAGNER, R. R. (Ed.). *DEXA (2)*. Springer, 2018. (Lecture Notes in Computer Science, v. 11030), p. 99–115. ISBN 978-3-319-98812-2. Disponível em: <http://dblp.uni-trier.de/db/conf/dexa/dexa2018-2.html#PradhanAP18>.

RAHM, E.; BERNSTEIN, P. A. A survey of approaches to automatic schema matching. *The VLDB Journal*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 10, n. 4, p. 334–350, 2001. Disponível em: <http://dx.doi.org/10.1007/s007780100057>.

REKATSINAS, T.; JOGLEKAR, M.; GARCIA-MOLINA, H.; PARAMESWARAN, A. G.; Ré, C. Slimfast: Guaranteed results for data fusion and source reliability. In: *SIGMOD Conference*. ACM, 2017. p. 1399–1414. Disponível em: <http://dblp.uni-trier.de/db/conf/sigmod/sigmod2017.html#RekatsinasJGPR17>.

SCHALLEHN, E. *Efficient similarity-based operations for data integration*. Tese (Doutorado) — Uni Magdeburg, 2004.

SUBRAHMANIAN, V. d.; ADALI, S.; BRINK, A.; EMERY, R.; LU, J. J.; RAJPUT, A.; ROGERS, T. J.; ROSS, R.; WARD, C. *HERMES: A heterogeneous reasoning and mediator system*. [S.l.]: Technical report, University of Maryland, 1995.

VIEIRA, P. K. M.; LÓSCIO, B. F.; SALGADO, A. C. Incremental entity resolution process over query results for data integration systems. *J. Intell. Inf. Syst.*, v. 52, n. 2, p. 451–471, 2019. Disponível em: <https://doi.org/10.1007/s10844-019-00544-1>.

WANG, Y.; MA, F.; SU, L.; GAO, J. Discovering truths from distributed data. In: *ICDM*. IEEE Computer Society, 2017. p. 505–514. Disponível em: <http://dblp.uni-trier.de/db/conf/icdm/icdm2017.html#WangMSG17>.

XIAO, H.; GAO, J.; LI, Q.; MA, F.; SU, L.; FENG, Y.; ZHANG, A. Towards confidence in the truth: A bootstrapping based truth discovery approach. In: *KDD*. ACM, 2016. p. 1935–1944. Disponível em: <http://dblp.uni-trier.de/db/conf/kdd/kdd2016.html#XiaoGLMSFZ16>.

YIN, X.; HAN, J.; YU, P. S. Truth discovery with multiple conflicting information providers on the web. *IEEE TKDE*, v. 20, n. 6, p. 796–808, 2008. Disponível em: <http://dblp.uni-trier.de/db/journals/tkde/tkde20.html#YinHY08>.

YIN, X.; TAN, W. Semi-supervised truth discovery. In: *WWW*. ACM, 2011. p. 217–226. Disponível em: <http://dblp.uni-trier.de/db/conf/www/www2011.html#YinT11>.

ZHANG, H.; LI, Q.; MA, F.; XIAO, H.; LI, Y.; GAO, J.; SU, L. Influence-aware truth discovery. In: *CIKM*. ACM, 2016. p. 851–860. Disponível em: <http://dblp.uni-trier.de/db/conf/cikm/cikm2016.html#ZhangLMXLGS16>.

ZHANG, J.; WANG, S.; WU, G.; ZHANG, L. A effective truth discovery algorithm with multi-source sparse data. In: SHI, Y.; FU, H.; TIAN, Y.; KRZHIZHANOVSKAYA, V. V.; LEES, M. H.; DONGARRA, J. J.; SLOOT, P. M. A. (Ed.). *ICCS (3)*. Springer, 2018. (Lecture Notes in Computer Science, v. 10862), p. 434–442. ISBN 978-3-319-93713-7. Disponível em: <http://dblp.uni-trier.de/db/conf/iccS/iccS2018-3.html#ZhangWWZ18>.

ZHAO, B.; RUBINSTEIN, B. I. P.; GEMMELL, J.; HAN, J. A bayesian approach to discovering truth from conflicting sources for data integration. *CoRR*, abs/1203.0058, 2012. Disponível em: <http://dblp.uni-trier.de/db/journals/corr/corr1203.html#abs-1203-0058>.

ZIKOPOULOS, P. C.; EATON, C.; DEROOS, D.; DEUTSCH, T.; LAPIS, G. *Understanding Big Data - Analytics for Enterprise Class Hadoop and Streaming Data*. [S.l.: s.n.], 2012.

APÊNDICE A – RESULTADO TANE

Dataset abalone

```
List of all FDs:  [['EFH', 'A'], ['EFH', 'C'], ['EFH', 'B'], ['EFH', 'D'], ['EFH'
    , 'G'], ['EFH', 'I'], ['EGH', 'A'], ['EGH', 'C'], ['EGH', 'B'], ['EGH', 'D'],
     ['EGH', 'F'], ['EGH', 'I'], ['CEF', 'A'], ['CEG', 'A'], ['EFG', 'A'], ['BEF'
    , 'I'], ['EFG', 'C'], ['CEF', 'G'], ['ABEF', 'C'], ['ABEF', 'D'], ['ABEF', 'G
    '], ['ABEF', 'H'], ['AEGI', 'C'], ['AEGI', 'B'], ['AEGI', 'D'], ['AEGI', 'F'
    ], ['AEGI', 'H'], ['BCEF', 'D'], ['BCEF', 'H'], ['BCEH', 'A'], ['BCEH', 'D'],
     ['BCEH', 'G'], ['BCEH', 'F'], ['BCEH', 'I'], ['BDEF', 'A'], ['BDEF', 'C'], [
    'BDEF', 'G'], ['BDEF', 'H'], ['BDEG', 'A'], ['BDEG', 'C'], ['BDEG', 'F'], ['
    BDEG', 'I'], ['BDEG', 'H'], ['BEFG', 'D'], ['BEFG', 'H'], ['BEGI', 'A'], ['
    BEGI', 'C'], ['BEGI', 'D'], ['BEGI', 'F'], ['BEGI', 'H'], ['BFGI', 'A'], ['
    BFGI', 'C'], ['BFGI', 'E'], ['BFGI', 'D'], ['BFGI', 'H'], ['CDEF', 'B'], ['
    CDEF', 'I'], ['CDEF', 'H'], ['CEFI', 'B'], ['CEFI', 'D'], ['CEFI', 'H'], ['
    CEGI', 'B'], ['CEGI', 'D'], ['CEGI', 'F'], ['CEGI', 'H'], ['DEFG', 'B'], ['
    DEFG', 'I'], ['DEFG', 'H'], ['DFGH', 'A'], ['DFGH', 'C'], ['DFGH', 'B'], ['
    DFGH', 'E'], ['DFGH', 'I'], ['DFGI', 'A'], ['DFGI', 'C'], ['DFGI', 'B'], ['
    DFGI', 'E'], ['DFGI', 'H'], ['EFGI', 'B'], ['EFGI', 'D'], ['EFGI', 'H'], ['
    BCDE', 'A'], ['ABFG', 'C'], ['BDEI', 'A'], ['BEHI', 'A'], ['BFGH', 'A'], ['
    ABFG', 'H'], ['ADEG', 'C'], ['CDEH', 'A'], ['CFGH', 'A'], ['CFGI', 'A'], ['
    AFGI', 'C'], ['AFGI', 'H'], ['CDFG', 'B'], ['BDFG', 'C'], ['BFGH', 'C'], ['
    CFGI', 'H'], ['ABDFG', 'E'], ['ABDFG', 'I'], ['ABGHI', 'C'], ['ABGHI', 'E'],
    ['ABGHI', 'D'], ['ABGHI', 'F'], ['ACDFG', 'E'], ['ACDFG', 'I'], ['ACDFG', 'H'
    ], ['BCDEI', 'G'], ['BCDEI', 'F'], ['BCDEI', 'H'], ['BCDGI', 'A'], ['BCDGI',
    'E'], ['BCDGI', 'F'], ['BCDGI', 'H'], ['BCGHI', 'A'], ['BCGHI', 'E'], ['BCGHI
    ', 'D'], ['BCGHI', 'F'], ['CDEHI', 'B'], ['CDEHI', 'G'], ['CDEHI', 'F'], ['
    ABCGH', 'F'], ['BCFHI', 'A'], ['ADEHI', 'B'], ['ABDEH', 'I'], ['BDFHI', 'A'],
     ['ABDFH', 'I'], ['CDFHI', 'A'], ['CDGHI', 'A'], ['ACDGH', 'I'], ['ABCDFH', '
    E'], ['ABCDFH', 'G'], ['ABCDFI', 'E'], ['ABCDFI', 'G'], ['ABCDFI', 'H'], ['
    ABCDGH', 'E'], ['BCDFHI', 'E'], ['BCDFHI', 'G']]
Total number of FDs found:  137
```

APÊNDICE B – CODIFICAÇÃO

As partes principais da codificação do algoritmo de descoberta da verdade podem ser vistas abaixo. Os principais parâmetros foram inicializado, de acordo com a publicação original.

- Dampening factor: 0,3
- Relatedness factor: 0,5
- Confiabilidade inicial das fontes: 0.9
- Função de similaridade: cosine
- Similaridade: 0,5
- Máximo de iterações: 10
- Tolerância: 0.001

```
def __init__(self, df, df_fato, df_truth, fact, obj, implication = None,
    initial_trust = 0.9, dampening_factor = 0.3, relatedness_factor = 0.5,
    base_sim = 0.5):
        self.df = df
        self.fact = fact
        self.object = obj
        self.df_fato = df_fato
        self.df_truth = df_truth

        if implication==None:
            self.implication = cosine_sim
        else:
            self.implication = implication

        self.initial_trust = initial_trust
        self.dampening_factor = dampening_factor
        self.relatedness_factor = relatedness_factor
        self.base_sim = base_sim
```

No trecho de código a seguir, pode-se visualizar o core do algoritmo. O modelo itera entre as funções $self.compute_fact_confidence()$ e $self.compute_source_trust()$ até a convergência. O retorno é um conjunto de entidades com seus valores verdadeiros para um dado atributo.

```
    def compute(self, max_it = 10, tolerance = 0.001, progress = False):
```

```
        self.df['trust'] = self.initial_trust
        self.df['confidence'] = 0.0
        self.df['trust_new'] = 0.0

        for i in range(max_it):
            t1 = self.df.drop_duplicates("source")["trust_new"]

            self.compute_fact_confidence()
            self.compute_source_trust()

            t2 = self.df.drop_duplicates("source")["trust_new"]

            error = (t1 @ t2.T) / (np.linalg.norm(t1)*np.linalg.norm(t2))
            error = 1 - error

            if progress:
                print("Iteration: {}, Error: {}".format(i, error))

            if error > tolerance:
                break

        return self.extract_truth()
```