CFI-BLOCKING: UMA ESTRATÉGIA EFICAZ PARA BLOCAGEM EM PAREAMENTO PROBABILÍSTICO DE REGISTROS

RAMON GONÇALVES PEREIRA

CFI-BLOCKING: UMA ESTRATÉGIA EFICAZ PARA BLOCAGEM EM PAREAMENTO PROBABILÍSTICO DE REGISTROS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais - Departamento de Ciência da Computação como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Wagner Meira Junior
Coorientador: Augusto Afonso Guerra Júnior

Belo Horizonte

Março de 2016

Pereira, Ramon Gonçalves

P436c CFI-Blocking: Uma estratégia eficaz para blocagem em pareamento probabilístico de registros / Ramon Gonçalves Pereira. — Belo Horizonte, 2016
xxii, 65 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais - Departamento de Ciência da Computação

Orientador: Wagner Meira Junior
Coorientador: Augusto Afonso Guerra Júnior

1. Computação - Teses. 2. Mineração de Dados(Computação). 3. Big Data. 4. Saúde Pública - Brasil - estatística. I. Título.

CDU 519.6*73(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

CFI-Blocking: uma estratégia eficaz para blocagem em pareamento probabilístico de registros

RAMON GONÇALVES PEREIRA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. WAGNER MEIRA JÚNIOR - Orientador
Departamento de Ciência da Computação - UFMG

PROF. AUGUSTO AFONSO GUERRA JÚNIOR - Coorientador
Faculdade de Fármacia - UFMG

PROF. ADRIANO ALONSO VELOSO
Departamento de Ciência da Computação - UFMG

PROF. ALTIGRAN SOARES DA SILVA
Departamento de Ciência da Computação - UFAM

PROF. ANTONIO LUIZ PINHO RIBEIRO
Departamento de Clínica Médica - UFMG

PROF. OSVALDO SÉRGIO FARHAT DE CARVALHO
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 28 de junho de 2016.

Aos meus pais, minha namorada, meus professores e amigos, pessoas fundamentais na minha vida.

Agradecimentos

Agradeço primeiramente a Deus por ter me dado capacidade e por ter aberto portas que permitiram alcançar os meus objetivos. Agradeço ao professor Augusto Afonso Guerra Júnior por ter acreditado e confiado no meu potencial me dando as primeiras oportunidades como acadêmico e pesquisador. Agradeço ao meu orientador, professor Wagner Meira Júnior, por me ter permitido aprimorar minhas habilidades técnicas e me proporcionado desenvolvimento profissional e pessoal, me apoiando desde os momentos mais fáceis aos mais difíceis. Agradeço ao professor Mohammed J. Zaki por ter me recebido em sua universidade *Rensselaer Polytechnic Institute* para um intercâmbio acadêmico e ter elevado os níveis da minha pesquisa com sua contribuição e conhecimento aplicado.

Agradeço a Universidade Federal de Minas Gerais em dois âmbitos, no primeiro, especificamente ao Departamento de Ciência da Computação, por ter me dado recursos e suporte para a realização deste trabalho. No segundo, a Faculdade de Farmácia que como minha empregadora, autorizou e deu suporte para o meu comparecimento as aulas, reuniões e atividades extra classe de escrita da dissertação. Agradeço ao Ministério da Saúde do Brasil pelo consentimento e fornecimento dos dados utilizados para a esta pesquisa. Agradeço ao *Rensselaer Polytechnic Institute* por ter me concedido espaço, estrutura e aulas que aprimoraram este trabalho.

Agradeço aos meus amigos do CCATES, aos meus colegas do Laboratório de Software Livre(LSL) em especial ao Michel Boaventura e Guilherme Maluf por suas colaborações e esclarecimentos de dúvidas. Agradeço principalmente aqueles que passaram pelos momentos mais difíceis do mestrado ao meu lado, em especial Júlio Reis e Diego Augusto, que estiveram comigo nas tentativas frustradas de ingressar no mestrado, nos momentos de dúvidas, anseios e tomada de decisões durante esta pesquisa. Agradeço também a Michel Silva, Alberto Ueda, Danilo Boechat e Paulo Henrique Nonaka que estiveram ao meu lado desde as difíceis aulas de Projeto e Análise de Algoritmos até a correção da minha dissertação. Agradeço ao River Template time de futsal do campeonato do DCC pelos jogos e conquistas que me auxiliaram como um

momento de distração paralelo ao trabalho de dissertação.

Agradeço aos meus pais por acima de tudo terem aceitado e apoiado as minhas decisões. Agradeço a minha namorada Bárbara Rodrigues por estar ao meu lado nos momentos de estresses, por ter me dado forças durante o período de intercâmbio e por ser uma pessoa excepcional ao meu lado.

Agradeço a todos aqueles que torceram pelo meu sucesso, colaboraram de alguma forma e/ou estiveram ao meu lado ouvindo reclamações e comemorações. Sem me esquecer de ninguém, cada um tem sua parcela no sucesso desta dissertação. O meu sincero muito obrigado!

"Agradecer é um dom de reconhecer que todo seu esforço seria em vão se não fosse a colaboração coletiva das pessoas ao seu redor." (Ramon Pereira)

"Você ganha força, coragem e confiança através de cada experiência em que você realmente para e encara o medo de frente."

(Eleanor Roosevelt)

Resumo

O CFI Blocking é um algoritmo proposto para otimizar a enumeração de blocos através da mineração de padrões frequentes e do conhecimento intrínseco das instâncias dos atributos no pareamento probabilístico de registros. O pareamento de registros é um processo de integração entre bases de dados com o intuito de garantir a univocidade dos registros pertencentes a esta base de dados. Este processo pode ser executado de forma determinística ou probabilística. Em um mundo ideal o desejado é verificar registro a registro se existe outro igual a ele na base mas isso é inviável computacionalmente para grandes bases de dados, tendo um custo aproximadamente de $O(n^2)$. Para tornar o pareamento viável blocagem é responsável por pré selecionar e agrupar registros com maior probabilidade de pertencerem a mesma entidade no mundo real. As estratégias de blocagem atuais são definidas pelo conhecimento prévio do pesquisador. Neste contexto, o objetivo deste trabalho é apresentar um novo conceito de algoritmo para enumeração dos blocos no pareamento probabilístico, para eliminar essa dependência e otimizar o processo de blocagem, foram utilizadas propriedades de conjuntos fechados para enumeração automatizada dos blocos. O algoritmo foi executado em uma base de dados real de saúde pública do Brasil em uma amostra extraída por referência de localidade da região metropolitana de Belo Horizonte. Para avaliação, o *CFI Blocking* foi comparado com o método de blocagem padrão, *standard blocking* e com os conjuntos maximais. Foi possível concluir que o CFI Blocking apresenta melhor desempenho que outras abordagens existentes. »»»> 6a4ab76011e6dfc4618769d28a9206e5a9df5c17

Palavras-chave: Blocagem, Pareamento de Registros, Pareamento Probabilístico, Mineração de Dados, Dados Brasileiros.

Abstract

The CFI Blocking is an algorithm designed to optimize the blocking step through frequent pattern mining of the knowledge about attributes' instances in the probabilistic record linkage.The record linkage is a integration process between databases to try to solve the entity resolution problem. This process could be executed determistic or probabilistic. In the ideal world is desirable to compare each register with the others to check if they are the same but this is computational impraticable since that we are using big databases and this operation cost is approximately O(n^2). The blocking step allows to execute the record linkage selecting the records that are more likely to refer to the same entity in the real world.This blocks are separated to decreanse the cost to compare all registers. Current blocking strategies are based solely on analyst's knowledge. In this context, the goal of this work is present a new concept of algorithm, the CFI Blocking, to enumerate blocks in the record linkage process. CFI Blocking exploits properties of closed frequent patterns to perform an automatic enumeration of blocks. We also evaluated its performance using a real dataset from the Brazilian public health system in a sample extracted by locality of reference of the metropolitan region of Belo Horizonte. We concluded that CFI Blocking outperforms significantly other existing approaches.

Keywords: Blocking Scheme, Record Linkage, Probabilistc Record Linkage, Data Mining, Health Public Database.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

O pareamento de registros, ou *record linkage*, também conhecido por outras instâncias do problema como *the semantic integration problem* ou *the instance identification problem* (Wang & Madnick [1989]) permite encontrar registros diferentes de uma mesma entidade em bases de dados distintas como é exemplificado na figura 1.1, ou identificar registros duplicados em uma mesma bases de dados.

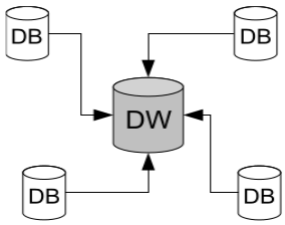


Figura 1.1. Processo de combinação de bases de dados. (dos Santos [2008])

O processo de pareamento de registros pode ser tratado de duas formas: determinístico ou probabilístico. O pareamento determinístico considera como equivalentes os registros definidos como iguais em uma determinada chave de atributos (conjunto de identificadores) (Coeli & Camargo Jr. [2002]). Esse pareamento é indicado para bancos de dados cujo é possível determinar esse conjunto de identificadores.

Na ausência desses identificadores, a tarefa se torna complexa, podendo ser utilizada uma combinação de atributos. Pode se considerar, por exemplo, equivalentes os registros que apresentarem, datas de nascimento e nomes idênticos (de Queiroz [2007]). Nesses casos, o pareamento probabilístico é o mais indicado, cuja função é classificar pares de registros, independente dos seus identificadores, levando em consideração as possibilidades de erros de preenchimento, grafia ou ocorrência de homônimos fazendo o

uso de estatística, pontuações e comparações atributo por atributo a fim de determinar duplicidades.

O pareamento probabilístico de registros teve sua formalização teórica e matemática com o trabalho de Fellegi & Sunter [1969], baseado na contribuição pioneira de Newcombe [1967]. Os registros são comparados em pares e, posteriormente, classificados em prováveis, improváveis ou duvidosos. Entretanto, existem problemas práticos que podem tornar o pareamento probabilístico ineficiente.

Na prática, uma vez que o tamanho dos bancos de dados de origem é geralmente muito grande, comparar todos os registros existentes nesta bases de dados é inviável (Nin et al. [2007]). Assim, combinar bases de dados ou cruzar informações transacionais com dados cadastrais pertencentes a diferentes origens em um modelo único tornou-se um problema difícil e importante para muitas organizações (Hernández & Stolfo [1998]). Uma solução adotada pelo pareamento de registros é recorrer a métodos de blocagem. Antes de descrevermos a estratégia de blocagem padrão, introduzimos o conceito de blocagem ótima, que é aquela que resulta na verificação apenas de pares verdadeiros. Considerando que os registros a serem pareados ou deduplicados são vértices de um grafo, isso seria equivalente a determinar uma floresta geradora, onde cada árvore geradora seria uma entidade. Assim, considerando n registros referentes a m entidades, seriam necessárias apenas $n - m + 1$ comparações. Não localizamos na literatura nenhum algoritmo que gere uma blocagem ótima, o que é explicado pela irregularidade do problema e o conceito subjetivo de similaridade entre registros, além de desafios como ruídos e valores ausentes.

Os métodos de blocagem, ou estratégias de blocagem, estão destinados a agrupar registros que apresentam um potencial de igualdade, fazendo assim com que só sejam gerados pares dentro de cada bloco, o que implica em uma real diminuição do número de pares a serem comparados, tornando o pareamento probabilístico possível computacionalmente.

Com o passar dos anos, diversas estratégias de blocagem foram adotadas no processo de pareamento de registros (Coeli & Camargo Jr. [2002]). Esforços foram feitos na tentativa de comparar essas estratégias em relação ao custo para o processamento e a quantidade de dados duplicados encontrados .

A blocagem padrão consiste em escolher alguns atributos e determinar os blocos a partir de instâncias destes atributos que sejam comuns aos registros que compõem um bloco. O problema neste caso é que nem sempre as instâncias dos atributos são discriminativas o suficiente, podendo gerar blocos muito grandes e, portanto, computacionalmente ineficientes. Essas possibilidades de erro permitem colocar a seguinte questão: Será que existe alguma maneira eficiente e rápida computacionalmente ca-

paz de gerar blocos de relevância ou blocos eficientes para a etapa de blocagem dos registros em um pareamento de registros sem a necessidade do conhecimento prévio do pesquisador?

Alguns esforços já foram colocados na realização de estratégias inteligentes de blocagem. (Nin et al. [2007]) propôs a análise sintática das informações existentes nas bases de dados para gerar os blocos. Entretanto existem peculiaridades nas bases de dados brasileiras, bem como nos algoritmos de blocagem existentes que permitem um estudo mais aprofundado em alguns pontos importantes no processo de enumeração destes blocos. Neste caso, é interessante buscar uma estratégia de blocagem que se aproximasse da blocagem ótima, o que é alcançado neste trabalho pela determinação de blocos a partir de conjuntos fechados.

Um conjunto fechado, no contexto deste trabalho, é um conjunto de registros que satisfaz um predicado conjuntivo de instâncias de atributos. Este predicado tem a propriedade de ser o fechamento em relação a todos os predicados associados ao conjunto de registros mencionado, sendo então a descrição mais precisa e restritiva destes registros. Cada conjunto fechado define um bloco, que verifica, através da comparação de pares de registros, quais se referem a uma mesma entidade.

Em suma, esta dissertação apresenta as seguintes contribuições:

- (i) uma estratégia de blocagem baseada em conjuntos fechados;
- (ii) a integração da nova estratégia ao algoritmo de pareamento de registros;
- (iii) validação utilizando dados reais do Sistema Único de Saúde do Brasil.

1.1 Motivação

Identificar dados duplicados é um problema inerente a quase todos os bancos de dados relacionais e transacionais no mundo. No Brasil, agravado pela precariedade dos sistemas de informação, este problema é frequentemente encontrado nos grandes sistemas públicos como nas bases de dados do Sistema Único de Saúde (SUS,) no sistema de informação da Previdência Social, entre outros sistemas.

Diversas pesquisas tem sido realizadas com o intuito de integrar as bases de dados brasileiras. (Fonseca et al. [2010]), (Migowski et al. [2011]), (Queiroz et al. [2009] utilizaram do arcabouço de pareamento de registros na tentativa de solução para este problema, além disso, estes trabalhos avaliaram a qualidade dos dados integrados em pesquisas nas suas respectivas áreas de atuação.

Com o intuito de otimizar e qualificar o processo de pareamento de registros, através de métodos otimizadas de blocagem, este trabalho tem como motivação permitir estudos em outras áreas da ciência através da utilização de dados qualificados, consistentes e validados, bem como contribuir para a ciência da computação através de esforços da criação de algoritmos e modelos de análise de dados para prover qualidade no processo de pareamento de registros.

1.2 Objetivos

O objetivo deste trabalho consiste em pesquisar, implementar e avaliar um novo método de blocagem através da mineração de conjuntos fechados, o *CFI Blocking*, que é capaz de utilizar os valores das instâncias de atributos presentes em uma base de dados como blocos para a etapa de blocagem no pareamento de registros.

Assim, este trabalho tem como objetivos específicos:

- Gerar e avaliar conjuntos fechados de instâncias de atributos utilizando técnicas de mineração de dados, para permitir realizar uma blocagem mais eficiente no pareamento de registros;

- Transformar esses conjuntos fechados de instâncias de atributos gerados em blocos de registros para a etapa de blocagem no processo de pareamento de registros;

- Propor um algoritmo capaz de selecionar, dentre estes blocos gerados, os melhores candidatos, isto é: os blocos com maior probabilidade de possuírem pares verdadeiros, para serem enviados a etapa de comparação através de pareamento de registros;

- Avaliar os resultados deste algoritmo em comparação com os métodos atuais existentes em uma base de dados sintética, para analisar a viabilidade do algoritmo e em uma base de dados real do Sistema Único de Saúde do Brasil utilizando como parâmetros o custo computacional, a qualidade e métricas de avaliação que permitem mensurar a precisão e a revocação do algoritmo de blocagem proposto.

1.3 Justificativa e Relevância

O principal problema no pareamento de registros com várias bases de origem é que normalmente estas bases incluem identificadores que são diferentes entre os conjuntos

de dados ou simplesmente erradas devido a uma variedade de razões incluindo digitação ou transcrição com erros seja por descuido ou proposital de atividade fraudulenta (Hernández & Stolfo [1998]).

Desde o surgimento das grandes bases de dados, as pesquisas vem tentando reduzir a complexidade do problema particionando a base de dados em duas ou mais partições ou *clusters*, os quais possuem maior potencialidade para o casamento de registros associados ao mesmo cluster (Hernández & Stolfo [1998]).

A blocagem é parte do processo de pareamento de registros e tem como principal contribuição diminuir o custo computacional da comparação de registros. Apesar da importância da blocagem para a eficiência do processo de pareamento de registros, poucos foram os estudos que buscaram avaliar as vantagens da adoção de determinados esquemas de blocagem (Coeli & Camargo Jr. [2002]), ou diferentes algoritmos e nenhum destes estudos foram desenvolvidos tendo como objeto de estudo/trabalho bases de dados brasileiras.

Tipicamente os métodos tradicionais de blocagem no pareamento de registro são: *standard blocking* (Jaro [1989]) ou *sorted neighborhood* (Hernández & Stolfo [1998]) que são baseados na informação sintática de cada registro (Nin et al. [2007]). O problema é que a qualidade dos resultados providos por esses métodos é extremamente dependente dos registros classificados em cada bloco e da qualidade dos dados (Nin et al. [2007]). A solução para este problema, então, é reduzir a restrição na criação do blocos, construindo grandes blocos, ineficientes, que permitem comparar mais registros para encontrar todos, ou quase todos os registros duplicados.

Sabendo que a escolha inadequada de uma estratégia de blocagem para o pareamento de registros pode impactar de forma negativa o pareamento probabilístico de registros, gerando blocos de alto custo de avaliação, considerando a importância de se ter uma base íntegra, por exemplo, para o acompanhamento clínico de um paciente em uma bases de dados do SUS (Jaro [1989]), torna-se perceptível a necessidade de medição, compreensão, caracterização e análise de fenômenos racionados ao uso do *CFI Blocking* para enumeração dos blocos no processo de pareamento de registros com o intuito de diminuir o custo computacional e melhorar a qualidade deste processo.

Capítulo 2

Referencial Teórico

O presente trabalho tem como objetivo principal avaliar a viabilidade do uso de conjuntos fechados de instâncias de atributos para a enumeração de blocos lógicos, isto é, blocos formados por combinações de "e" ou "ou" para instâncias variadas dos atributos de uma base de dados com o uso do *CFI Blocking*. Para tornar possível a compreensão dos termos e algoritmos utilizados nesta dissertação, este capítulo apresenta os principais conceitos utilizados na avaliação e criação desses métodos: Pareamento de Registros e Mineração de Dados.

A seção 2.1 descreve o processo de pareamento de registros com explicações sobre o problema, o processo e suas etapas. Nesta seção, também está descrito em detalhes o que é blocagem, incluindo exemplos de execução dos métodos tradicionais.

A seção 2.2 descreve alguns conceitos e padrões de mineração de dados que permitem compreender como o uso desses conceitos e algoritmos podem possibilitar a otimização do processo de blocagem no contexto de pareamento probabilístico. A seção 2.3 apresenta trabalhos relacionados que apresentam como foco principal a pesquisa em pareamento de registros, estratégias de blocagem e mineração de dados como solução para a etapa de blocagem.

2.1 Pareamento de Registros

O pareamento de registros é um processo que envolve o cruzamento de dados com o intuito de gerar univocidade em uma base de dados. Esse cruzamento pode ser de duas bases de dados distintas (*linkage*) ou dentro de uma mesma base (deduplicação).

O *linkage* consiste na crença da existência de uma base unívoca e o objetivo é encontrar registros de uma outra base de dados na mesma. A deduplicação é a

busca por registros duplicados dentro de uma mesma base de dados ou em uma base "única"isto é uma base que é resultante da união de várias bases.

Além disso, o pareamento de registros pode ser de dois tipos: determinístico ou probabilístico, o determinístico visa encontrar registros exatamente iguais que por algum motivo tiveram sua chave primária diferente ou pertencem a bases de dados diferentes e é um problema de fácil solução, pois ele busca somente registros com todos os atributos exatamente iguais exceto a chave primária.

O pareamento probabilístico trata o pareamento de registros diferentes que possuem certo grau de semelhança. O resultado deste pareamento consiste na comparação dos registros, previamente considerados semelhantes no qual é dado uma probabilidade de um registro a ser igual ao outro registro b para cada par(a,b) de registros com probabilidade de serem os mesmos formado.

Assim, podemos formalizar os tipos de pareamento como: Dado um par(a,b), o pareamento determinístico é capaz de determinar se $a = b$, se, e somente se, todos os atributos de a $(a_1, a_2, ..., a_n)$ são respectivamente equivalentes aos atributos de b $(b_1, b_2, ..., b_n)$ excetuando-se a chave primária. O pareamento probabilístico é capaz de dar uma probabilidade de $a = b$ dado a proximidade ou equivalência dos atributos de a em comparação com b, ou seja, para cada atributo de a $(a_1, a_2, ..., a_n)$ comparado com o respectivo elemento em b $(b_1, b_2, ..., b_n)$ resulta em uma nota para cada atributo. A probabilidade final de $a = b$ segundo o pareamento probabilístico é dada por: $\sum_{i=1}^{n} P(A_i = B_i)$ onde n= número de atributos da base de dados e P = probabilidade dos elementos serem iguais. O cálculo detalhado foi definido no trabalho de Fellegi & Sunter [1969].

Esta dissertação trata do processo de pareamento probabilístico em bases de dados com foco na otimização do processo de blocagem, com o intuito de prover qualidade e eficiência na etapa de enumeração dos blocos e consequentemente no resultado do pareamento.

A subseção 2.1.1 descreve o problema a ser solucionado pelo pareamento probabilístico, enquanto a subseção 2.1.2 descreve de uma maneira geral o processo de pareamento de registros. A subseção 2.1.3 detalha o processo de blocagem, etapa alvo deste trabalho.

2.1.1 Problema

Com o surgimento de sistemas de informações e o início do processo de gravação digital de dados surgiram alguns problemas de inconsistência devido as más práticas de programação por parte dos programadores e também do mal uso dos sistemas por

parte dos usuários. Diversos dados são inseridos erroneamente nos bancos de dados, o que impossibilita, por exemplo, garantir a univocidade dos mesmos ou ainda efetuar análises reais desses dados.

O pareamento de registros tem o intuito de solucionar um problema conhecido na literatura como resolução de entidade. A resolução de entidade consiste em garantir que cada registro em uma base de dados corresponda a somente uma entidade do mundo real, por exemplo, João dos Santos Neto na base de dados corresponda a um, somente um, João dos Santos Neto no mundo real.

Existem diversos tipos de bases de dados: bases de dados de saúde, bases de dados de transações bancárias, bases de dados de programas governamentais, bases de dados empresariais, entre outros. Uma análise errada dos dados pode ser diretamente responsável por uma tomada de decisão errônea. Portanto, solucionar este problema é prover qualidade aos dados e consequentemente tornar possível uma melhor tomada de decisão.

2.1.2 Processo

O processo de pareamento probabilístico de registros, conforme a figura 2.1, inclui as seguintes etapas: Padronização e Limpeza, Análise dos Dados, Blocagem, Comparação e Classificação.

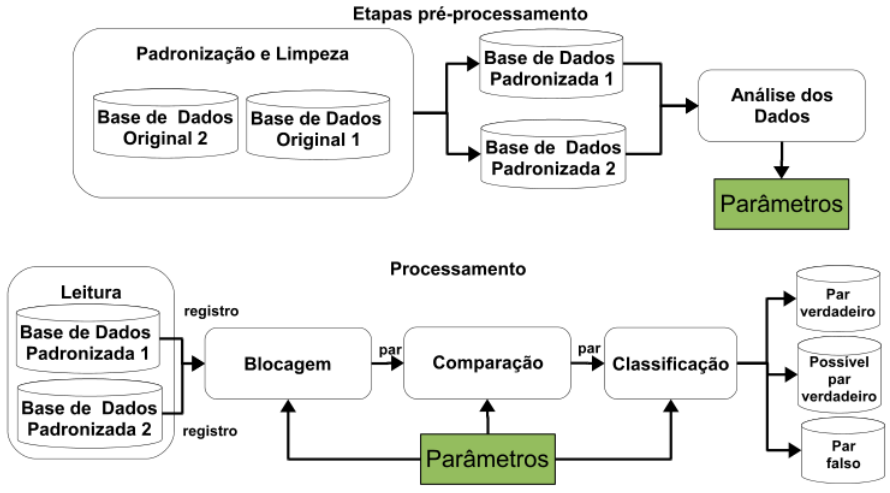


Figura 2.1. *Pipeline* de processos para o pareamento de dados (dos Santos [2008])

A primeira etapa consiste na limpeza e padronização dos dados, que inclui: classificação de nomes, padronização de campos, remoção de dados inconsistentes entre

outros processos. Quanto mais limpo e padronizado é um banco de dados, melhor qualidade será encontrada no resultado do pareamento.

A segunda etapa consiste na análise dos dados e criação de parâmetros, que são os parâmetros indicados para comparação dos registros e futura classificação como duplicado ou não. Esta etapa é de suma importância pois uma simples variação nos parâmetros pode transformar todo o resultado.

A terceira etapa consiste no pareamento probabilístico. O pareamento é dividido em três principais etapas: Blocagem, Comparação, Classificação. A etapa de blocagem é responsável por separar grupos de registros que tem maior probabilidade de serem a mesma pessoa e desonerar o custo da etapa de comparação. Esta etapa é o alvo de melhoria desta dissertação e está melhor detalhada na seção 2.1.3.

A etapa de comparação consiste em usar os parâmetros definidos na segunda etapa do processamento, análise dos dados, para comparar os registros de cada grupo, atribuindo os valores definidos nos parâmetros para cada atributo e calculando uma nota para cada par baseada na concordância e discordância de cada atributo entre eles. Após a comparação, os valores calculados são passados como parâmetros para a etapa de classificação.

A classificação determina sobre os resultados obtidos da comparação quais pares serão considerados verdadeiros, falsos ou duvidosos. Esta decisão é baseada em um processo de conferência, que define uma nota minima para o par ser considerado verdadeiro e uma nota máxima para o par ser considerado falso.

2.1.3 Blocagem

Um dos aspectos mais significantes do pareamento de registros está relacionado a blocagem. A blocagem tem como foco principal diminuir o número de comparações entre pares de registros e aumentar a velocidade de execução do processo de pareamento probabilístico. Percorrer toda uma base de dados procurando registros duplicados é uma tarefa custosa de solução, $O(n^2)$, e muitas vezes não faz sentido, por exemplo, comparar um registro de nome Lucas, nascido na cidade de Belo Horizonte do sexo masculino com um registro de nome Sâmara, nascida na cidade de Belém e do sexo feminino. Existe um desperdício computacional ao se realizar o método força bruta de comparação, gerando comparações desnecessárias.

O processo de blocagem pode ser dividido em duas principais etapas: enumeração dos blocos e geração dos pares. A primeira etapa, no método de blocagem padrão (Jaro [1989]), possui como parâmetro a indicação de condições lógicas de restrição por parte do pesquisador na seleção dos atributos, por exemplo, nome e sexo. Isto significa que

os registros da base de dados serão reagrupados e separados em blocos de registros que contenham em comum os atributos nome e sexo. Esta abordagem se mostra eficiente, pois visa com um julgamento prévio comparar registros que são mais parecidos isto é, possuem atributos em comum e fazem maior sentido para a etapa de comparação.

A segunda etapa é a enumeração dos pares para cada bloco. Dado que um bloco é formado por uma condição lógica que restringe os elementos deste bloco, ou seja, todos os elementos pertencentes a um bloco possuem no mínimo os atributos da condição de blocagem em comum, todos os registros desse bloco devem ser comparados entre si. Esta comparação deve ser realizada pois esses registros possuem uma probabilidade dada pela condição de blocagem de representarem a mesma entidade no mundo real. O custo da criação dos pares é de aproximadamente $O(n^2)$, entretanto na comparação de entidades, existe a referência direta, ou seja, formar o par (1,2) é o mesmo que formar o par (2,1). A ordem dos registros não importa para o par e sua respectiva comparação.

Sendo assim, essa enumeração é dada pela seguinte fórmula: $C_2^n = \binom{n}{2} = \frac{n!}{2!\cdot(n-2)!}$, que é a combinação simples de dois elementos para todos os elementos de um mesmo bloco. A saída da segunda etapa são pares a serem comparados por cada bloco, por exemplo, para um bloco x que contém os registros (1,2,3) os pares são: registro 1 com registro 2, registro 1 com registro 3, registro 2 com registro 3.

A definição da estratégia de blocagem deve considerar tanto os erros em atributos quanto a frequência dos valores dos atributos, pois são aspectos que influenciam diretamente o número de pares gerados e a qualidade do resultado final. Erros nos atributos podem prejudicar a qualidade dos pares gerados, ao levar a exclusão de pares que seriam, de fato, verdadeiros (Goncalves et al. [2008]). Assim, a escolha dos atributos deve levar em conta a qualidade da informação ali contida. Idades, por exemplo, podem ser declaradas com erro ou arredondadas para zero ou cinco (último dígito), nomes podem ser escritos de diferentes maneiras, o que pode dificultar a determinação de pares verdadeiros.

A tabela 2.1 mostra um exemplo de uma base de dados em seu estado original, com erros de digitação, abreviações, dados ausentes, duplicações determinísticas, entre outros. É uma tabela de exemplo com 4 atributos diferentes (Nome, Sobrenome, CPF e Cidade) na qual existem ainda erros de semântica, localização e padrão e é usada. nesta dissertação, como exemplo para a criação de blocos através do método padrão de blocagem.

A figura 2.2 exibe algumas possíveis estratégias de blocagem geradas para a tabela 2.1 e seus respectivos blocos. Foram enumeradas 3 estratégias de blocagem: A estratégia um agrupando registros que possuem os atributos nome e cidade equivalentes, a estratégia 2 considerando apenas nomes iguais e a estratégia 3 que leva em consi-

Tabela 2.1. Erros Comuns nos Bancos de Dados

ID	NOME	SNOME	CPF	CIDADE
1	Lucas	Sousa	145396	BH
2	Lucas		145396	Belo Horizonte
3	Lucas	Souza	325	

deração apenas o atributo CPF para igualdade. É possível perceber que existe uma perda de possíveis pares verdadeiros no processo de formação dos pares pelos blocos, por exemplo, a estratégia 1 não forma nenhum par. Estes tipos de erro podem aparecer com maior ou menor frequência de acordo com a estratégia de blocagem escolhida.

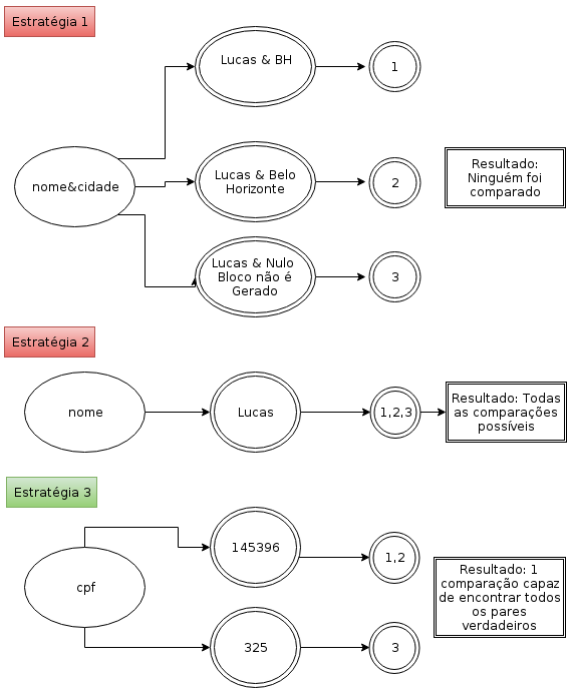


Figura 2.2. Estratégias de Blocagem para a Tabela 2.1.

Enquanto a estratégia 1 não gera nenhum par, a estratégia 2 gera apenas um bloco grande que compreende todos os registros da base. A formação deste bloco não é interessante pois o custo de criação/comparação dos pares dentro de um bloco é de aproximadamente $O(n^2)$, ou seja cada registro pertencente ao bloco deve ser comparado com todos os outros do mesmo bloco. Gerar blocos grandes não é o objetivo, uma vez que isto pode onerar computacionalmente o processo de comparação com a realização de comparações desnecessárias.

A estratégia 3 gera dois blocos distintos e faz com que as comparações verdadeiras estejam separadas em apenas um bloco, o que atinge o objetivo de reduzir o número de comparações e maximizar a possibilidade de encontrar os pares verdadeiros. Essa

estratégia mostra que a escolha de uma estratégia de blocagem é fortemente dependente do conhecimento prévio da base por parte do pesquisador e da qualidade dos dados nesta base.

Este trabalho apresenta o *CFI Blocking*, um algoritmo para a enumeração de blocos baseado em conjuntos fechados frequentes de instâncias dos atributos da base de dados. A figura 2.3 mostra a diferença entre o método de blocagem tradicional e o *CFI Blocking*, ou seja, na estratégia para enumerar blocos, vistas na etapa 2 da figura.

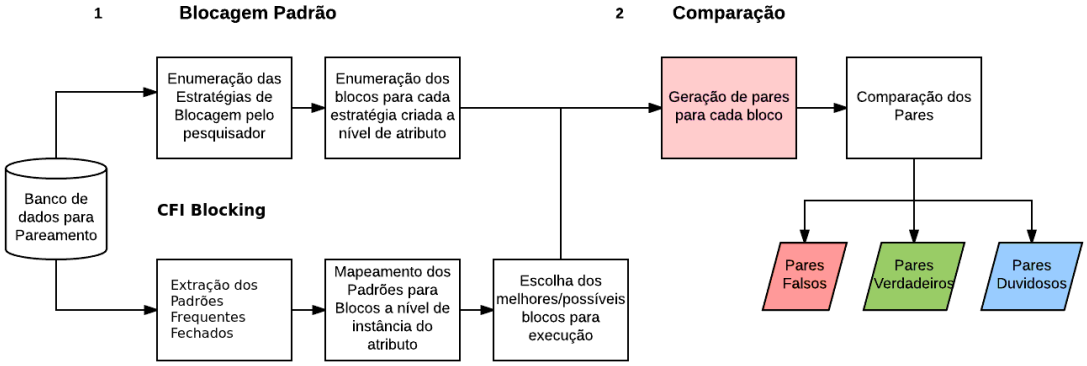


Figura 2.3. Blocagem Tradicional x CFI Blocking

2.2 Mineração de Dados

A mineração de dados é um processo de descoberta de conhecimento através dos dados. Minerar é a capacidade de procurar, descobrir informações presentes nos dados não descobertas anteriormente ou não pesquisadas diretamente por um pesquisador ou interessado nos dados.

Alguns autores citam mineração de dados como descoberta de conhecimento em banco de dados ou aplicação de algoritmos específicos para extração de padrões dos dados (Fayyad et al. [1996]), análise exploratória dos dados (Kantardzic [2002]), descoberta de correlações utilizando técnicas estatísticas e matemáticas (Larose [2004]), descoberta de *insights* e modelos preditivos em grandes bases de dados (Zaki & Meira Jr [2014]).

2.2.1 Conjuntos de Itens Frequentes

A etapa de blocagem no pareamento de registros, bem como muitas outras aplicações, está interessada em saber quantas vezes dois ou mais objetos de interesse coocorrem buscando tentar identificar se esta co-ocorrência pode significar um dado duplicado

ou não. Assim, recorrer a busca por conjuntos de itens frequentes pode melhorar a qualidade do processo de enumeração dos blocos na etapa de blocagem e consequente melhora no resultado do pareamento.

Seja I = $\{x_1, x_2, ..., x_m\}$ um conjunto de elementos chamado itens. Um conjunto $X \sqsubseteq I$ é chamado de *Itemset* ou conjunto de itens. Um conjunto de itens de tamanho k e chamado de *k-itemset* ou k-conjuntos-de-itens. $I(k)$ é um conjunto de todos os k-conjuntos-de-itens que são sub conjuntos de I com tamanho k (Zaki & Meira Jr [2014]).

Seja T = $\{t_1, t_2, ..., t_n\}$ um outro conjunto de elementos chamado transações. Um conjunto $L \sqsubseteq T$ é chamado *tidset* ou conjunto de transações. Os conjuntos de itens e conjuntos de transações devem ser mantidos ordenados em ordem lexicográfica.

Uma transação é uma tupla, ou linha na forma (L, X) onde $L \in T$ é um identificador único da transação e X é um conjunto de itens. O conjunto de transações T pode ser um conjunto de pessoas, um conjunto de compras ou outros para efeitos de melhor clareza do problema, uma transação (L, X) é referida nesta dissertação apenas como (L) pois uma transação sempre está ligada a um conjunto de itens X. (Zaki & Meira Jr [2014])

O suporte de um conjunto de itens X em um banco de dados D, chamado de $sup(X, D)$ é o número de transações em D que contem X. Um conjunto de itens X é dito frequente em D se $sup(X, D) >= minsup$ onde $minsup$ é um suporte mínimo definido pelo usuário. Assim, encontrar padrões frequentes é encontrar conjuntos de itens, que sozinhos, ou combinados, tenham um suporte maior ou igual ao suporte mínimo estabelecido pelo usuário.

O espaço de busca de conjuntos de itens frequentes é usualmente muito grande e cresce exponencialmente com o número de itens. Em particular, um baixo valor de suporte pode resultar em incontáveis conjuntos de itens frequentes. Uma alternativa é usar representações que resumam as características essenciais desses dados. Duas representações foram utilizadas neste trabalho: os conjuntos de itens frequentes maximais e os conjuntos de itens frequentes fechados.

2.2.2 Conjuntos Maximais de Itens Frequentes

Dado um banco de dados binário $D \subseteq TxI$ com um conjunto de transações T e itens I, seja F um conjunto de todos os conjuntos de itens frequentes dado por:

$$F = \{X | X \subseteq L \text{ and } sup(x) >= minsup\} \text{ (Zaki \& Meira Jr [2014])}$$

Um conjunto de itens $X \in F$ é um conjunto de itens frequentes maximal em um conjunto F se X é frequente, e não existe nenhum super-conjunto de itens tal que $X \subseteq Y$ e Y é também frequente em F.

$$M = \{X | X \in F \text{ e } \nexists Y \supset X, \text{ no qual } Y \in F\} \text{ (Zaki \& Meira Jr [2014])}$$

O conjunto M de maximais é uma representação condensada do conjunto de todos os conjuntos de itens frequentes F. Os conjuntos de itens frequentes maximais permitem determinar se qualquer conjunto de itens X é frequente ou não. Se existe um conjunto maximal Z e $X \subseteq Z$, X tem que ser frequente.

2.2.3 Conjuntos Fechados de Itens Frequentes

Um conjunto de itens $X \in F$ é fechado em um conjunto de dados F se não existe nenhum super-conjunto de itens tal que $X \subseteq F$ e Y tenha o mesmo suporte que X em F.

$$C = \{X | X \in F \text{ e } \nexists Y \supset X \text{ no qual } sup(X) = sup(Y)\} \text{ (Zaki \& Meira Jr [2014])}$$

$X \in F$ é um conjunto de itens frequente fechado se todos os super conjuntos de X tem suporte estritamente menor que X. $sup(X) > sup(Y)$ para todos $Y \supset X$.

As relações entre os tipos de conjuntos de itens frequentes respeitam a seguinte ordem:

$$M \subseteq C \subseteq F \text{ (Zaki \& Meira Jr [2014])}$$

Conhecer os conjuntos maximais e sua frequência permite reconstruir todo o conjunto de itens frequentes. Conhecer todos os conjuntos fechados e sua frequência, permite reconstruir o conjunto de todos os conjuntos de itens frequentes e suas frequências.

2.3 Trabalhos Relacionados

Nesta seção está descrito brevemente estratégias de blocagem populares, metodologias de avaliação dessas estratégias e estratégias de blocagem baseadas em aprendizado de máquina e mineração de dados.

O método mais popular de blocagem é a blocagem padrão, do inglês *standard blocking* (SB), que é uma técnica tradicional que agrupa registros em blocos que são idênticos a uma dada chave de blocagem separadas a nível de atributo (Jaro [1989]).

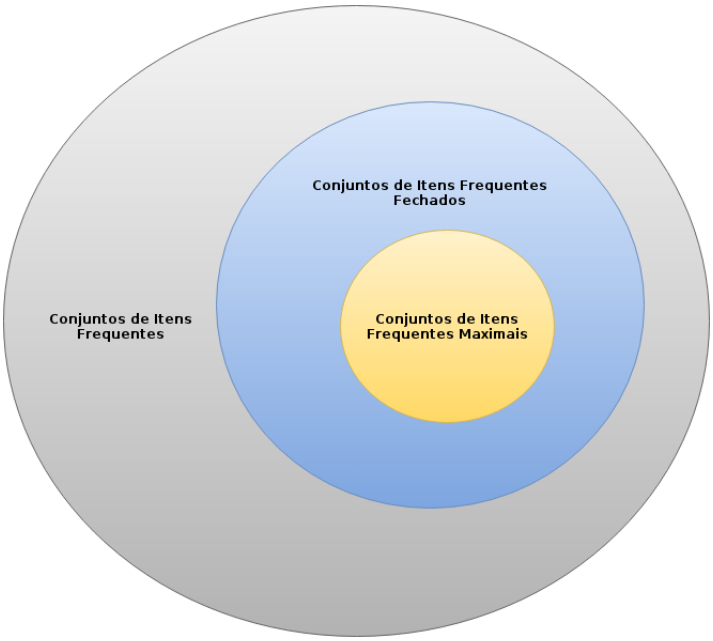


Figura 2.4. Definição dos tipos de conjuntos de itens frequentes.

Outro método é o *Bigram Indexing* (BI) (Baxter et al. [2003]), uma abordagem difusa da blocagem (*fuzzy blocking*), possibilitando que alguns erros tipográficos possam ser captados. Outros exemplos são *sorted neighborhood* (Hernández & Stolfo [1998]) e *canopy clustering* (McCallum et al. [2000].

Em termos de metodologias de avaliação, o trabalho de O trabalho de (Coeli & Camargo Jr. [2002]) teve como objetivo comparar a eficiência de diferentes esquemas de blocagem e estudar a eficiência da utilização de uma rotina de padronização desenvolvida pelos autores, que aplica a mesma grafia para as primeiras sílabas de nomes com o mesmo som. Foi realizado o procedimento de relacionamento de uma base de dados de mortalidade com 59.065 óbitos com uma base de óbitos hospitalares com 531 registros, que apresentavam um registro correspondente na base de mortalidade. Foram apresentadas diversas estratégias de blocagem, a estratégia de blocagem em múltiplos passos foi mais eficiente, permitindo a identificação de todos os pares verdadeiros com a formação de um número total de pares que foi inferior ao obtido em duas rotinas diferentes de passo único. Já entre as estratégias de passo único avaliadas, a que se baseou no emprego da chave formada pela combinação do código *soundex* do primeiro nome e sexo apresentou o melhor resultado. (Nin et al. [2007]) trabalhou com o intuito de mostrar que a exploração das relações (por exemplo, chave estrangeira) entre uma ou mais fontes de dados, permite explorar um novo tipo de método de blocagem semântica que melhora o número de acessos e reduz o esforço.

No contexto da aplicação de técnicas de aprendizado de máquina e mineração de dados ao problema de blocagem, podemos destacar os seguintes trabalhos. (de Carvalho et al. [2012]) apresenta um algoritmo de programação genética para deduplicação de registros. O algoritmo produz uma função de deduplicação de redundância que é capaz de identificar se duas ou mais entradas em um repositório são réplicas ou não. O trabalho de (McNeill et al. [2012]) utiliza uma técnica de classificação para tentar prever os melhores blocos a serem avaliados, o que é feito de forma incremental, a cada novo atributo considerado. (Kenig & Gal [2013]) apresenta o primeiro algoritmo para blocagem com o uso de mineração de padrões frequentes, o MFI Blocking. Este algoritmo determina blocos baseado em conjunto de itens frequentes maximais, que são os maiores conjuntos que satisfazem um limiar de frequência pré-definido chamado suporte e cujos sub-conjuntos também satisfazem o mesmo limiar. A estratégia também realiza uma análise de qualidade dos blocos para estimar a possibilidade de existência de pares duplicados dentro do bloco. CFI Blocking é semelhante à esta última, mas o uso de conjuntos fechados permite obter blocagens mais eficientes.

Dentre os outros algoritmos exemplos dos algoritmos para enumeração de blocos O método *Bigram Indexing* (BI) (Baxter et al. [2003]) permite uma abordagem difusa da blocagem (fuzzy blocking), possibilitando que alguns erros tipográficos possam ser captados. A ideia básica é converter a chave de blocagem em uma lista de bigramas (sub-strings de dois caracteres), por exemplo, uma chave de blocagem com o valor "priscila"gera os bigramas ("pr","ri","is","sc","ci","il","la"). A partir desta lista de bigramas, sub-listas serão geradas usando todas as permutações possíveis sob um limite passado como parâmetro t com valor entre 0 e 1. As listas resultantes são convertidas em chaves de blocagem e o processo de blocagem continua da mesma forma do que a blocagem padrão (Goncalves et al. [2008]). Este método foi implementado no sistema Fril de *record linkage*.

O *Sorted Neighbourhood* (SN) (Hernández & Stolfo [1998]) é uma técnica que se baseia na ordenação dos registros pela sua chave de blocagem e na utilização de uma janela deslizante de tamanho fixo w (parâmetro). Somente registros dentro da mesma janela formarão pares candidatos. Desta forma, o número de comparações realizadas tem sua complexidade reduzida de $O(n^2)$ para $O(w * n)$ (Baxter et al. [2003]), com w sendo o tamanho da janela. Nessa técnica, o conceito de blocos é difuso, visto que um bloco é formado por todos os registros contidos em uma janela deslizante. Assim, são $(n * w) + 1$ blocos, onde n e número total de registros a serem analisados (Goncalves et al. [2008]).

(Baxter et al. [2003]) compara dois novos algoritmos de blocagem, *Bigram indexing* e *Canopy Clustering*, com os algoritmos popularmente conhecidos, *standard*

blocking e *sorted Neighbourhood*. (Evangelista et al. [2009]) propõe uma blocagem adaptativa e flexível usando algoritmos genéticos. Os resultados mostram que as novas estratégias de blocagem possuem maior escalabilidade melhorando ou mantendo o padrão de qualidade do *record linkage*. Estes novos métodos são potencialmentes mais rápidos e mais precisos.

É importante ressaltar que a maioria desses trabalhos enumeram blocos a partir de conjuntos de atributos e não instâncias isoladas, como CFI Blocking, o que pode resultar em blocos maiores de registros e consequentemente com maior custo computacional.

Capítulo 3

Metodologia e Modelagem

Esta capítulo descreve as origens dos dados utilizados nesta pesquisa, bem como a metodologia utilizada no processamento, análise e avaliação das soluções propostas nesta dissertação. Na seção 3.1 está definido o escopo das bases de dados utilizadas para o pareamento. A seção 3.2 descreve o modelo utilizado para criação do algoritmo proposto. A seção 3.3 define as métricas utilizadas para análise e avaliação dos resultados obtidos.

3.1 Bases de Dados

3.1.1 Base de Dados Sintética

Para avaliar a confiabilidade do CFI Blocking foram executados testes utilizando dados sintéticos. A base de dados sintética foi gerada através do software FEBRL (Christen [2008]) contendo 5000 registros em distribuição Poisson, com 1000 registros duplicados e com alteração randômica dos valores das instâncias dos seus atributos.

3.1.2 Base de Dados Real

No Brasil, o Sistema Único de Saúde (SUS) possui sistemas de informação com grande potencial como fonte de informações que auxiliam na definição de prioridades e diretrizes para a gestão do sistema de saúde. No entanto, esses sistemas são fragmentados e desarticulados, tendo uma visão centrada nos procedimentos e não no indivíduo, o que dificulta o acompanhamento longitudinal dos pacientes.

Os grandes sistemas de informação em saúde nacionais são apresentados por (da Cruz Gouveia Mendes et al. [2000]) que classifica-os como sistemas de informações

assistenciais ou sistemas de informações epidemiológicas. Entre os primeiros, podemos enumerar o Sistema de Informações Hospitalares (SIH) e o Sistema de Informações Ambulatoriais (SIA). Entre os segundos, o Sistema de Informação sobre Mortalidade (SIM), o Sistema de Informações sobre Nascidos Vivos (SINASC) e Sistema de Informações de Agravos de Notificação (SINAN). O mesmo autor aponta a desagregação e a não padronização das informações como um problema dos sistemas de informação em saúde no Brasil.

Para realização deste trabalho e aplicação do processo de pareamento de registro ilustrado na figura 2.1 foram utilizados os dados do Sistema Único de Saúde (SUS) do Brasil que se encontra disponível no departamento de Farmácia Social (FAS) na faculdade de Farmácia da Universidade Federal de Minas Gerais (UFMG), mediante autorização e acordo com o Ministério da Saúde do governo federal. Os dados contemplam o período de 2000-2010 contendo os bases de dados de registro de mortalidade(SIM), internação hospitalar (AIH), e de farmácia ambulatorial (SIA-APAC).

A base de dados disponibilizada, inicialmente, era composta por aproximadamente 500 milhões de registros possivelmente ser únicos. Após a execução das etapas de padronização e limpeza, análise dos dados (Pereira et al. [2015]), constatou-se nesses dados, diversas duplicações de registros. Assim, tornou-se necessário a submissão dos mesmos ao processo de pareamento de registros.

A primeira execução do pareamento foi realizada de forma determinística e teve como resultados uma base de dados condensada em aproximadamente 400 milhões de registros, totalizando 106 milhões de registros únicos. Esta diminuição pode ser explicada pelo modelo original da base de dados, que em alguns anos, utiliza cada linha como uma operação do negócio e não como um paciente único. Com isto, o paciente perde a característica de univocidade pois para um mesmo paciente existem diversas operações.

Como o objetivo do pareamento é encontrar identificações unívocas para as uma mesma entidade do mundo real, os dados estavam duplicados. O problema foi apenas parcialmente solucionado. Isto ocorreu pois o pareamento determinístico trabalha apenas com registros exatamente iguais não sendo levado em conta probabilidades de proximidade, erros de digitação ou inconsistência nos dados, havendo uma demanda pela aplicação do método de pareamento probabilístico.

Realizar um pareamento probabilístico em grandes bases de dados demanda tempo e poder de processamento. Testes preliminares foram realizados para avaliar o custo de usar à base de dados do estado de Minas Gerais para os experimentos e foram encontrados aproximadamente 13.000.000 de registros com o número aproximado de 250.000 itens. Este número foi considerado custoso computacionalmente para expe-

rimentos, assim, foi extraída uma amostra da região metropolitana de Belo Horizonte contendo aproximadamente 1.700.000 de registros com 150.000 itens distintos.

Nesta dissertação, optou-se por utilizar, selecionados por referência de localidade, os dados da região metropolitana de Belo Horizonte - Minas Gerais (MG). Esta seleção possui o intuito de maximizar a possibilidade de se encontrar registros duplicados bem como diminuir a quantidade de registros totais, tornando a avaliação factível no âmbito desta dissertação. Esta amostra foi extraída com a utilização de filtro nos dados por código de município do Brasil.

A região metropolitana de Belo Horizonte, de acordo com o Instituto Brasileiro de Geografia e Estatística (IBGE), é formada por 34 municípios, com um total aproximado de 5,8 milhões de habitantes - População das Regiões Metropolitanas". Instituto Brasileiro de Geografia e Estatística (IBGE). 28 de agosto de 2014. Consultado em 24 de fevereiro de 2016. A amostra extraída conta com 1.666.921 registros encontrados nas bases de dados de internação hospitalar, serviços ambulatoriais e mortalidade do SUS.

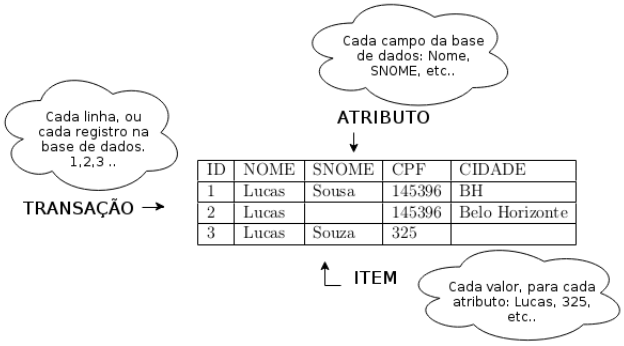
Esta amostra não tende a afetar a qualidade e a veracidade dos resultados gerados na avaliação, uma vez que o maior custo na etapa de mineração dos dados é baseado no tamanho do domínio de itens e este tamanho não cresce proporcionalmente ao número de transações (registros). Isto acontece porque o universo dos dados está limitado, por exemplo, para sexo só existem duas opções: masculino e feminino, para estados brasileiros, são apenas 27 estados, entre outros.

3.2 Modelo

Esta seção tem como objetivo conceituar os padrões e termos adotados no processo de desenvolvimento do algoritmo *CFI Blocking*. Este algoritmo tem como objetivo utilizar os conjuntos fechados de itens frequentes como blocos no processo do pareamento de registros.

Conforme explicado no capítulo 2, uma transação é um registro ou uma "linha"presente na base de dados. Os conceitos de itens e transações foram criados nos primórdios da mineração de dados que possuía o intuito de encontrar padrões em compras (transações) de produtos (itens) mas pode ser adaptado para o presente trabalho onde um atributo é um campo ou uma coluna de uma base de dados. Uma transação é uma tupla, ou uma linha de uma base de dados que contem um identificador único e vários itens. Um item é a representação de cada valor distinto de uma instância de um atributo. Um conjunto de itens, é formado por transações que contém os itens que

pertencem a a este conjunto de itens com um suporte mínimo S. Estes termos são representados na figura 3.1 que exibe os termos adotados para cada parte da tabela 2.1 onde podemos ter, por exemplo, o conjunto de itens {Lucas} que é encontrado com suporte 3, isto é, o item Lucas está presente nas transações 1, 2 e 3.



ID	NOME	SNOME	CPF	CIDADE
1	Lucas	Sousa	145396	BH
2	Lucas		145396	Belo Horizonte
3	Lucas	Souza	325	

Figura 3.1. Termos do Modelo.

Nesta dissertação os blocos, são formados a partir das instâncias dos atributos de cada transação, ou seja, por conjunto fechados de itens encontrados. Esses predicados são gerados através da mineração de dados aplicada nas bases de dados de origem. Os conjuntos de itens encontrados determinam quais atributos em comum existem entre as transações. Um conjunto de itens representado por {cpf=145396 & nome=Lucas}, é composto pelas transações 1 e 2 nas quais esses itens ocorrem conforme a figura 3.1. Para ara o pareamento de registros, isto quer dizer que os registros 1 e 2 possuem pelo menos o valor dos atributos cpf e nome da mãe em comum.

Como o pareamento probabilístico visa encontrar registros que referem a uma mesma entidade, é intuitivo que quanto maior a cardinalidade do conjunto fechado de instâncias de atributos frequentes, mais atributos em comum existem entre os registros que contém estas instâncias e, consequentemente, maior é a probabilidade desses registros se referirem à mesma entidade. Por exemplo, um conjunto de itens composto pelo item {nome=Lucas}, quer dizer que todas as transações com o nome Lucas pertencem a este bloco.

Na prática, para o exemplo da figura 3.1, isto tem pouca significância no mundo real pois é fato que que existem várias pessoas com o nome Lucas e que não são a mesma pessoa. Entretanto um conjunto de itens composto pelos itens {nome=Lucas & cidade=Belo Horizonte & cpf=145396}, indica que os registros associados possuem pelo menos três itens em comum. Sendo assim, podemos afirmar que um bloco oriundo de um conjunto de instâncias de atributos I_n com n valores de instâncias tem maior

probabilidade dos registros pertencerem a mesma entidade do que um bloco formado por conjunto de tamanho I_{n-1} tal que I_{n-1} é um subconjunto de I_n.

3.3 Métricas de Avaliação

Esta seção descreve as métricas utilizadas na literatura para análise e validação das estratégias de blocagem e do pareamento de registros que foram utilizadas para avaliação no capitulo 5.

O trabalho realizado por (Gu et al. [2003]) sobre os estágios e trabalhos futuros de pareamento de registros contém uma lista de métricas utilizadas na análise da qualidade de um pareamento probabilístico, algumas dessas métricas são utilizadas nas avaliações dos resultados desta dissertação e estão listadas nesta seção.

Com o intuito de facilitar a formalização das métricas listadas abaixo, as seguintes siglas devem ser consideradas.

- p_v - número de pares verdadeiros
- p_t - número de pares comparados totais
- p_f - número de pares falsos
- p_{v_t} - número de pares verdadeiros totais
- p_{f_t} - número de pares falsos totais

3.3.1 Sensitividade - Precisão

A sensitividade de um pareamento/bloco corresponde ao número de pares verdadeiros encontrados dividido pelo total de pares verdadeiros existentes. A sensitividade mede a porcentagem de acerto nos pares verdadeiros encontrados e também é chamada na literatura como métrica de revocação.

$$Sensitividade = \frac{p_v}{p_{v_t}}$$

3.3.2 Especificidade

A especificidade de um pareamento/bloco corresponde ao número de pares identificados como falsos dividido pelo número total de pares falsos. A especificidade mede a porcentagem de acerto nos pares falsos encontrados.

$$Especificidade = \frac{p_f}{p_{f_t}}$$

3.3.3 Valor Preditivo Positivo - Revocação

O valor preditivo positivo mede a porcentagem de acertos de um pareamento/bloco em relação ao tamanho total do bloco e corresponde ao número de pares verdadeiros encontrados dividido pelo número de pares comparados totais. O valor preditivo positivo (ppv) também é conhecido na literatura como uma métrica de precisão.

$$ppv = \frac{p_v}{p_t}$$

3.3.4 Taxa de Registros Pareados

A taxa de registros pareados corresponde ao número total de pares comparados dividido pelo número total de pares verdadeiros. Essa métrica é utilizada para mostrar o tamanho do espaço de busca em relação a quantidade de pares verdadeiros que devem ser encontrados.

$$\text{trl} = \frac{p_t}{p_v}$$

3.3.5 Média Harmônica entre Precisão e Revocação

A média harmônica corresponde a relação de equilíbrio entre duas métricas. Esta métrica foi utilizada para uma avaliação mais just entre as métricas de precisão e revocação em relação ao tempo de execução do algoritmo.

$$\text{F1} = \frac{2 * sensitividade * ppv}{sensitividade + ppv}.$$

3.3.6 Trajetórias

A fisibilidade das trajetórias foram avaliadas utilizando-se a media, a mediana da similaridade dos elementos de um mesmo cluster e pela métrica chamada aqui de trajetória, que é árvore geradora mínima de um grafo dada por:

$$\text{T} = \frac{n^o dearestas}{n^o devertices - 1}$$

Capítulo 4

Algoritmos

Este capítulo apresenta uma descrição detalhada do *CFI Blocking* segundo os termos e modelos apresentado no capítulo 3. Na seção 4.1 está descrita a primeira etapa do algoritmo: mapeamento do banco de dados, na seção 4.2 esta descrito o processo de extração dos conjuntos de itens frequentes pelo algoritmo, enquanto na seção 4.3 está descrito o processo de criação e seleção dos blocos. Para finalizar o capítulo, a seção 4.4 apresenta uma análise de complexidade deste algoritmo.

O *CFI Blocking* visa criar blocos de forma mais eficaz através das propriedades de fechamento. Para isto seu algoritmo foi dividido em 3 partes principais: mapeamento do banco de dados, extração dos conjuntos fechados de itens frequente e criação dos blocos de comparação. Duas outras partes secundárias devem ser consideradas: a etapa de comparação e a etapa de geração de identificadores únicos, essas etapas fazem parte do processo final do pareamento probabilístico de registros sendo utilizada neste artigo para avaliação do método proposto. A tabela 4.1 é um exemplo de entrada para o algoritmo e se propõe a auxiliar no entendimento dos passos a seguir.

Tabela 4.1. Tabela de Exemplo: Estado Inicial.

ID	NOME	CIDADE	ESTADO	CEP	NOME DA MÃE	DT NASC	CPF
1	Mary Jane Berg		MG	12180	Sarah J Santos	18-11-1950	157890
2	Mary J Berg	BH	MG		Sarah J Santos	18-11-1950	157890
3	Maryah Jani Berg	BH	MG	12180	Sarah J Santos		

4.1 Mapeamento do Banco de Dados

Na mineração de dados, com o intuito de aumentar a velocidade e o poder computacional dos algoritmos de determinação de conjunto de itens frequentes, é comum utilizar

números inteiros para representar elementos dentro de uma transação e/ou até mesmo bases de dados binárias. Nos bancos de dados relacionais e/ou tradicionais, os dados, na maioria das vezes, estão armazenados como texto. Sendo assim, se faz necessário uma adaptação dos dados transacionais para os dados a serem utilizados no modelo proposto.

O algoritmo 1 representa o algoritmo utilizado para o mapeamento da base de dados. Este algoritmo tem como objetivo assinalar um número inteiro que corresponde a cada item distinto na base de dados original. Para a tabela 4.1 o conjunto de atributos S_f é dado por $\{nome, cidade, estado, cep, nome\ da\ me, dtnasc, cpf\}$. A linha 9 do algoritmo 1 exibe um comentário sobre a repetição dos itens em campos distintos, isto é: para o atributo nome é possível existir o item SILVA, enquanto para o atributo sobrenome também é possível existir o item SILVA. Apesar de possuírem o mesmo valor sintático, eles possuem valores semânticos diferentes. Sendo assim, são itens distintos pois correspondem a diferentes entidades do mundo real, neste caso, nome e sobrenome).

Algorithm 1 Mapeamento do Banco de Dados.
Precondition: *bancodedados* original separado por espaço

```
1:  function MapearDB(bancodedados)
2:      count ← Inteiro para Cada Atributo
3:      S_f ← Conjunto de Atributos
4:      S_e ← Conjunto de Itens
5:      I_u ← Conjunto de Itens Únicos com número, item, frequência
6:      for i ← 1 to S_f.Lenght do
7:          S_e ← S_{f_i.itens}
8:          for j ← 1 to Se.Lenght do
9:              item ← S_{e_i} + '_' + i ▷ Diferenciar valores iguais em atributos distintos.
10:             if !I_u.contains(item) then
11:                 I_u.update(I_u.freq + 1)
12:             else
13:                 I_u.add(count, item, 1)
14:                 count++
15:             end if
16:         end for
17:     end for
18:     return corte(I_u)
19: end function
```

Além disso, estes dois itens possuem um valor específico de frequência para cada um e esses valores devem ser tratados individualmente na etapa de determinação de conjunto de itens frequentes. O artifício utilizado neste pseudocódigo consiste em

concatenar a posição do atributo (coluna) a cada item, tornando assim o item sempre distinto. Assim, para cada item, caso ele não tenha sido descoberto antes, é assinalado um número inteiro distinto. Caso o item se repita dentro de um mesmo atributo linha 10, a frequência do item é incrementada em 1. Essa frequência pode ser utilizada como uma estratégia de corte para a utilização dos dados na etapa de determinação dos blocos.

Este corte foi utilizado pois dado que um item I tenha frequência 1 para uma determinada base de dados, isto significa que este item possui suporte = 1. Como o pareamento probabilístico visa encontrar pares de registros com probabilidade de representarem a mesma entidade no mundo real, um conjunto de itens com um item de suporte 1 não é capaz de gerar um par, pois só existe uma transação com este item. Assim, consequentemente todos os subconjuntos e combinações de conjuntos de itens frequentes com o item I, não possuem capacidade de formar um par. Para se evitar o custo de combinações entre os itens com suporte 1, os mesmos foram eliminados da base de dados e assinalados como nulos.

A figura 4.1 mostra o resultado do mapeamento aplicado na tabela 4.1. É possível visualizar na figura 4.1(a) o conjunto de itens únicos, que é formado pelos itens $\{A, B, C, D, E, F, G, H, I\}$. Além disso na figura 4.1(b), é possível ver a substituição dos itens pelos valores criados para os mesmos onde, por exemplo, para o atributo *Estado*, o item MG, para as três transações, se transforma no item F. Como esta é uma tabela de exemplo, não foi aplicada a última etapa de corte dos itens com frequência 1.

ITEM	VALOR
A	Mary Jane Berg
B	Mary J Berg
C	Maryah Jani Berg
D	BH
E	MG
F	12180
G	Sarah J Santos
H	18-11-1950
I	157890

(a)

ID DA TRANSAÇÃO	ITENS						
1	A	NULO	E	F	G	H	I
2	B	D	E	NULO	G	H	I
3	C	D	E	F	G	NULO	NULO

(b)

Figura 4.1. Tabela de Exemplo: Mapeamento.

4.2 Extração dos Conjuntos de Itens Frequentes

Os conjuntos fechados e maximais foram extraídos com o uso dos algoritmos Eclat e Charm (Zaki & Hsiao [2002]), respectivamente. A utilização de conjuntos fechados

advém do fato que existem diversos predicados que estão associados ao mesmo conjunto de transações, o que é desnecessário para o pareamento probabilístico, pois, uma vez que duas transações são selecionadas para a etapa de comparação, não é mais necessário compará-las novamente. Além disso, foi utilizada uma estratégia de avaliação de pares dentro dos blocos para diminuir o custo computacional de se comparar novamente um par enumerado, avaliando se o par já foi previamente comparado em algum outro bloco.

Independente do conjunto de itens escolhido, é necessário o uso de um suporte mínimo, isto é: qual a quantidade de transações mínimas com determinado item ou determinados itens para este conjunto ser considerado frequente?

Embora os conjuntos de itens frequentes sejam capazes por si só de formarem os blocos para o pareamento de registros. ainda existe uma etapa importante para permitir a execução completa do pareamento de registros: escolher os possíveis melhores blocos em termos de pares gerados *versus* pares verdadeiros.

A figura 4.2 mostra todos os conjuntos de itens frequentes extraídos com suporte 2, número mínimo para se formar um par, da tabela mapeada na figura 4.1. É possível observar as diferenças explicadas no capítulo 2 entre as várias representações dos conjuntos de itens frequentes existentes na base de dados. Por exemplo, o conjunto de itens $\{E \ \& \ G\}$ é um conjunto fechado mas não é um conjunto maximal pois existe um subconjunto, $\{D \ \& \ E \ \& \ G\}$, frequente, que também é maximal. Na prática não é possível, utilizando os conjuntos maximais, fazer as comparações das transações 1,2 e 3 com apenas um bloco tendo que se utilizar três blocos para realizar essa comparação o que pode aumentar o número de comparações.

PADRÕES FREQUENTES

Padrão	Transações	Padrão	Transações
D	2,3	H & E	1,2
E	1,2,3	H & G	1,2
F	1,3	I & E	1,2
G	1,2,3	I & G	1,2
H	1,2	E & G	1,2,3
I	1,2	D & E & G	2,3
D & E	2,3	F & E & G	1,3
D & G	2,3	H & I & E	1,2
F & E	1,3	H & I & G	1,2
F & H	1,3	H & I & E & G	1,2
H & I	1,2	H & E & G	1,2
I & E & G	1,2		

(a)

PADRÕES FECHADOS

Padrão	Transações
D & E & G	2,3
F & E & G	1,3
H & I & E & G	1,2
E & G	1,2,3

(b)

PADRÕES MAXIMAIS

Padrão	Transações
D & E & G	2,3
F & E & G	1,3
H & I & E & G	1,2

(c)

Figura 4.2. Tabela de Exemplo: Extração de Conjuntos de Itens.

4.3 Enumeração dos Blocos para Comparação

Cada conjunto de itens frequentes encontrado é considerado então um possível bloco para o pareamento de registros. Entretanto, em grandes bases de dados a quantidade de conjuntos de itens frequentes encontrados é extremamente grande, o que faz com que a capacidade de computação desses blocos seja inviável $O(n^2)$ para entradas na ordem de milhões de registros para cada bloco e é conhecida a existência de repetição de pares dentre diferentes blocos, ou seja, um custo alto para repetir comparações. Para solucionar este problema é necessário voltar a uma das questões listadas na introdução desta dissertação: como escolher os possíveis melhores blocos para alcançar um melhor resultado no pareamento probabilístico de registros?

O modelo visa responder esta questão ao explorar algumas propriedades dos conjuntos fechados de itens frequentes na extração dos blocos. A primeira premissa assumida aqui é a de ordenação, ou seja, os blocos determinados por conjuntos frequentes com maior cardinalidade em número de itens são os blocos que, a princípio, possuem maior probabilidade da ocorrência de registros duplicados. No outro extremo, os blocos com menor cardinalidade em número de transações são os blocos mais fáceis de se computar. Portanto, a ordem considerada na seleção é: primeira ordenação, maior cardinalidade em número de itens no conjunto, segunda ordenação, menor cardinalidade em número de transações. Qual ou quais blocos devem ser avaliados? É um compromisso entre custo e capacidade, uma vez que o tamanho do bloco cresce em ordem linear, mas o custo para computar um bloco cresce em ordem quadrática. A segunda etapa do *CFI Blocking* foi projetada para selecionar os melhores blocos a serem avaliados.

A figura 4.3 representa uma estrutura gerada por alguns conjuntos de itens frequentes encontrados que ocorrem o item E. É possível ver que existe um super conjunto de itens formado pelo item E apenas, que ocorre em 3 transações na base de dados, um sub conjunto, (E & F) que ocorre em 2 transações, outro sub conjunto (E & G) que ocorre em 3 transações e por último um sub conjunto (E & D) que ocorre em 2 transações. Esses conjuntos de itens frequentes foram retirados da tabela 4.2 e possuem suporte $>= 2$.

O algoritmo 2 foi projetado para selecionar dos blocos a serem avaliados. Se o algoritmo selecionar o bloco formado pelos itens (E & D) significa que o algoritmo estará avaliando um espaço de 2 transações. Se o bloco com o item (E) for escolhido, o espaço coberto será de 3 transações. É um compromisso entre custo e capacidade uma vez que o tamanho do bloco cresce em ordem linear mas o custo para computar um bloco cresce em ordem quadrática. É importante ressaltar que ao selecionar o bloco com

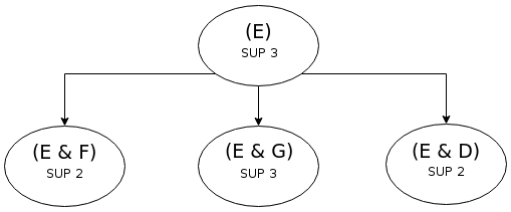


Figura 4.3. Exemplo de Blocos.

o item (E), é possível automaticamente excluir os blocos que são seus subconjuntos, uma vez que todas as transações estarão cobertas pelo bloco selecionado. Entretando isto gera um custo maior de comparações uma vez que serão comparados todos os registros. Por isto a ordem em que os blocos são avaliados pelo algoritmo de seleção é de extrema importância para o *CFI Blocking*.

Algorithm 2 Enumeração dos Blocos para Comparação.

Precondition: *padroes* frequentes gerados pela extração de conjuntos de itens frequentes

```
1: function SelecionarBlocos(padroes)
2:     limiar ← Inteiro
3:     S_b ← Conjunto de Blocos
4:     S_td ← Lista de Blocos Selecionados
5:     Sort(padroes) ▷ Ordenar os conjuntos de itens frequentes por ordem de maior quantidade de itens no conjunto de itens e menor número de transações
6:     for i ← 1 to S_b.Lenght do
7:         if S_bi.size()>limiar then
8:             Continue
9:         else
10:            if S_bi.isSubSet(de algum bloco previamente executado) then
11:                Continue
12:            else
13:                S_td.add(S_bi)
14:            end if
15:        end if
16:    end for
17:    return S_td
18: end function
```

A seleção é baseada em um parâmetro implementado como *limiar*, ou limite superior. Este parâmetro define para o algoritmo qual o tamanho máximo, em número de transações, permitido para um bloco. Portanto, dado um bloco X, este bloco só será avaliado se a cardinalidade de $X <= limiar$ especificado e se X não for subconjunto,

em termos de instâncias de atributos, de algum bloco avaliado anteriormente.

Assim é possível controlar o número de comparações de acordo com a capacidade computacional. Caso o interesse seja de se efetuar mais comparações, o *limiar* deve ser maior, ou seja, permitir a avaliação de blocos maiores. Isto fará com que o número de blocos comparados seja menor uma vez que vários blocos menores se tornarão subconjunto de um bloco mas, ao mesmo tempo, aumentará o número de comparações, uma vez que mais transações estarão dentro de um mesmo bloco, gerando mais pares.

Caso o interesse seja de efetuar menos comparações, o limiar deverá ser menor, ou seja, permitir a execução apenas de blocos pequenos e, consequentemente, com menos pares para comparação. Existe uma certa liberdade de escolha do usuário/especialista ao especificar estes parâmetros mas este compromisso deve ser considerado. Estas escolhas são avaliadas na seção 5.

4.4 Análise de Complexidade

Para o mapeamento de dados a é de $O(n \text{ x } m)$ em tempo uma vez que para cada atributo (m) deve ser percorrido todos os registros da base de dados (n) de entrada. A complexidade é $O(n \text{ x } m)$ em espaço se todos os itens forem distintos, pois todos devem ser armazenados.

Para encontrar os conjuntos fechados de itens frequentes utilizando o algoritmo Eclat, a complexidade é de $O(n * 2^i)$, no pior caso, uma vez que podem existir 2^i conjuntos de itens frequentes e a interseção de dois conjuntos de transações é $O(n)$. A complexidade de espaço é $O(2^i/i)$ (Zaki & Meira Jr [2014]).

Para os conjuntos maximais de itens frequentes a complexidade para cada busca é de $O(MFI)$ no pior caso, onde MFI é o número de padrões maximais encontrados. Além disso, é necessário um tempo de $O(M)$ para checar se esses padrões são maximais ou não, no pior caso o tempo de execução do algoritmo pode ser de $O(MFI \text{ x } M)$ (Gouda & Zaki [2005])

Para a seleção, a complexidade é de $O(n)$ em tempo e $O(n)$ em espaço, caso todos os blocos sejam selecionados. Para a criação e comparação é de aproximadamente $O(n^2)$ dentro de cada bloco. Para a etapa de comparação o custo de tempo é $O(n)$ de em número de atributos e de $O(1)$ em espaço.

Capítulo 5

Avaliação e Resultados

Este capítulo apresenta a aplicação das métricas citadas no capítulo 3 com o intuito de avaliar *CFI Blocking* nos quesitos eficácia e eficiência em relação a enumeração dos blocos para o pareamento probabilístico. A seção 5.1 apresenta os parâmetros utilizados na execução do *CFI Blocking* bem como do método de blocagem padrão. A seção 5.4 apresenta um conceito de deduplicação por referência ou verdade absoluta utilizado na avaliação. A seção 5.4.2 faz uma análise de precisão e revocação. A seção 5.5 faz uma análise de tempo de execução. A seção 5.6 faz uma análise de grupos e trajetórias. Todas as análises foram realizadas em relação aos blocos gerados pelo *CFI Blocking*, pelos conjuntos maximais de itens frequentes e pelo método de blocagem padrão.

5.1 Parâmetros para Avaliação

Conforme explicado no capítulo 4, o *CFI Blocking* exige a configuração de alguns parâmetros para sua execução. Assim como o método atual de blocagem exige a escolha, por parte do pesquisador, de uma estratégia de blocagem o *CFI Blocking* exige a escolha de um limite superior e de um limite inferior para a enumeração dos blocos. Esta seção apresenta os parâmetros, as estratégias e os valores para os cálculos de comparação utilizados nesta dissertação.

A estratégia de blocagem padrão utilizada foi um combinado de 5 estratégias dado por: {cns}, {primeiro nome, último nome, ano de nascimento, sexo}, {primeiro nome da mãe, sobrenome da mãe, data de nascimento}, {município de nascimento, data, sexo}, {primeiro nome, último nome, primeiro nome da mãe, último nome da mãe, sexo} que foram avaliadas no trabalho de Coeli & Camargo Jr. [2002] e utilizadas nos trabalhos de de Queiroz [2007], Pereira et al. [2015]. Assim, para o método de blocagem padrão, foram gerados n blocos distintos com registros que são agrupados por possuir os itens de

qualquer uma dessas estratégias definidas em comum. Por exemplo, o registro com {primeiro nome=Guilherme,ultimo nome=Junior, data de nascimento=04/08/1980 nascimento=1980, sexo=Masculino,cns=202020, cidade=Belo Horizonte} pode ser agrupado com qualquer outro registro que tenha {cns=202020} por causa da estrategia 1, com qualquer registro que tenha {primeiro nome=Guilherme,ultimo nome=Júnior, ano de nascimento=1980, sexo=Masculino,cns=202020} e assim sucessivamente.

Para a execução do *CFI Blocking*, o primeiro parâmetro de avaliação a ser ajustado é o suporte mínimo para enumeração dos conjuntos. Valores baixos de suporte tendem a gerar blocos menores, mas em maior número, também caracterizados por predicados mais específicos e com maior expectativa de serem pares verdadeiros. Assim foram estabelecidos os valores de suporte mínimo com variação de 2 até 5 para análises e experimentos. É importante ressaltar que o valor de suporte é sempre mínimo e que os grandes blocos serão gerados da mesma maneira.

O segundo parâmetro é o limite superior de execução para o tamanho de um bloco, o limiar. Este limiar tem o intuito de permitir a execução do pareamento com baixo custo computacional, sendo assim foram adotados valores de *sup* até 20 para análise e experimentos uma vez que executar um bloco com limiar=100 é o mesmo que executar um bloco de sup=100 o que foi identificado como inviável na etapa anterior.

Para efeitos de avaliação, foram realizados testes utilizando o sistema de codificação de nomes: soundex em sua versão brasileira implementada no trabalho de (dos Santos [2008]) e utilizada pra codificação dos nomes similares *versus* itens. Na avaliação utilizando soundex, nomes que supostamente poderiam ser o mesmo foram detectados através da fonética e foram mapeados como um mesmo item, por exemplo, WAGNER e VAGNER foram considerados como iguais para a formação dos blocos.

Tabela 5.1. Tabela de Itens Gerados

ITENS				
Atributo	**Usado**	**Eliminado**	**Total**	**Soundex**
Primeiro Nome	27228	34823	62051	2706
Sobrenome	62718	133046	195764	3853
Último Nome	12816	16213	29029	2388
Primeiro Nome da Mãe	11314	16889	28203	1928
Último Nome da Mãe	8109	12328	20437	1973
Data de Nascimento	37244	1371	38615	37244
Sexo	2	0	2	2
Cpf	23953	133183	157136	23953
Cns	25506	290723	316229	25506
Cidade	519	143	662	519
Cep	20562	2062	22624	20562
Total de Itens Utilizados	229971			120631

Foram encontrados 229.971 itens únicos na base de dados e com o agrupamento

por soundex foram encontrados 120.631 itens distintos como é possível ver na tabela 5.1. Para os atributos com a utilização do soundex é possível perceber uma redução de até duas ordens de grandeza no agrupamento dos nomes e como impacto disso, é possível perceber um número menor de blocos consequentemente com mais registros.

A etapa subsequente à etapa de blocagem no pareamento probabilístico de registros é a etapa de comparação. Conforme explicado no capítulo 2, para esta etapa é necessário definir os valores estatísticos das comparações probabilísticas campo a campo.

Estes valores são utilizados pelo modelo de Fellegi & Sunter [1969] para o pareamento de registros. Cada atributo A_i tem um valor estatístico e probabilístico associado para os casos de *match* onde os registros concordam neste atributo e para os casos de *unmatch* onde os registros discordam no atributo A_i. Os valores utilizados na etapa de comparação para o cálculo da probabilidade entre os pares foram os valores utilizados anteriormente nos trabalhos dos especialistas em saúde (de Queiroz [2007], Pereira et al. [2015]), conforme a tabela 5.2.

Tabela 5.2. Valores Estatísticos para Comparação

Atributo	**M**	**U**	**Missing**	**Standard**	**Tipo**	**Valor de Aproximação**
nomep	0.98	0.012	1.59	9.2103	approx	0.90
nomem	0.92	0.008	0.7	9.2103	approx	0.85
nomeu	0.95	0.29	1.35	9.2103	approx	0.88
nome mae p	0.75	0.05	0.99	9.2103	approx	0.90
nome mae u	0.70	0.05	1.07	9.2103	approx	0.88
sexo	0.98	0.51	-1.84		exata	
data nascimento	0.90	0.01	-1.84		exata	
cpf	0.80	0.0053	-0.58		exata	
cns	0.80	0.0053	-0.58		exata	
municipio	0.77	0.0083	-0.58	9.2103	exata	
cep	0.77	0.0053	-0.58		exata	

A probabilidade de M(concordância) é o valor para dado quando um par é considerado verdadeiro e o atributo deste par concorda em ambos os registros. A probabilidade U(discordância) é o valor assumido quando um par é considerado verdadeiro e o atributo deste par discorda nos registros. A probabilidade *Missing* é dada para um atributo que for nulo em um dos registros do par comparado. A probabilidade *Standard* é o valor padrão para um atributo que possui tabela de frequência mas não é possível encontrar o item nesta tabela. A probabilidade de *aproximação* é a probabilidade dada para definir se o atributo é considerado igual entre os registros. De acordo com essas probabilidades, os pares são classificados como verdadeiros, falsos ou duvidosos (zona cinzenta). Os dados demonstrados nesta seção utilizam apenas os pares considerados como verdadeiros, acima da probabilidade previamente calculada.

Para incremento ou decremento desses valores ainda foram utilizadas tabelas de frequências para inferir equilíbrio aos pesos uma vez que encontrar um par com um nome comum, por exemplo, Maria em uma cidade grande como São Paulo não tem o mesmo significado estatístico de encontrar um par com um nome como, Jaciara na mesma cidade de São Paulo. Também foram utilizados algoritmos de comparação por proximidade para atributos do tipo nome pois estes atributos estão mais sujeitos a erros de digitação.

5.2 Avaliações

O *CFI Blocking* foi avaliado e comparado com os conjuntos maximais e com o método de blocagem tradicional/padrão. Para a realização desta avaliação foi executado o ciclo completo de pareamento de registros alterando se apenas o método de blocagem. O resultado obtido foi contrastado em um arcabouço de precisão x revocação, métricas citadas no capítulo 3. Estas métricas avaliam duas capacidades distintas dos algoritmos: quão bom é um bloco em termos de enumerar pares verdadeiros e a capacidade do método de blocagem em encontrar os pares verdadeiros em relação ao total de pares verdadeiros existentes, respectivamente. A precisão aqui avaliada é representada pela métrica, Valor preditivo positivo enquanto a revocação é representada pela métrica Sensitividade.

Dado as características do problema e dos dados, é perceptível na literatura que a precisão de um bloco é normalmente baixa em relação a sua revocação. Isto é intuitivo uma vez que registros parecidos nem sempre são iguais. Para um bloco ser considerado bom, é importante haver um equilíbrio entre estas métricas, ou seja, é importante ser preciso mas mais importante que isso é que haja revocação sendo possível encontrar os pares duplicados em um tempo computacional hábil.

Além disso, os parâmetros de avaliação referentes ao que é um par verdadeiro estão descritos na seção 5.1 e se aplicam a comparação em ambos os métodos de blocagem. A diferença nos resultados descritos nesta dissertação se da pela capacidade do método de blocagem indicar o par para comparação, isto é: dentre os blocos formados existirem os dois registros possivelmente duplicados que formam este par. É vedada a hipótese de um par ser considerado falso por um método e verdadeiro por outro.

5.3 Base de Dados sintética

O CFI Blocking foi executado em ambiente controlado de dados sintético e avaliado baseado nos atributos e parâmetros para comparação listados na tabela 5.2. Foi possível perceber que o CFI Blocking foi superior a blocagem por mineração de conjuntos maximais e também em relação a blocagem tradicional em precisão e revocação, obtendo até 97% de revocação enquanto os outros alcançaram 91% e 64%, respectivamente. Estes valores servem como base para justificativa de utilizar o CFI Blocking nas bases de dados reais. O gráfico da figura 5.1 ilustra a revocação dos algoritmos.

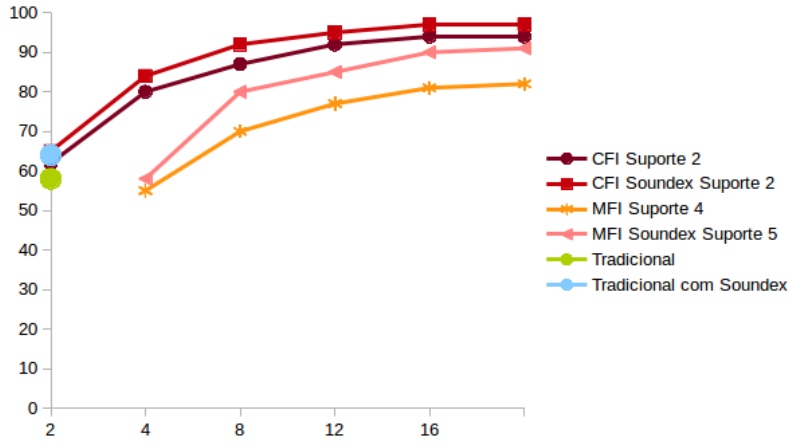


Figura 5.1. Sintético - Pares x Revocação.

5.4 Base de Dados Real

No mundo ideal, comparar todos os registros de uma base de dados permite eliminar a etapa de blocagem mas isso exige um esforço de comparação na ordem de $O(n^2)$ o que é inviável até mesmo para bases de dados pequenas como a amostra utilizada: 1.666.921 milhões de registros. Sendo assim, temos um problema: como saber o número total de registros duplicados, e.g pares verdadeiros dentro desta base de dados? Esta informação é essencial para à avaliação de todas as métricas citadas no capítulo 3. Para resolver este problema foram realizadas algumas tentativas de estimar o número de pares verdadeiros totais existentes nesta amostra extraída.

5.4.1 Pares Verdadeiros Totais

Foram realizados testes com o *CFI Blocking* com suporte mínimo 2, ou seja, o item precisou ocorrer em pelo menos duas transações para se gerar um bloco o que permite

a execução até a capacidade máxima da máquina utilizada, ou seja, permitir a enumeração de blocos ineficientes para tentar mensurar a quantidade de pares verdadeiros totais existente na base de dados utilizada. Com isto, foi possível encontrar um alto número de pares de registros formado por diferentes instâncias de atributos a fim de verificar o número máximo de pares verdadeiros existentes.

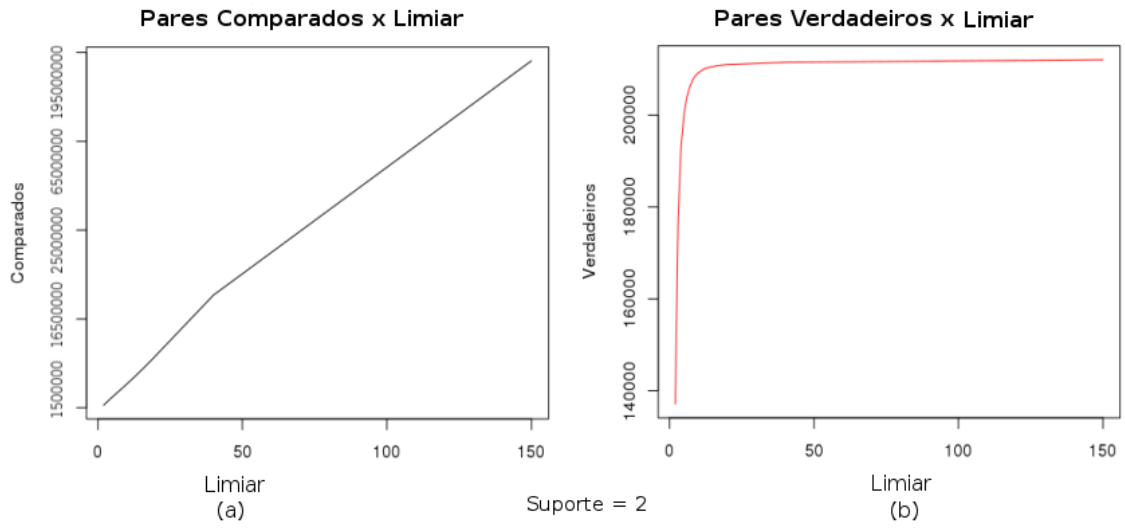


Figura 5.2. Pares x limiar.

Durante esta execução foi possível perceber conforme o gráfico da figura 5.2(b), que em determinado momento, $limiar >= 15$, o crescimento do número de pares verdadeiros tornou-se marginal em relação ao crescimento do número de pares comparados. Isto é explicado pelo fato de que o limiar é o limite superior inerente ao número de transações existentes em um bloco. Este limiar determina se um bloco será executado ou não baseado no seu número de transações, ou seja, quanto maior é este limiar, maior é o número de blocos que serão considerados para execução.

Assim, embora seja possível encontrar mais pares verdadeiros com o aumento do limiar e consequentemente, de pares comparados, é possível perceber no gráfico da figura 5.2, que o número de pares verdadeiros encontrados tende estabilizar, ou seja, o número de comparações cresce exponencialmente mas o número de pares verdadeiros cresce linearmente.

Com esta execução, o compromisso foi de encontrar o número de pares verdadeiros mais próximo do real, assumiu-se então o valor encontrado no fim da execução, com $limiar = 150$ sendo comparados aproximadamente 196 milhões de pares o qual resultou em 212.930 pares verdadeiros definidos como a verdade absoluta. A verdade absoluta

(Thomas & Thorne [1992]) é um termo usado para se referir a informação obtida através de observação direta ou em *machine learning* um valor reconhecido e provado para uma base de dados.

5.4.2 Precisão x Revocação

O gráfico da figura 5.3 apresenta uma análise de precisão dos blocos gerados nos diferente testes realizados. As figuras a e c, chamadas aqui de grupo 1, representam os conjuntos testados sem a utilização da técnica de *soundex* no mapeamento dos dados enquanto as figuras b e d, chamadas de grupo 2, representam os algoritmos com o uso da técnica de *soundex*.

É possível perceber que os conjuntos do grupo 1 são mais precisos, aproximadamente (9%) representado nas figuras a e c, do que os conjuntos do grupo 2, aproximadamente (6%) representado nas figuras b e d. Esta é uma característica esperada uma vez que, sem a utilização do *soundex*, os grupos são formados por itens exatamente iguais o que consequentemente gera uma maior precisão.

Todos os conjuntos, independentemente do grupo ao qual pertencem, apresentam uma tendência de queda na precisão com o aumento do limiar. Essa característica é decorrente do crescimento do tamanho dos blocos, ou seja, em determinado momento o número de pares gerados cresce com maior velocidade que o número de pares verdadeiros existentes nos blocos analisados.

Em todas os testes realizados na figura 5.3 os blocos gerados pelo *CFI Blocking* e o conjunto de maximais, possuem precisão maior do que os testes realizados com o método padrão de blocagem. Este resultado é explicado pela estratégia de enumeração dos blocos no *CFI Blocking*: através da instâncias dos atributos em comum das transações na base de dados, enquanto que o método de blocagem padrão enumera os blocos de acordo com os atributos selecionados pelo pesquisador, para todos os registros, e não para instâncias específicas dos atributos.

Os resultados com melhor precisão são apresentados no gráfico da figura 5.4. Os melhores resultados foram encontrados utilizando o suporte 2 e para quase todos os experimentos realizados o *CFI Blocking* possui uma precisão maior que a precisão do método de blocagem padrão. Entretanto resultado encontrado, para precisão, foi com a utilização dos conjuntos maximais com suporte 2 cujo a precisão alcança 9.06% enquanto que o melhor caso do método de blocagem padrão foi preciso em 1.74% das vezes.

O *CFI Blocking* foi compatível com os conjuntos maximais com uma precisão de 9.04% , muito próxima do valor obtido pelos conjuntos maximais e superior ao

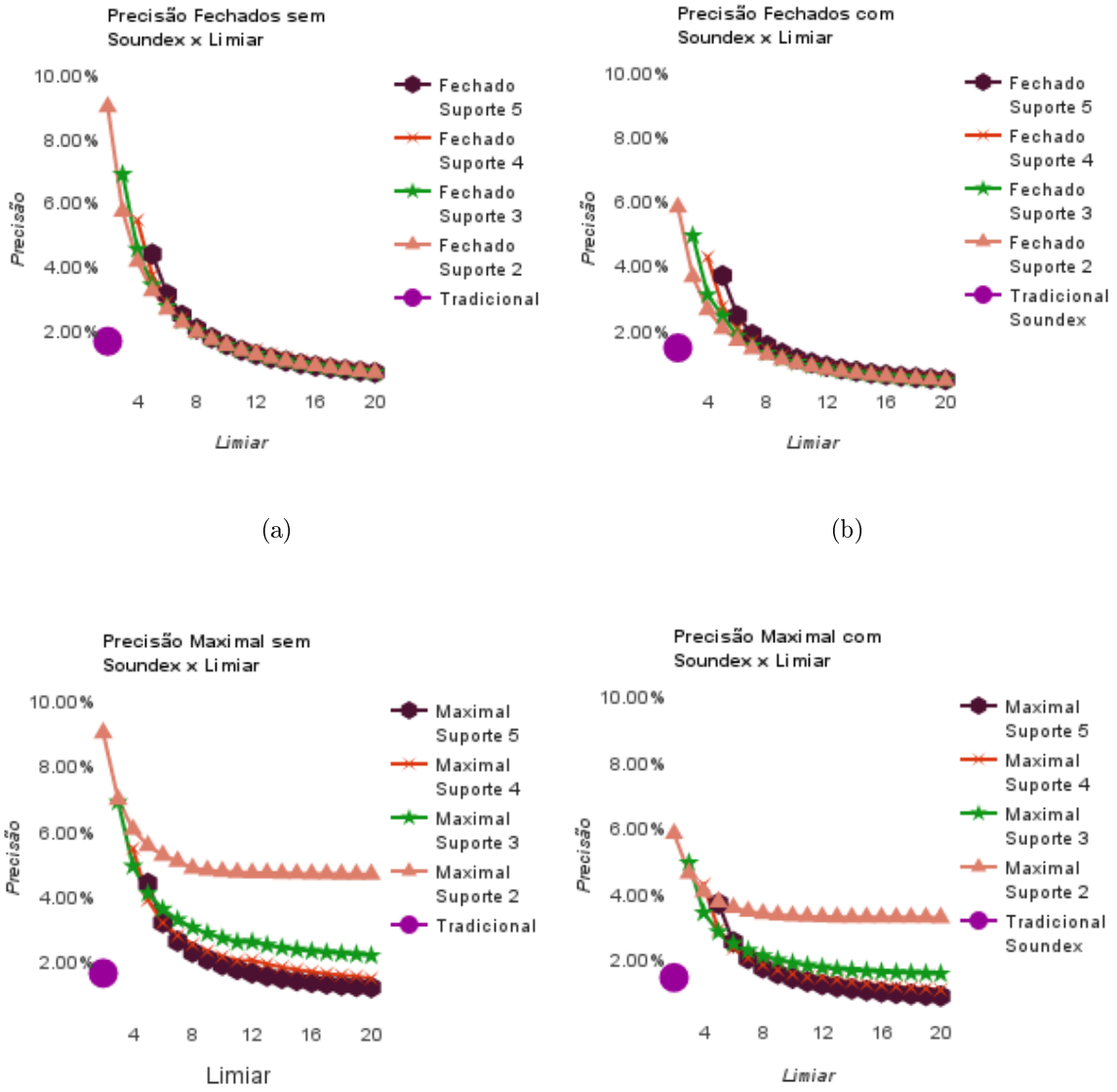


Figura 5.3. Precisão x limiar

método de blocagem padrão. Nesta mesma figura é possível observar uma queda na precisão à medida que o limiar aumenta. Isto ocorre pela característica do algoritmo no qual valores altos de limiar permitem a avaliação de grandes blocos de registros, o que consequentemente gera um elevado número de comparações e de pares falsos, diminuindo a precisão.

Os resultados demonstram que os testes realizados utilizando o algoritmo de *soundex* para o mapeamento dos dados são menos precisos se comparados com o mapea-

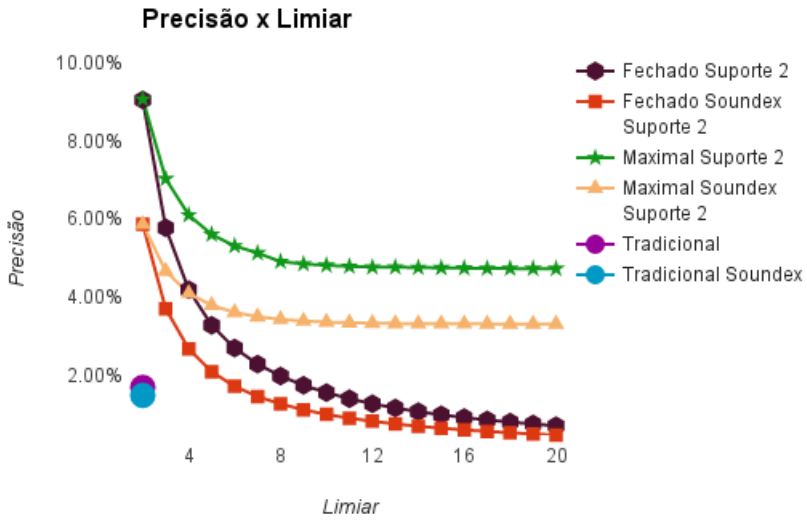


Figura 5.4. Precisão x limiar.

mento simples, ou seja, sem a utilização do *soundex*. Isto ocorre porque o agrupamento de nomes por proximidade aumenta o número de elementos em um mesmo bloco, que podem não representar a mesma entidade no mundo real.

Os conjuntos maximais são mais precisos se comparados ao *CFI Blocking*, isto ocorre dada a característica de formação mais restrita de blocos em número de instâncias que compõem um bloco. O número de pares gerados é relativamente menor nestes conjuntos. Entretanto não permitir super conjuntos faz com que o aumento do limiar, não altere o número de pares verdadeiros encontrados. Assim, existe um crescimento no número de pares gerados mas com aumento insignificante no número de pares verdadeiros encontrados e isso faz com que seja mantida a proporção no valor de precisão o que causa uma falsa impressão de que a precisão seja estável com a variação do limiar. O *CFI Blocking*, ao executar super conjunto gera um maior número de pares comparados com o aumento do limiar fazendo com que a tendência de queda na precisão aconteça em testes com maiores volumes de dados em um mesmo bloco.

O gráfico da figura 5.5 exibe 4 sub gráficos separados em dois grupos; os gráficos a e c pertencem ao grupo 1 de conjuntos testados sem a utilização do mapeamento com *soundex* e os gráficos b e d pertencem ao grupo 2 de conjuntos testados com a utilização do mapeamento com *soundex* para avaliação da revocação dos algoritmos utilizados.

Os testes realizados com o grupo 1 apresentam uma revocação ligeiramente menor que os testes realizados com o grupo 2, essa diferença é explicada pela capacidade que

o grupo 2 possui de agrupar registros por aproximação e consequentemente comparar mais transações e encontrar mais pares verdadeiros. No método de blocagem padrão também é percebida essa melhoria, enquanto a revocação é de 65.61% sem a utilização do mapeamento com *soundex*, utilizando *soundex* foi encontrado uma revocação de 68.05%.

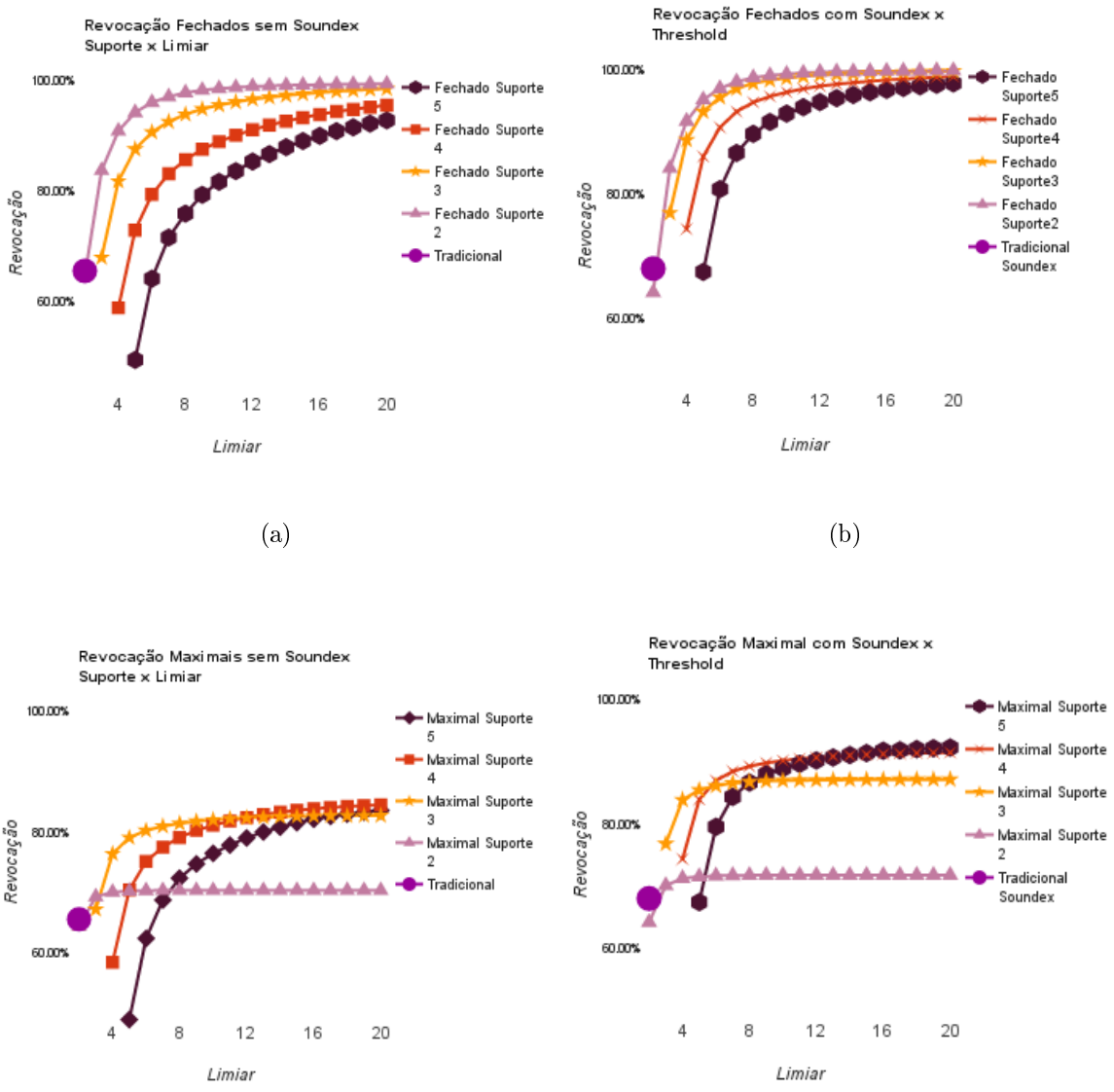


Figura 5.5. Revocação x limiar

O *CFI Blocking* foi superior ao método de blocagem padrão em revocação, tendo encontrado, em seu melhor resultado, 100% dos pares verdadeiros em relação a Verdade

Absoluta enquanto o método de blocagem padrão encontrou 68.47%. Esses resultados podem ser vistos nos gráficos da figura 5.6, que representa o resultado das quatro variações analisadas: o CFI Blocking, os conjuntos maximais, utilizando *soundex* ou não, com os melhores resultados encontrados. Além disso, foi possível perceber que quanto maior o limiar, maior é a revocação alcançada por uma estratégia de blocagem.

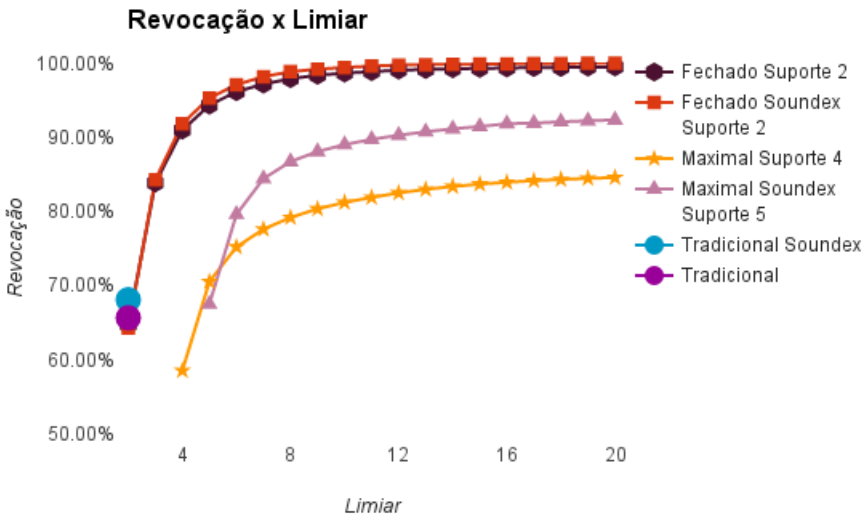


Figura 5.6. Revocação x limiar.

O fenômeno observado em relação a métrica de precisão para os algoritmos com conjuntos maximais também pode ser observado em relação a métrica de revocação. Ao se utilizar valores de suporte pequenos como 2 ou 3, é possível perceber na figura 5.6 - c e d que o algoritmo tende a ter um comportamento assimptótico no valor de revocação, isto ocorre pois conjuntos maximais com n transações restringem a formação de conjuntos maximais com suporte maior n, ou seja, ao aumentar o limiar o número de pares verdadeiros encontrados é apenas ligeiramente maior, pois o número de pares gerados também é somente ligeiramente maior devido a condição de restrições dos blocos. Sendo assim, para os conjuntos maximais com suporte pequeno, aumentar o limiar e permitir blocos maiores não causa diferenças significativas no número de pares verdadeiros encontrados.

A média harmônica retrata uma relação de equilíbrio entre as métricas de precisão e revocação, o gráfico da figura 5.7 exibe os testes executados para o *CFI Blocking* com suporte 2, os com melhores resultados em termos de revocação. Foi possível observar que o formato da curva no gráfico é similar ao formato de precisão uma vez que os valores de precisão são menores que os valores de revocação e isto influência na média

harmônica. O *CFI Blocking* alcança resultados maiores mas ao longo do limiar é possível ver uma queda na média harmônica. Esta queda é explicada, também, pela influência da métrica de precisão nessa média uma vez que limiares maiores aumenta o espaço de busca diminuindo a precisão, o que não ocorre com os conjuntos maximais. O *CFI Blocking* e os conjuntos maximais possuem resultados melhores que os método de blocagem padrão.

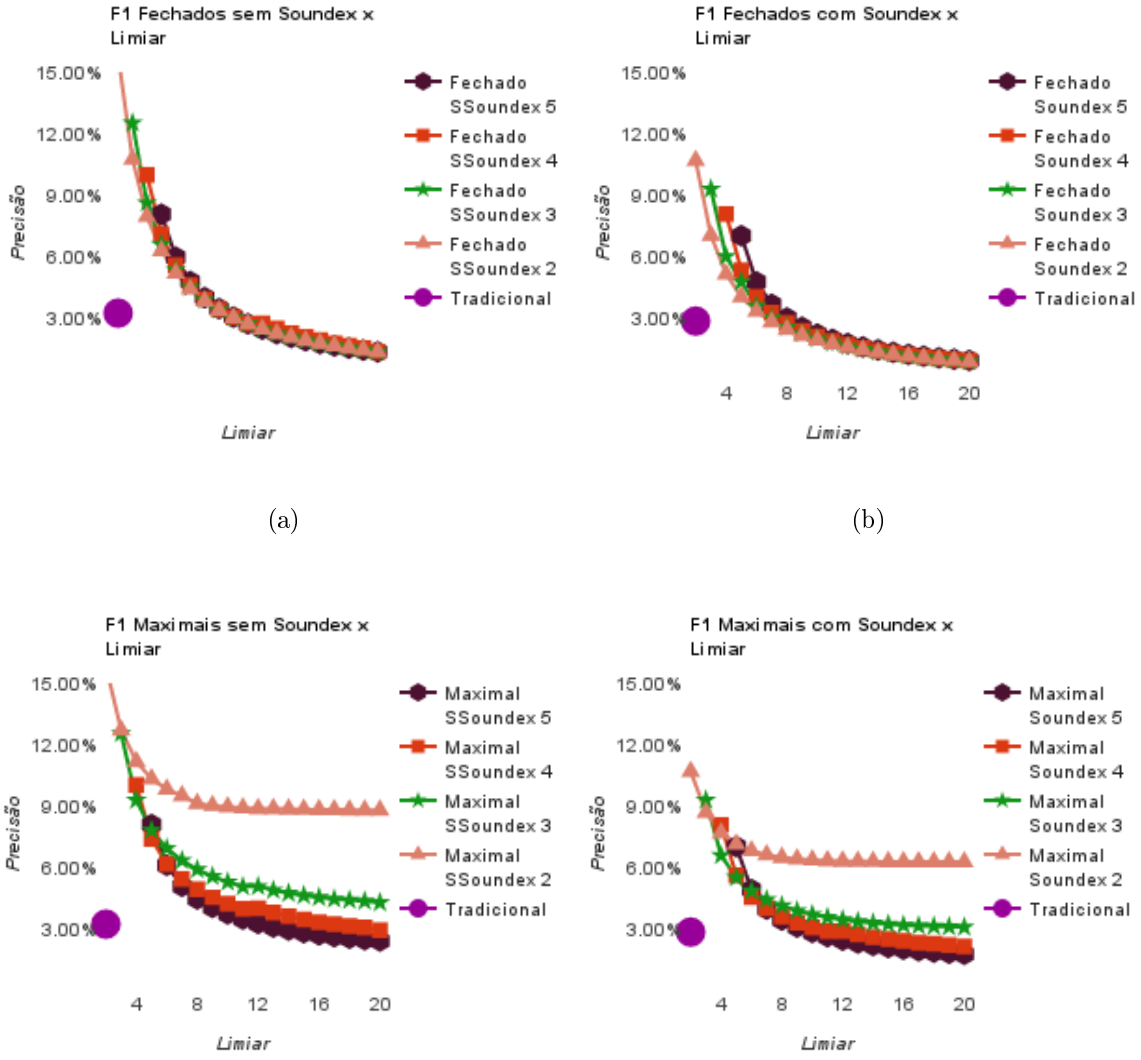


Figura 5.7. F1 x limiar

Embora os algoritmos tenham se mostrado eficientes, com resultados melhores que os métodos tradicionais em precisão, revocação e com a média harmônica entre as

duas métricas, é importante avaliar o quesito tempo de execução. Esta análise está descrita na seção 5.5.

5.5 Análise de Tempo

Ao analisar a qualidade do algoritmo proposto é importante verificar a variável tempo de execução. Qual o custo da busca pelos blocos no novo método em relação ao método de blocagem padrão? O algoritmo é eficiente? Nesta seção são apresentados os resultados dos testes e análises em relação ao tempo de execução com o intuito de tentar responder essas perguntas.

Tabela 5.3. Testes x Tempo

Conjunto	Blocagem	Comparação	Total	Precisão	Revocação	F1
MaxCSoundex 5-5	0:00:56	0:01:29	0:02:25	3.74%	67.47%	7.08%
MaxSSoundex 5-5	0:01:32	0:01:03	0:02:35	4.45%	48.97%	8.15%
CFI CSoundex 5-5	0:01:20	0:01:33	0:02:53	3.73%	67.52%	7.08%
MaxCSoundex 4-4	0:01:24	0:01:31	0:02:55	4.31%	74.39%	8.15%
CFI SSoundex 5-5	0:01:27	0:01:29	0:02:56	4.43%	49.28%	8.13%
CFI SSoundex 3-3	0:02:48	0:00:49	0:03:37	6.92%	68.05%	12.57%
CFI CSoundex 4-4	0:01:01	0:02:37	0:03:38	4.31%	74.42%	8.14%
CFI SSoundex 4-4	0:02:15	0:01:25	0:03:40	5.49%	58.68%	10.04%
MaxCSoundex 5-6	0:01:29	0:02:13	0:03:42	2.57%	79.58%	4.99%
MaxSSoundex 4-4	0:02:54	0:00:51	0:03:45	5.52%	58.48%	10.08%
CFI CSoundex 5-6	0:01:59	0:02:05	0:04:04	2.49%	80.86%	4.83%
MaxCSoundex 3-3	0:02:36	0:01:33	0:04:09	4.98%	76.86%	9.35%
Tradicional	0:00:45	0:03:19	0:04:04	1.70%	65.34%	3.31%
Tradicional CSoundex	0:00:40	0:04:08	0:04:48	1.49%	68.05%	2.92%

* As letras C e S antes da palavra *soundex* indicam respectivamente COM e SEM. Os números indicados após o nome do algoritmo indicam respectivamente: suporte e limiar

A tabela 5.3 contém os resultados dos tempos de execução dos experimentos realizados analisados e seus valores de média harmônica, precisão e revocação. Esta tabela está listada em ordem crescente de tempo de execução e exibe os 12 experimentos mais rápidos em tempo computacional. Os resultados indicados nesta tabela mostram que o CFI Blocking obteve os melhores resultados para precisão (CFI SSoundex 3-3) e revocação (CFI CSoundex 5-6). Os testes com conjuntos maximais foram compatíveis, ou seja, tiveram resultados com tempo de execução próximos ao *CFI blocking* e com revocação e precisão compatíveis. O metódo de blocagem padrão/tradicional foi o pior avaliado em tempo e precisão e possuindo revocação equivalente as piores configurações de execução do CFI Blcoking e dos Conjuntos Maximais. Os resultados presentes na tabela 5.3 são a média para cinco execuções de cada teste. Todos os experimentos realizados estão detalhados nas tabelas presentes no apêndice A.

Tabela 5.4. Trajetórias x Algoritmos

ALGORITMO	AGM	MEDIA GRUPO
CFI SSoundex 5	1.285687951	3.63645
CFI SSoundex 4	1.256833374	3.53941
CFI CSoundex 5 - 5	1.145123919	3.53547
CFI CSoundex 4-12	1.244046019	3.4774
CFI CSoundex 2-20	1.240143515	3.45599
CFI SSoundex 2-20	1.238111183	3.43735
Trad Soundex	1.186950288	2.22651
MaxSSoundex 5-5	1.118683133	2.3799
MaxSSoundex 4-4	1.269450729	2.37017
MaxCSoundex 5-5	1.049410367	2.36152
CFI SSoundex 3	1.249399434	2.3434
CFI CSoundex 3-8	1.242125332	2.32607
MaxCSoundex 4-6	1.234229051	2.32553
MaxSSoundex 3-4	1.215948948	2.32197
MaxCSoundex 3-7	1.198468229	2.29529
Trad	1.181737687	2.24617
MaxSSoundex 2-20	1.043356016	2.17173
MaxCSoundex 2-20	1.052050271	2.1648

* As letras C e S antes da palavra *soundex* indicam respectivamente COM e SEM. Os números indicados após o nome do algoritmo indicam respectivamente: suporte e limiar

5.6 Análise de Trajetórias

Analisar trajetórias é entender o comportamento do resultado do pareamento, uma vez que o objetivo do pareamento é identificar registros únicos é possível rastrear estes registros ao longo do tempo. Esta seção apresenta uma análise dos grupos gerados pelo resultado do pareamento utilizando o *CFI Blocking*, os conjuntos maximais e o método de blocagem padrão.

A tabela 5.4 apresenta os resultados com maior tamanho médio de grupo para cada suporte testado. É possível perceber que os testes realizados com os conjuntos sem a utilização do *soundex* possuem um tamanho médio de grupo maior que os testes utilizando o *soundex*. Este fato pode ser explicado pela revocação dos pares uma vez que ao utilizar o *soundex* foi possível alcançar resultados melhores em revocação e encontrar mais pares únicos o que diminui o tamanho médio do grupo. Os experimentos com um baixo valor de suporte resultam em grupos com um tamanho médio menor, isto também está associado a revocação dos mesmos.

A relação entre a AGM e o tamanho médio dos grupos percebida, é diretamente proporcional, ou seja, quanto mais registros em um mesmo grupo, maior é a média da

AGM do método utilizado. A métrica da AGM é importante para avaliar dentre os pares verdadeiros a não existência de um par pois uma vez que um grupo representa uma entidade única, todos os registros pertencentes a ele devem possuir relações entre si, o que não ocorre para todos os casos. Todos os testes para análise de trajetórias estão detalhados nas tabelas presentes no apêndice B.

5.7 Hardware

Todos estes experimentos foram realizados em um servidor Dell com 120GB de memória ram e 16 núcleos de processamento, com HD de 1TB. Estes experimentos também foram realizados em máquinas com menor potência computacional mas sofreram do problema de dimensão e tamanho dos dados.

Capítulo 6

Conclusão

Esta dissertação apresentou o *CFI Blocking* um algoritmo para enumeração de blocos no pareamento probabilístico de registros através dos valores das instâncias dos atributos de uma base de dados e suas propriedades de fechamentos.

A enumeração de blocos através do *CFI Blocking* se mostrou eficaz e eficiente sendo superior ao método de blocagem padrão/tradicional e sua utilização superou conjuntos maximais e o método de blocagem padrão. Os fatores determinantes de escolha entre os algoritmos foram o tempo de execução e o espaço de busca pois estes conjuntos possuem comportamentos parecidos para os experimentos mais relevantes. Entretanto, por definição, o CFI Blocking permite encontrar blocos com registros por suas instâncias e pode coexistir com super conjuntos de quantidade de registros, isso faz com que haja uma avaliação de registros previamente mais similares, ou seja, encontrando os pares verdadeiros.

Embora os testes com o suporte 2 tenham obtidos melhores resultados de precisão e revocação o custo computacional da execução destes blocos é alto, ou seja, é despendido muito tempo para a execução do algoritmo. Assim, os melhores valores encontrados considerando precisão, revocação e tempo, foram os resultados encontrados com suporte 3 e limiar até 5 para bases de dados com essas características. Para uma blocagem eficiente, e consequentemente um pareamento eficiente, é fundamental garantir a qualidade dos dados. Com o período analisado e os dados utilizados a análise de trajetória mostrou um comportamento compatível e esperado pelos pesquisadores e analistas que utilizam as bases de saúde para pesquisa.

6.1 Trabalhos Futuros

Como trabalhos futuros, foram pontos de melhoria no processo de extração dos conjuntos de itens frequentes adaptados ao pareamento de registros, por exemplo, o limiar utilizado é um limite superior, esta funcionalidade pode ser implementada diretamente no processo de extração dos conjuntos, existe uma técnica chamada de *Dual Search* que pode ser utilizada para efetuar o corte no limite superior e ainda dar um aumentar a velocidade do processo além de estratégias de ordenação dos blocos que aumentem a precisão do algoritmo.

Outra sugestão é a de verificar a significância do método proposto em relação a novas técnicas para blocagem e/ou técnicas que não foram verificadas neste trabalho. Pode se executar também uma melhor análise de trajetória com algoritmos especializados na geração dos ids únicos. Há também uma necessidade de criação de um software utilizando o *CFI Blocking* capaz de realizar a execução completa do pareamento de registros.

Referências Bibliográficas

Baxter, R.; Christen, P. & Churches, T. (2003). A comparison of fast blocking methods for record linkage. Em *ACM SIGKDD '03 Workshop on Data Cleaning, Record Linkage, and Object Consolidation*, pp. 25--27.

Christen, P. (2008). Febrl: A freely available record linkage system with a graphical user interface. Em *Proceedings of the Second Australasian Workshop on Health Data and Knowledge Management - Volume 80*, HDKM '08, pp. 17--25, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

Coeli, C. M. & Camargo Jr., K. R. (2002). Avaliação de diferentes estratégias de blocagem. Em *Rev. Bras. Epidemiol.*, volume 5.

da Cruz Gouveia Mendes, A.; da Silva Junior, J. B.; Medeiros, K. R.; LyraI, T. M.; de Melo Filho, D. A. & de SáI, D. A. (2000). Avaliação do sistema de informações hospitalares (sih/sus) como fonte complementar na vigilância e monitoramento de doenças de notificação compulsória. Em *Informe Epidemiológico do SUS*, pp. 67--86.

de Carvalho, M. G.; Laender, A. H. F.; Gonçalves, M. A. & da Silva, A. S. (2012). A genetic programming approach to record deduplication. *IEEE Trans. Knowl. Data Eng.*, 24(3):399–412.

de Queiroz, O. V. (2007). Relacionamento probabilístico de registros na integracao de sistemas de informacao do sus: O caso da base nacional de dados em terapia renal substitutiva. Mestrado, PPGSP, Faculdade de Medicina, Universidade Federal de Minas Gerais.

dos Santos, W. (2008). Algoritmo paralelo e eficiente para o problema de pareamento de dados. Mestrado, PPGCC, Departamento de Ciência da Computação, Universidade Federal de Minas Gerais.

Evangelista, L. O.; Vilarinho, E. C. C.; da Silva, A. S. & Meira, W. (2009). Blocagem adaptativa e flexível para o pareamento aproximado de registros. Em *SBBD*.

Fayyad, U. M.; Piatetsky-Shapiro, G. & Smyth, P. (1996). Advances in knowledge discovery and data mining. Em Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. & Uthurusamy, R., editores, *American Association for Artificial Intelligence*, capítulo From Data Mining to Knowledge Discovery: An Overview, pp. 1--34. American Association for Artificial Intelligence, Menlo Park, CA, USA.

Fellegi, I. & Sunter, A. (1969). A theory for record linkage. *American Statistical Association Journal*, pp. 1183--1210.

Fonseca, M. G. P.; Coeli, C. M.; Lucena, F. d. F. d. A.; Veloso, V. G. & Carvalho, M. S. (2010). Accuracy of a probabilistic record linkage strategy applied to identify deaths among cases reported to the Brazilian AIDS surveillance database. *Cadernos de Saúde Pública*, 26:1431 – 1438. ISSN 0102-311X.

Goncalves, C. F.; Santos, W.; Flores, L. F.; S.Vilela, M.; Machado, C.; Jr, W. M. & Silva, A. (2008). Avaliação de técnicas paralelas de blocagem para resolução de entidades e deduplicação. Em *IV Workshop em Algoritmos e Aplicações de Mineração de Dados*.

Gouda, K. & Zaki, M. J. (2005). Genmax: An efficient algorithm for mining maximal frequent itemsets. *Data Min. Knowl. Discov.*, 11(3):223--242. ISSN 1384-5810.

Gu, L.; Baxter, R.; Vickers, D. & Rainsford, C. (2003). Record linkage: Current practice and future directions. Relatório técnico, CSIRO Mathematical and Information Sciences.

Hernández, M. A. & Stolfo, S. J. (1998). Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Min. Knowl. Discov.*, 2(1):9--37. ISSN 1384-5810.

Jaro, M. A. (1989). Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. Journal of the American Statistical Association 84:414–420.

Kantardzic, M. (2002). *Data Mining: Concepts, Models, Methods and Algorithms*. John Wiley & Sons, Inc., New York, NY, USA. ISBN 0471228524.

Kenig, B. & Gal, A. (2013). Mfiblocks: An effective blocking algorithm for entity resolution. *Inf. Syst.*, 38(6):908--926. ISSN 0306-4379.

Larose, D. T. (2004). *Discovering Knowledge in Data: An Introduction to Data Mining*. Wiley-Interscience. ISBN 0471666572.

McCallum, A.; Nigam, K. & Ungar, L. H. (2000). Efficient clustering of high-dimensional data sets with application to reference matching. Em *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pp. 169--178, New York, NY, USA. ACM.

McNeill, B.; Kardes, H. & Borthwick, A. (2012). Dynamic record blocking: Efficient linking of massive databases in mapreduce. Em *10th International Workshop on Quality in Databases (QDB), in conjunction with VLDB 2012*.

Migowski, A.; Chaves, R. A. B. M.; Coeli, C. M.; Ribeiro, A. L. P.; Tura, B. R.; Kuschnir, M. C. C.; Azevedo, V. M. P.; Floriano, D. B.; Magalhãpoundses, C. A. M.; Pinheiro, M. C. C. M. & Xavier, R. M. d. A. (2011). Acurácia do relacionamento probabilístico na avaliação da alta complexidade em cardiologia. *Revista de Saúde Pública*, 45:269 – 275. ISSN 0034-8910.

Newcombe, H. B. (1967). Record linking: The design of efficient systems for linking records into individual and family histories. *American Journal of Human Genetics*, 19(3):335--359.

Nin, J.; Muntés-Mulero, V.; Martínez-Bazan, N. & Larriba-Pey, J. L. (2007). Semantic blocking for record linkage. Em *Proceedings of the 2007 Conference on Artificial Intelligence Research and Development*, pp. 141--149, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Pereira, R. G.; Leal, G. S.; Dias, L. V.; Acúrcio, F.; Junior, A. A. G.; Cherchiglia, M. & Gurgel, E. I. (2015). Unified database creation applied to public brazilian health information systems from the hospital, outpatients and mortality information systems. Em *ISPOR 20th Annual International Meeting*.

Queiroz, O. V. d.; Guerra, A. A.; Machado, C. J.; Andrade, E. L. G.; Meira Júnior, W.; Acúrcio, F. d. A.; Santos Filho, W. d. & Cherchiglia, M. L. (2009). A construção da Base Nacional de Dados em Terapia Renal Substitutiva (TRS) centrada no indivíduo: relacionamento dos registros de óbitos pelo subsistema de Autorização de Procedimentos de Alta Complexidade (Apac/SIA/SUS) e pelo Sistema de Informações sobre Mortalidade (SIM) - Brasil, 2000-2004. *Epidemiologia e Serviços de Saúde*, 18:107 – 120. ISSN 1679-4974.

Thomas, G. & Thorne, R. E. (1992). Fisheries acoustics current status of training and education in fisheries acoustics. *Fisheries Research*, 14(2):135 – 141. ISSN 0165-7836.

Wang, Y. R. & Madnick, S. E. (1989). The inter-database instance identification problem in integrating autonomous systems. Em *Proceedings of the Fifth International Conference on Data Engineering, February 6-10, 1989, Los Angeles, California, USA*, pp. 46–55. IEEE Computer Society.

Zaki, M. J. & Hsiao, C.-J. (2002). CHARM: An efficient algorithm for closed itemset mining. Em *2nd SIAM International Conference on Data Mining*.

Zaki, M. J. & Meira Jr, W. (2014). *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, New York, NY, USA. ISBN 0521766338, 9780521766333.

Apêndice A

Tabelas de Tempo x Revocação x Limiar

Tabela A.1. Fechados sem Soundex - Tempo x Revocação x Limiar

FECHADOS SEM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo
2	-	-	-	-	-	-	64.39%	0:25:46
3	-	-	-	-	68.05%	0:03:37	83.48%	0:32:16
4	-	-	58.68%	0:03:40	81.87%	0:05:48	90.59%	0:35:45
5	49.28%	0:02:56	72.70%	0:05:06	87.67%	0:07:30	93.94%	0:37:32
6	63.94%	0:04:37	79.13%	0:06:23	90.75%	0:08:38	95.75%	0:38:57
7	71.36%	0:06:00	82.87%	0:07:27	92.64%	0:10:06	96.77%	0:39:52
8	75.68%	0:06:57	85.43%	0:08:17	93.99%	0:10:37	97.51%	0:40:57
9	79.08%	0:08:01	87.27%	0:09:15	94.95%	0:11:18	97.96%	0:41:51
10	81.42%	0:08:48	88.67%	0:10:10	95.65%	0:12:01	98.25%	0:42:49
11	83.35%	0:09:36	89.80%	0:11:04	96.20%	0:12:57	98.45%	0:43:42
12	85.08%	0:10:27	90.82%	0:11:51	96.71%	0:13:37	98.65%	0:44:31
13	86.42%	0:11:12	91.64%	0:12:28	97.08%	0:14:18	98.74%	0:45:25
14	87.67%	0:11:44	92.37%	0:13:15	97.41%	0:14:59	98.83%	0:46:20
15	88.78%	0:12:35	93.01%	0:14:22	97.67%	0:16:01	98.88%	0:47:26
16	89.71%	0:13:22	93.58%	0:14:50	97.93%	0:17:07	98.94%	0:48:18
17	90.56%	0:14:22	94.09%	0:15:47	98.15%	0:17:30	99.00%	0:49:20
18	91.25%	0:15:40	94.49%	0:16:48	98.31%	0:19:07	99.02%	0:50:30
19	91.93%	0:16:09	94.90%	0:18:02	98.48%	0:21:40	99.06%	0:51:09
20	92.51%	0:17:24	95.24%	0:18:45	98.62%	0:24:47	99.08%	0:57:26

Tabela A.2. Fechados com Soundex - Tempo x Revocação x Limiar

FECHADOS COM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo
2	-	-	-	-	-	-	64.22%	0:14:37
3	-	-	-	-	77.02%	0:04:00	84.24%	0:18:23
4	-	-	74.42%	0:03:38	88.79%	0:04:48	91.85%	0:20:49
5	67.52%	0:02:53	86.12%	0:05:47	93.37%	0:07:15	95.29%	0:23:51
6	80.86%	0:04:04	90.81%	0:07:14	95.69%	0:08:25	97.10%	0:25:16
7	86.69%	0:04:45	93.36%	0:08:12	97.08%	0:09:44	98.18%	0:26:45
8	89.78%	0:05:59	94.86%	0:09:40	97.94%	0:10:48	98.85%	0:27:46
9	91.70%	0:06:46	95.82%	0:11:06	98.44%	0:12:20	99.19%	0:29:50
10	93.04%	0:07:53	96.50%	0:12:20	98.78%	0:13:17	99.42%	0:30:42
11	94.08%	0:08:37	97.05%	0:13:09	99.06%	0:14:36	99.61%	0:31:19
12	94.91%	0:09:06	97.49%	0:14:27	99.28%	0:15:38	99.75%	0:32:20
13	95.54%	0:09:30	97.79%	0:15:33	99.40%	0:16:45	99.80%	0:33:19
14	96.05%	0:09:59	98.04%	0:17:16	99.49%	0:19:22	99.83%	0:34:45
15	96.49%	0:10:33	98.27%	0:17:55	99.58%	0:19:49	99.87%	0:35:28
16	96.87%	0:11:10	98.46%	0:19:24	99.66%	0:20:48	99.91%	0:36:33
17	97.19%	0:11:41	98.64%	0:19:54	99.73%	0:21:31	99.95%	0:38:05
18	97.46%	0:13:10	98.77%	0:20:52	99.79%	0:22:25	99.96%	0:39:35
19	97.70%	0:13:42	98.90%	0:21:45	99.85%	0:24:24	99.98%	0:40:02
20	97.92%	0:14:20	99.01%	0:22:42	99.89%	0:28:15	100.00%	0:42:59

Tabela A.3. Maximal sem Soundex - Tempo x Revocação x Limiar

MAXIMAL SEM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo
2	-	-	-	-	-	-	64.39%	0:32:06
3	-	-	-	-	67.25%	0:07:44	69.35%	0:39:02
4	-	-	58.48%	0:03:45	76.46%	0:11:14	70.10%	0:42:00
5	48.97%	0:02:35	70.49%	0:06:08	79.14%	0:13:06	70.28%	0:43:34
6	62.44%	0:04:33	75.18%	0:07:58	80.30%	0:14:49	70.37%	0:44:53
7	68.79%	0:05:56	77.58%	0:09:00	81.02%	0:15:40	70.39%	0:45:40
8	72.41%	0:06:56	79.16%	0:10:09	81.48%	0:16:20	70.39%	0:46:18
9	74.79%	0:07:32	80.32%	0:10:39	81.83%	0:17:02	70.39%	0:46:48
10	76.50%	0:08:23	81.19%	0:10:55	82.07%	0:17:21	70.39%	0:47:04
11	77.90%	0:08:48	81.87%	0:11:19	82.26%	0:17:50	70.39%	0:47:29
12	79.10%	0:09:12	82.47%	0:11:59	82.39%	0:18:12	70.39%	0:47:50
13	80.05%	0:09:50	82.96%	0:12:33	82.50%	0:18:37	70.39%	0:48:09
14	80.89%	0:10:20	83.34%	0:12:53	82.60%	0:18:55	70.39%	0:48:27
15	81.60%	0:10:47	83.66%	0:13:16	82.66%	0:19:15	70.39%	0:48:46
16	82.16%	0:11:07	83.92%	0:13:36	82.70%	0:19:31	70.39%	0:49:02
17	82.64%	0:11:32	84.13%	0:13:56	82.73%	0:19:53	70.39%	0:49:22
18	83.03%	0:11:53	84.30%	0:14:17	82.76%	0:20:09	70.39%	0:49:38
19	83.37%	0:12:18	84.44%	0:14:42	82.78%	0:20:30	70.39%	0:50:03
20	83.66%	0:17:26	84.55%	0:19:51	82.79%	0:25:47	70.39%	0:55:17

Tabela A.4. Maximal com Soundex - Tempo x Revocação x Limiar

MAXIMAL COM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo	Revocação	Tempo
2	-	-	-	-	-	-	64.22%	0:13:34
3	-	-	-	-	76.86%	0:04:09	70.14%	0:16:12
4	-	-	74.39%	0:02:55	83.92%	0:06:14	71.29%	0:17:42
5	67.47%	0:02:25	83.98%	0:04:35	85.50%	0:07:52	71.58%	0:18:45
6	79.58%	0:03:42	87.01%	0:05:36	86.18%	0:08:25	71.70%	0:19:25
7	84.40%	0:04:46	88.51%	0:06:42	86.62%	0:09:13	71.77%	0:20:14
8	86.71%	0:05:51	89.33%	0:07:26	86.81%	0:10:08	71.78%	0:20:45
9	88.06%	0:06:43	89.86%	0:08:10	86.94%	0:10:35	71.80%	0:21:15
10	89.00%	0:07:18	90.23%	0:08:35	87.02%	0:11:04	71.80%	0:21:38
11	89.72%	0:08:23	90.51%	0:09:05	87.06%	0:11:31	71.80%	0:22:12
12	90.27%	0:08:43	90.73%	0:09:49	87.10%	0:11:56	71.80%	0:22:34
13	90.74%	0:09:24	90.92%	0:10:38	87.13%	0:12:18	71.80%	0:23:06
14	91.14%	0:10:10	91.06%	0:11:00	87.16%	0:12:42	71.80%	0:23:33
15	91.45%	0:10:31	91.19%	0:11:26	87.17%	0:13:06	71.80%	0:23:54
16	91.83%	0:11:03	91.30%	0:11:42	87.18%	0:13:29	71.80%	0:24:12
17	91.90%	0:11:34	91.38%	0:12:04	87.19%	0:13:50	71.80%	0:24:32
18	92.08%	0:12:05	91.44%	0:12:36	87.20%	0:14:08	71.80%	0:24:51
19	92.22%	0:12:33	91.50%	0:13:09	87.21%	0:14:27	71.80%	0:25:08
20	92.34%	0:19:17	91.53%	0:19:59	87.21%	0:21:07	71.80%	0:31:48

Apêndice B

Tabelas de AGM, Média de Grupo e Limiar

Tabela B.1. Fechados sem Soundex - AGM x Média do Grupo

FECHADOS SEM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo
2	-	-	-	-	-	-	1.017080667	2.307
3	-	-	-	-	1.214665807	2.31982	1.163793033	2.89992
4	-	-	1.26844652	3.48128	1.241022514	2.34076	1.204604962	3.14544
5	1.28	3.63645	1.263159094	3.51457	1.249399434	2.3434	1.220591846	3.26345
6	1.27	3.59267	1.26106851	3.53332	1.252259528	2.34292	1.227994665	3.32698
7	1.26	3.56789	1.258713069	3.53661	1.252693937	2.34065	1.231716195	3.36181
8	1.26	3.55913	1.256833374	3.53941	1.252638464	2.33907	1.234113029	3.38705
9	1.25	3.54549	1.254952479	3.5362	1.251870477	2.33743	1.2353783	3.40273
10	1.25	3.53379	1.253102737	3.52972	1.250943527	2.33573	1.236186739	3.41224
11	1.25	3.52343	1.251484783	3.52242	1.249911437	2.33407	1.236688519	3.41833
12	1.25	3.51877	1.250437956	3.51941	1.249157699	2.33277	1.237168171	3.42508
13	1.25	3.50799	1.249142878	3.51159	1.248146406	2.33132	1.237335643	3.42762
14	1.24	3.50239	1.248207997	3.50655	1.24733875	2.33026	1.237506945	3.43043
15	1.24	3.49752	1.247301632	3.50081	1.24657267	2.32928	1.237664937	3.43186
16	1.24	3.4929	1.246477387	3.49617	1.245945717	2.32848	1.237811254	3.43373
17	1.24	3.48885	1.245856233	3.49218	1.245430633	2.32774	1.237959484	3.43541
18	1.24	3.48423	1.245384856	3.48852	1.244899332	2.32709	1.23799255	3.43585
19	1.24	3.48063	1.244993775	3.48543	1.244468683	2.32646	1.238073458	3.43685
20	1.24	3.47726	1.244309429	3.48145	1.243996986	2.3259	1.238111183	3.43735

Tabela B.2. Fechados com Soundex - AGM x Média do Grupo

FECHADOS COM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo
2	-	-	-	-	-	-	1.012890275	2.26287
3	-	-	-	-	1.188957504	2.28159	1.168451303	2.90286
4	-	-	1.24599157	3.34683	1.222167002	2.31224	1.210660991	3.168
5	1.145123919	3.53547	1.245689495	3.40429	1.234049545	2.32171	1.225827438	3.29052
6	1.255277213	3.50646	1.245907045	3.43744	1.238719275	2.32452	1.232348132	3.35521
7	1.251100986	3.5019	1.246004133	3.45779	1.240974677	2.32567	1.235796703	3.39358
8	1.249010878	3.49988	1.245817512	3.46918	1.242125332	2.32607	1.237694179	3.41666
9	1.247417406	3.49527	1.245392334	3.47283	1.242459819	2.32588	1.238646258	3.42864
10	1.246431381	3.49272	1.244985163	3.47503	1.242542025	2.32594	1.239162808	3.43669
11	1.245394915	3.49002	1.244363311	3.4759	1.242469929	2.32583	1.239490601	3.44304
12	1.244823766	3.48935	1.244046019	3.4774	1.242434216	2.32576	1.239752948	3.44812
13	1.244144198	3.48541	1.243662043	3.47558	1.242280031	2.32552	1.239848976	3.44975
14	1.243723805	3.48253	1.243316289	3.47369	1.242074928	2.32534	1.239884134	3.4506
15	1.243206072	3.47942	1.242919526	3.47195	1.241894075	2.32523	1.23993133	3.45167
16	1.242958104	3.47837	1.242683069	3.47167	1.241807243	2.32517	1.240004556	3.45318
17	1.24273235	3.47748	1.242500311	3.47124	1.241714042	2.3251	1.24006151	3.45439
18	1.242502099	3.47585	1.242279126	3.47005	1.241539922	2.32498	1.240053698	3.45476
19	1.242144471	3.47378	1.242058413	3.46911	1.24141824	2.32496	1.240098607	3.45547
20	1.241918787	3.47236	1.241897709	3.46836	1.241354779	2.32489	1.240143515	3.45599

Tabela B.3. Maximal sem Soundex - AGM x Média do Grupo

MAXIMAIS SEM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo
2	-	-	-	-	-	-	1.017080667	2.13841
3	-	-	-	-	1.215041758	2.32019	1.038886843	2.16629
4	-	-	1.269450729	2.37017	1.215948948	2.32197	1.042258233	2.17053
5	1.118683133	2.3799	1.254261429	2.35264	1.213532406	2.31911	1.043011717	2.17134
6	1.26253632	2.35599	1.247631154	2.34477	1.211267606	2.31632	1.043327274	2.17172
7	1.253657485	2.34532	1.243651741	2.33943	1.209560471	2.31379	1.043361317	2.17174
8	1.249089881	2.34032	1.24082555	2.33627	1.208455896	2.3124	1.043359903	2.17174
9	1.246052832	2.33638	1.23887946	2.33374	1.207595554	2.31127	1.043358843	2.17174
10	1.243438272	2.33359	1.237114907	2.33145	1.206842602	2.31025	1.043357783	2.17173
11	1.241564473	2.33109	1.235695649	2.3296	1.206197855	2.3094	1.043356723	2.17173
12	1.240233361	2.32945	1.234633513	2.32831	1.20573983	2.30881	1.04335637	2.17173
13	1.238873059	2.32776	1.233647072	2.32702	1.205321532	2.30832	1.043356016	2.17173
14	1.238059369	2.32664	1.232938342	2.32601	1.205026102	2.30795	1.043356016	2.17173
15	1.237647959	2.32616	1.232516746	2.32545	1.204828605	2.3077	1.043356016	2.17173
16	1.237151341	2.32544	1.232060698	2.32492	1.204695211	2.30755	1.043356016	2.17173
17	1.236870611	2.32493	1.231785492	2.32451	1.204653933	2.30749	1.043356016	2.17173
18	1.236582926	2.32446	1.231486637	2.32406	1.204561204	2.30739	1.043356016	2.17173
19	1.236389219	2.32399	1.231423253	2.32394	1.204511746	2.30731	1.043356016	2.17173
20	1.236167251	2.32362	1.231227983	2.32361	1.204471992	2.30728	1.043356016	2.17173

Tabela B.4. Maximais com Soundex - AGM x Média do Grupo

MAXIMALCOM SOUNDEX								
	Suporte 5		Suporte 4		Suporte 3		Suporte 2	
Limiar	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo	AGM	Media Grupo
2	-	-	-	-	-	-	1.012890275	2.11975
3	-	-	-	-	1.188997776	2.28161	1.043846661	2.15557
4	-	-	1.246150422	2.33802	1.197612607	2.29355	1.049876872	2.16225
5	1.049410367	2.36152	1.237796615	2.32934	1.198582281	2.29522	1.051221409	2.16372
6	1.251487241	2.34161	1.234229051	2.32553	1.198450356	2.29525	1.05170935	2.16428
7	1.246083701	2.33539	1.232730227	2.32375	1.198468229	2.29529	1.051955755	2.16465
8	1.243098958	2.33172	1.231404381	2.32231	1.198167776	2.29499	1.052003431	2.16474
9	1.241005093	2.32945	1.230343655	2.32109	1.19787408	2.29463	1.052051105	2.16479
10	1.239736949	2.32784	1.2296789	2.32046	1.197681589	2.29447	1.052050688	2.16479
11	1.238564918	2.3265	1.229072695	2.31985	1.197531892	2.29426	1.052050271	2.1648
12	1.237877313	2.32585	1.228701395	2.31952	1.197420214	2.29412	1.052050271	2.1648
13	1.237185125	2.32512	1.228312219	2.319	1.197320047	2.29403	1.052050271	2.1648
14	1.236791908	2.32471	1.22807915	2.31874	1.197266128	2.29398	1.052050271	2.1648
15	1.236385815	2.3242	1.227857581	2.31845	1.197223608	2.29393	1.052050271	2.1648
16	1.236091571	2.324	1.227617267	2.31834	1.197192463	2.29389	1.052050271	2.1648
17	1.235885176	2.32377	1.227438631	2.31822	1.197171277	2.29386	1.052050271	2.1648
18	1.235739966	2.32363	1.227354834	2.31809	1.197140151	2.29382	1.052050271	2.1648
19	1.235509246	2.32332	1.227261557	2.31793	1.197123772	2.29379	1.052050271	2.1648
20	1.235326541	2.32314	1.227216519	2.31785	1.197113947	2.29378	1.052050271	2.1648