
UNIVERSIDADE FEDERAL DO AMAZONAS - UFAM
INSTITUTO DE COMPUTAÇÃO - ICOMP
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA - PPGI

# Extração Descentralizada de Conhecimento Associativo para Internet das Coisas

Márcio André da Costa Alencar

MANAUS-AM
2019

**PODER EXECUTIVO**
**MINISTÉRIO DA EDUCAÇÃO**
**UNIVERSIDADE FEDERAL DO AMAZONAS**
**INSTITUTO DE COMPUTAÇÃO**
**PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**


# Extração Descentralizada de Conhecimento Associativo para Internet das Coisas

Márcio André da Costa Alencar

Dissertação apresentada ao Programa de Pós-Graduação em Informática, da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Mestre em Informática, na área de concentração em Sistemas Embarcados e Engenharia de Softwares.

**Orientador:** Raimundo da Silva Barreto, D.Sc.

**MANAUS-AM**

**2019**

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Alencar, Marcio André da Costa
A368e Extração Descentralizada de Conhecimento Associativo para Internet das Coisas / Marcio André da Costa Alencar. 2019
79 f.: il. color; 31 cm.

Orientador: Raimundo da Silva Barreto
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. Internet das Coisas. 2. Análise Associativa. 3. Sistemas Distribuídos. 4. Mineração Descentralizada. I. Barreto, Raimundo da Silva II. Universidade Federal do Amazonas III. Título

# FOLHA DE APROVAÇÃO

## "Extração Descentralizada de Conhecimento Associativo para Internet das Coisas"

## MÁRCIO ANDRÉ DA COSTA ALENCAR

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Raimundo da Silva Barreto - PRESIDENTE

Prof. Eduardo James Pereira Souto - MEMBRO INTERNO

Prof. Ruiter Braga Caldas - MEMBRO EXTERNO

Prof. Luigi Carro - MEMBRO EXTERNO

Manaus, 25 de Março de 2019

*À Maria Sofia, razão de minhas conquistas.*

# Agradecimentos

Primeiramente gostaria e agradecer à Deus por ter me capacitado de todas as formas para iniciar, trilhar e concluir esta jornada. Agradeco minha esposa, Carol Falcão, e minha filha, Maria Sofia, que diariamente fazem da minha vida a melhor possível me trazendo muito amor e carinho, e à Aysah Luanny, minha filha de coração, que com toda sua estranheza, completa nossa família.

Destaco também a importantíssima participação do meu orientador, Prof. Dr. Raimundo Barreto, que me recebeu de braços abertos, prestando todo o auxílio necessário para o desenvolvimento de minha pesquisa. Além disto, realizou atividades que transcendem às competências de um orientador, com conselhos, conversar e abrindo portas que me pareciam impossíveis. Uma delas, que me levou até o Canadá, onde tive a oportunidade de conhecer o Prof. Dr. Richard Pazzi, ao qual também sou igualmente agradecido por me conceder seu tempo de descanso para que eu pudesse melhorar minha pesquisa.

Agradeço em especial a minha Mãe, Maria Esperança que, através de seu exemplo, me ensinou a ser perseverante e buscar seus objetivos, não importando o tempo e as dificuldades. Minha sogra, Marilene e meu sogro, Luiz Atlas, que nos concederam um cantinho em seu lar para que pudessemos recuperar nossas energias em família e finalmente, aos meus companheiros de laboratório, Anderson Cruz, Gabriel Leitão, Nilmara Salgado e Romário Lira que deixaram o ambiente acadêmico mais tranquilo e amigável com boas conversas e risadas.

Muito obrigado à todos, nada disso seria possível sem vocês!

*"...for once You have spoken: All nature and science follow the sound of Your voice..."*

*(So Will I - Hillsong United.)*

# Resumo

A identificação dos padrões de comportamento do usuário é um dos recursos que pode ser incorporado à Internet das Coisas. Encontrar padrões e utilizá-los como conhecimento para a tomada de decisões pode proporcionar facilidade, conforto, praticidade e autonomia para a execução das atividades diárias. Embora a extração de conhecimento seja comum em ambientes centralizados, sua execução em uma arquitetura descentralizada ainda é um desafio computacional relevante considerando as restrições de armazenamento e processamento dos dispositivos IoT. Esta dissertação descreve um método para minerar correlações implícitas entre os padrões de ações de dispositivos de IoT por meio de análise associativa embarcada. Com base nas métricas *support*, *confidence* e *lift*, o método identifica as correlações mais relevantes entre um par de ações de diferentes dispositivos e sugere ao usuário a integração entre elas por meio de solicitações HTTP. Resultados experimentais mostram que, em média, as regras mais relevantes para ambas as arquiteturas são as mesmas em 99,75% dos casos. Além disso, o método proposto identificou correlações relevantes que não foram identificadas pela arquitetura centralizada. Esta pesquisa enfatiza que a análise do padrão de ações do dispositivo é uma abordagem eficiente para fornecer um ambiente altamente integrado e inteligente, contornando os problemas do ponto único de falha e do armazenamento excessivo de dados em dispositivos IoT.

**Palavras-chave:** Internet das Coisas, Análise Associativa, Sistemas Distribuídos, Mineração Descentralizada.

# Abstract

Identifying user behavior patterns is one of the features that can be incorporated into the Internet of Things. Finding standards and using them as knowledge for decision making can provide ease, comfort, practicality and autonomy for the execution of daily activities. Although knowledge extraction is common in centralized environments, its execution in a decentralized architecture is still a relevant computational challenge considering the storage and processing constraints of IoT devices. This dissertation describes a method for mining implicit correlations between IoT device action patterns through embedded associative analysis. Based on the metrics support, confidence and lift, the method identifies the most relevant correlations between a pair of actions from different devices and suggests to the user the integration between them through HTTP requests. Experimental results show that, on average, the most relevant rules for both architectures are the same in 99.75% of cases. In addition, the proposed method identified relevant correlations that were not identified by the centralized architecture. This research emphasizes that device action pattern analysis is an efficient approach to provide a highly integrated and intelligent environment by circumventing single point failure problems and excessive data storage on IoT devices.

**Keywords:** Internet of Things, Associative Analysis, Distributed Systems, Decentralized Mining.

# Lista de Figuras

# Lista de Tabelas

# Lista de Abreviaturas e Siglas

| | |
|---|---|
| **API** | *Application Programming Interface* |
| **CoAP** | *Constrained Application Protocol – Protocolo de Aplicação Restrita* |
| **CVI** | *Cluster Validity Index* |
| **EGrC** | *Embedded Granular Computing* |
| **FARM** | *Freshness Association Rule Mining* |
| **GPIO** | *General Purpose Input/Output* |
| **HAC** | *Hierarchical Agglomerative Clustering* |
| **HTTP** | *Hypertext Transfer Protocol* |
| **ID3** | *Inductive Decision 3(Tree)* |
| **IoT** | *Internet of Things* |
| **JSON** | *JavaScript Object Notation* |
| **KNN** | *K-Nearest Neighbor* |
| **LFP** | *Local Frequent Patterns* |
| **LSHAP** | *Loop Scheduling with Heterogeneous Assignment with Probability* |
| **MAKE** | *Embedded Associative Knowledge Extraction* |
| **MPSoC** | *Multiprocessor System-on-Chip* |
| **MQTT** | *Message Queuing Telemetry Transport* |
| **OACCR** | *Obtaining Accurate and Comprehensible Classification Rules* |
| **PCA** | *Principal Components Analysis* |
| **RFID** | *Radio-Frequency Identification* |
| **ROM** | *Read Only Memory* |
| **SVM** | *Support Vector Machine* |

| | |
|---|---|
| **TITArl** | *Temporal Interval Tree Association Rule Learning* |
| **URL** | *Uniform Resource Locator* |
| **W3C** | *World Wide Web Consortium* |
| **WSU-CASAS** | *Washington State University: Center for Advanced Studies in Adaptative Systems* |
| **WoT** | *Web of Things* |
| **WS** | *Web Service* |
| **WT** | *Web Thing* |
| **WTM** | *Web Thing Model* |
| **WTC** | *Weighted Transitive Clustering* |
| **WTE** | *Web Thing Extended* |
| **WTS** | *Web Thing Semantic* |

# Lista de Símbolos

| | |
|---|---|
| $\mathbf{\Omega}$ | Espaço de resultados |
| $\times$ | Produto cartesiano entre dois conjuntos |
| $\mathbf{\lvert X \rvert}$ | Número de elementos do conjunto $X$ |
| $\Rightarrow$ | Implicação |
| $\leftarrow$ | Atribuição |
| $\cup$ | União de conjuntos |
| $\in$ | Pertence |
| $\cdot$ | Produto |
| $\forall$ | Para todos |
| $\mathbf{M_{ij}}$ | Matriz de $i$ linhas e $j$ colunas |
| $\mathbf{M_{ij}}$' | Matriz de $i$ linhas e $j$ colunas após transformação logaritmica |
| $\mathbf{c_{ij}}$ | Elemento da matriz |
| $\mathbf{P}$ | Padrão de ações |
| $\mathbf{D}$ | Base de transações |

# Sumário

# Capítulo 1

# Introdução

Embora não haja uma definição universal, o termo Internet das Coisas (em inglês *Internet of Things* - IoT) geralmente se refere ao cenário em que coisas (objetos, dispositivos, construções, plantas, animais, pessoas, etc.) são capazes de se conectar a uma rede de comunicação e realizar o sensoriamento, processamento, geração, consumo e troca de dados sem nenhuma intervenção humana. Suas aplicações atendem a diversas áreas, como por exemplo, residências, agricultura, transportes, segurança, educação, cuidados com a saúde, indústrias e outros cenários, com intuito de melhorar a qualidade de vida das pessoas (McArthur *et al.* 2012; Gonzalez e Amft 2015; Verhelst e Moons 2017).

O contínuo desenvolvimento de tecnologias, componentes, sistemas e infraestrutura ampliam a capacidade de conectividade, armazenamento e processamento em dispositivos inteligentes (Yazici *et al.* 2018). Tais capacidades, aliadas às técnicas de reconhecimento de padrões, proporcionam mais robustez, autonomia a esses dispositivos. O crescimento comercial de produtos com essas características impulsionam a expansão no número de ambientes nos quais estamos inseridos.

Atualmente, estima-se que haja 1 bilhão de dispositivos conectados diariamente em todo o mundo. Projeções indicam que até 2021 as empresas irão investir aproximadamente US$1,4 trilhões e, com isso, espera-se que até 2020 haja mais de 50 bilhões de dispositivos conectados à internet (Rose *et al.* 2015; Perez *et al.* 2018).

Tais estatísticas atestam que a Internet das Coisas já está presente nas vidas das pessoas, de forma direta ou indireta, coletando/gerando dados com padrões, tendências e

correlações implícitas que podem auxiliar na tomada de decisões em diversos cenários de aplicações.

As atividades diárias de uma pessoa, e a forma como ela interage com as coisas ao seu redor, geram um volume de dados nos quais estão inseridas informações relevantes quanto aos seus padrões de uso e preferências. A identificação e extração destas informações implícitas vai ao encontro do potencial esperado para a Internet das Coisas, possibilitando integração entre dispositivos, automatização de atividades e predição de ações ressaltando assim a importância dos estudos nesta área.

## 1.1 Contexto

O tipo de inteligência que pode ser agregado à Internet das Coisas pode variar de acordo com o seu propósito, como por exemplo, a capacidade de identificar padrões de uso e, até mesmo, correlações implícitas nos registros de atividades entre os dispositivos que compõem o ambiente. A arquitetura sobre a qual é implementado este ambiente, implica diretamente na técnica utilizada para extração de conhecimento.

Em uma arquitetura centralizada, as informações coletadas são armazenadas unicamente no dispositivo central, simplificando o processo de coleta e análise de dados. Porém, há a necessidade de que este dispositivo tenha recursos suficientes para armazenar e processar todos os dados gerados pelos sensores do ambiente. Além disso, a relação de dependência para com este nó central se destaca como um ponto de vulnerabilidade e que inviabilizaria o funcionamento adequado do ambiente em caso de falha.

Por outro lado, uma arquitetura descentralizada não demanda a presença de um elemento único de armazenamento e processamento de dados. Os problemas de dependência e exigência de muitos recursos seriam mitigados pela capacidade dos dispositivos processarem seus próprios volumes de dados em ambiente embarcado. Em contrapartida, cada dispositivo deverá ser responsável em oferecer ao usuário todos os mecanismos necessários para configuração, conectividade, controle, armazenamento e processamento de dados. Neste contexto, a falha de um dos dispositivos não afetaria diretamente aos demais, destacando-se como um cenário mais interessante, além de suportar melhor o acréscimo de novos dispositivos sendo,

portanto, uma solução escalável.

O estado da arte em reconhecimento de padrões são os algoritmos de aprendizagem de máquinas (e.g.: Redes Neurais Profundas) que têm o custo de armazenamento e processamento elevados, apresentando-se como desafio computacional significativo em todo o espectro de dispositivos de computação, desde clientes com poucos recursos até servidores em nuvem (Chen *et al.* 2015; Rose *et al.* 2015; Verhelst e Moons 2017). Uma estratégia para contornar tais adversidades é a utilização de técnicas de mineração de dados, mas especificamente das regras de associação, que buscam identificar padrões similares em um conjunto dados de modo que satisfaça critérios de confiabilidade mínimos (Tan *et al.* 2006; Chen *et al.* 2015; Li *et al.* 2018; Nazerfard 2018; Kireev *et al.* 2019).

Embora sejam aplicadas frequentemente em bases de dados volumosas, a simplicidade dos algoritmos de análise associativa possibilitam que suas implementações não exijam um grande volume de dados para serem armazenadas, sendo uma abordagem relevante no contexto de dispositivo IoT. Outra observação é que, geralmente, tais algoritmos visam identificar o maior e mais frequente conjunto de itens em uma base de dados. Para isso, existem vários algoritmos de análise associativa, tais como o Apriori (Agrawal e Srikant 1994), *FP-Growth* (Han *et al.* 2004), Mineração Baseada em Restrições (Boulicaut e Jeudy 2005), entre outros. Diferentemente destes, o método proposto visa identificar as correlações mais fortes entre dois itens, ou seja, identificar conjuntos com 2 itens (2-*itemsets*) que estão fortemente correlacionados.

Visando explorar as vantagens de uma arquitetura descentralizada combinadas à extração de conhecimento através da análise associativa, os estudos conduzidos nesta dissertação buscam desenvolver um mecanismo descentralizado para identificar correlações implícitas entre dispositivos de um ambiente e oferecer ao usuário sugestões de correlações entre as ações/estados dos dispositivos de tal maneira que seja possível sincronizar seus estados satisfazendo aos padrões de iteração do usuário com ambos dispositivos (p. ex. ao fechar a janela do quarto, ligar a luz do quarto).

## 1.2 Definição do Problema

Assumindo como premissa que o ambiente é composto por vários dispositivos de baixo custo, consequentemente com poucos recursos, é necessário descentralizar o processo de extração de conhecimento associativo capacitando tais dispositivos a executar, em ambiente embarcado, a extração deste conhecimento. Vale destacar ainda que, embora o cenário possibilite a utilização de equipamentos com muitos recursos, o uso destes seria um desperdício desnecessário uma vez que não haverá necessidade de aglomeração de um grande volume de dados além dos recursos permanecerem ociosos a maior parte do tempo, já que a extração de conhecimento é executada de tempos e tempos.

O problema tratado nesta dissertação pode ser expresso através da seguinte pergunta: é possível prover aos dispositivos IoT a capacidade de extrair conhecimento associativo, por meio da análise de dados embarcada, para identificar as correlações mais fortes, entre pares de dispositivos, sem a necessidade de agregar todos os dados simultaneamente em um único nó concentrador?

Tendo como premissa o cenário exposto, os desafios desta abordagem consistem em:

- Respeitar a capacidade de armazenamento e processamento dos dispositivos;
- Identificação de dispositivos na rede;
- Tratamento de dados em ambiente embarcado;
- Restrições temporais das correlações identificadas;
- Definição de um protocolos de comunicação apropriado para a abordagem.

Cada desafio citado implica em uma série de problemas adjacentes tais como: tipo de estrutura de dados, tecnologia e protocolos usados, sintaxe de comunicação, registros de mudança de estado dos dispositivos, tratamento de requisições, confiabilidade do conhecimento e alta sensibilidade a novos registros.

## 1.3 Motivação

O interesse da indústria no desenvolvimento de Internet das Coisas impulsiona a fabricação de produtos e tecnologias voltadas para a melhoria da qualidade de vida das pessoas. Através destas tecnologias é possível levar segurança, praticidade e comodidade nas atividades do dia a dia dos usuários. Destaca-se ainda que, para grupos de pessoas que necessitem de cuidados especiais, como idosos e pessoas com restrições física, a Internet das Coisas se apresenta como ferramenta assistiva, dando suporte à autonomia e independência dessas pessoas (Rose *et al.* 2015; Buyya e Dastjerdi 2016; Perez *et al.* 2018; Kireev *et al.* 2019).

Outro fator motivador é a exploração da fronteira do conhecimento em relação a Internet das Coisas. Embora haja muitos estudos voltados para o avanço de ambiente utilizando uma arquitetura centralizada, explorar técnicas de mineração de dados embarcadas em arquiteturas descentralizadas é uma abordagem pouco usual no contexto atual em que as redes neurais profundas tem se apresentado como o estado da arte em relação ao reconhecimento de padrões. Porém, com o custo muito elevado (processamento e armazenamento), essa abordagem limita seus cenários de aplicações sendo necessário explorar as áreas não cobertas por esta abordagem.

## 1.4 Objetivos

O objetivo principal desta dissertação é demonstrar a capacidade de dispositivos IoT em identificar correlações implícitas entre seus padrões de ação por meio da mineração de dados descentralizada.

Para que tal objetivo seja alcançado, faz-se necessário o cumprimento dos seguintes objetivos específicos:

- Possibilitar que os dispositivos sejam capazes identificar seus próprios padrões de uso por meio da análise de dados em ambiente embarcado;
- Demonstrar a eficiência da análise de associativa descentralizada quanto a capacidade de identificação de correlações implícitas entre os padrões de uso dos dispositivos; e

- Propor e validar um modelo para armazenamento de dados embarcado que possibilite a redução do consumo de espaço para armazenamento sem perda da capacidade de extração de conhecimento;

## 1.5 Organização da dissertação

No **Capítulo 1** é apresentada introdução deste trabalho através da contextualização, definição do problema, as motivações e os objetivos desta dissertação.

O **Capítulo 2** enfatiza a fundamentação teórica, explorando assuntos acerca de sistemas embarcados, arquiteturas, modelos, técnicas e algoritmos utilizados durante o desenvolvimento desta dissertação, de tal forma que sejam explanados conceitos e definições básicas para compreensão da metodologia apresentada.

Os **Capítulos 3 e 4** descrevem, respectivamente, o protocolo da revisão sistemática e a discussão dos trabalhos correlatos referentes ao objeto de pesquisa nesta dissertação. Ambos capítulos possibilitam uma visão geral das técnicas e algoritmos usado para solucionar problemas similares aos apresentados durante a definição do problema.

O método proposto é detalhado no **Capítulo 5** de tal forma que é possível observar, inicialmente, uma visão geral do mecanismo e, em um segundo momento, as particularidades do funcionamento individual e colaborativo dos dispositivos.

O **Capítulo 6** descreve a metodologia de avaliação, os parâmetros definidos e os *datasets* utilizados para validar o método proposto. Além destes, apresentam os resultados obtidos pela aplicação das técnicas definidas no Capítulo 5.

No **Capítulo 7** são discutidos os resultados obtidos durante os experimentos bem como são levantadas as considerações finais abordando a cerca de suas possíveis cenários de aplicações e as suas limitações do método proposto. Também são levantados os pontos de pesquisas futuros e as principais contribuições desta pesquisa.

# Capítulo 2

# Fundamentação Teórica

Para melhor compreensão dos elementos que compõem este estudo, faz-se necessário o entendimento prévio de alguns mecanismos, técnicas e tecnologias, como estas se adequam para o desenvolvimento dos experimentos e como corroboram para alcançar os objetivos desta pesquisa. Neste capítulo é introduzido conceitos sobre Análise Probabilistica (Seção 2.1) apresentando conceitos sobre espaço de resultados e eventos aleatórios, Sistemas Embarcados (Seção 2.2), apontando suas características, particularidades e desafios de planejamento, desenvolvimento e implementação. Posteriormente, apresenta-se uma consequência do contínuo avanço dos Sistemas Embarcados, a Web das Coisas (Seção 2.4), que é uma sub-área da Internet das Coisas onde são descritos conceitos, arquiteturas e soluções deste paradigma. Também é incorporado a este capítulo conceitos sobre Mineração de Dados (Seção 2.5) e suas aplicações no contexto de Internet das Coisas e Sistemas Embarcados, focando-se principalmente em análise associativa.

## 2.1 Análise Probabilística

A teoria matemática da probabilidade nos dá as ferramentas básicas para a construção e análise de modelos matemáticos para fenômenos aleatórios. Ao estudar um fenômeno aleatório, estamos lidando com um experimento cujo resultado não é previsível antecipadamente. Experiências deste tipo que, imediatamente vêm à mente, são as que surgem nos jogos de azar. De fato, o primeiro desenvolvimento da teoria da probabilidade nos séculos

XV e XVI foi motivado por problemas desse tipo (Soong 2004).

DeGroot e Schervish (2012) define que **um experimento** é qualquer processo, real ou hipotético, no qual os resultados possíveis podem ser identificados com antecedência e **um evento** é um conjunto bem definido de possíveis resultados do experimento. Neste contexto, a importância do estudo de eventos aleatórios está nas inferências que são possíveis realizar sobre os eventos que podem ocorrer, como por exemplo:

$$E_1 : \textit{O lançamento de uma moeda e a observação da face voltada para cima}$$

O espaço de resultados ($\Omega$) é **discreto** se há um número, finito ou infinito, numerável de elementos. Se $\Omega$ contém um intervalo, finito ou infinito, de números reais, então o espaço de resultados é **contínuo**. No contexto do $E_1$ temos um espaço de resultados discreto.

$$\Omega_{E_1} = \{cara, coroa\}$$

Este espaço possui dois possíveis eventos: *cara* e *coroa*. Cada elemento de $\Omega$ é identificado como ponto amostral ou acontecimento. Em muitos experimentos é necessário identificar qual a probabilidade de ocorrência de um dado acontecimento. Para isso, a Regra de Laplace define que, em um experimento que possua $N$ possíveis resultados ($|\Omega|$) mutuamente excludentes, é possível calcular a probabilidade de um acontecimento $A$ de tal forma que:

$$P(A) = \frac{N_A}{N} = \frac{\textit{número de elementos em } \Omega \textit{ iguais à } A}{\textit{número de elementos em } \Omega} \tag{2.1}$$

A combinação de dois eventos gera também uma combinação de seus espaços de resultados. Considere, por exemplo, um experimento E3 que consiste no lançamento de duas moedas e a observação de suas faces voltadas para cima, ou seja, $E_1$ é o lançamento e observação da primeira moeda e $E_2$ é lançamento e observação da segunda moeda. O espaço de resultados deste experimento é gerado pelo produto cartesiano entre ambos espaços de

resultados para experimento do lançamento de cada moeda ($E_1$ e $E_2$):

$$\textbf{Seja:} \quad E_1 = E_2; \quad e \quad \Omega_{E_1} = \Omega_{E_2} = \{cara, coroa\}$$

$$\textbf{Então:} \quad \Omega_{E_3} = \Omega_{E_1} \times \Omega_{E_2} = \{cara, coroa\} \times \{cara, coroa\}$$

$$\textbf{Logo:} \quad \Omega_{E_3} = \{(cara, cara), (cara, coroa), (coroa, cara), (coroa, coroa)\}$$

Considerando a combinação de espaços de resultados para o experimento, no qual $|\Omega_{E_3}| = 4$. A **probabilidade** de **ambas moedas** caírem com a face **cara** voltada **para cima** é dada por:

$$\textbf{E}_\textbf{3} : P_{E_3}(cara, cara) = 1/4 = 0,25 = 25\%$$

$$\textbf{Ou} : P_{E_3}(cara, cara) = P_{E_1}(cara) \cdot P_{E_2}(cara) = 0,5 \cdot 0,5 = 0,25$$

A probabilidade é de 25% pois há apenas um único acontecimento em $\Omega$ igual ao resultado dentre os 4 possíveis. Outra forma de identificar esta probabilidade é através do produto das probabilidades dos eventos individuais ($P_{E_1}(cara) \cdot P_{E_2}(cara)$), ou seja, a probabilidade do primeiro lançamento cair cara e a probabilidade do segundo lançamento também cair cara. Esta análise possibilita avaliar em um conjunto de espaço amostral a probabilidade de um dado acontecimento ocorrer. Este espaço amostral pode se ampliar ou reduzir dada as características do experimento, o número de repetições e as combinações dos espaços de resultados individuais.

## 2.2 Sistemas Embarcados

Embora haja diversas definições, para Health (2003), sistemas embarcados é um sistema baseado em microprocessador desenvolvido para controlar uma funcionalidade ou um conjunto de funções específicas não programadas pelo usuário final. Consoante a tal definição, Barr

e Massa (2006) definem como uma combinação de hardware e software - e talvez periféricos adicionais - projetados para executar uma função dedicada. No contexto da Internet das Coisas, um sistema embarcado pode ser considerado adequado para o desenvolvimento de aplicativos em IoT quando o dispositivo utilizado for integrado ou compatível com alguma interface de comunicação de dados Alvarado Moreno *et al.* (2018).

Os sistemas embarcados para IoT estão presentes em muitos objetos de nosso dia a dia tais como, micro-ondas, televisão, condicionador de ar, geladeiras dentre outros. Por se tratarem de sistemas com funcionalidades específicas é possível encontrar algumas restrições, como as citadas por Berger e Berger (2002) e Alvarado Moreno *et al.* (2018):

- São sensíveis aos custos: dimensão, quantidade e qualidade de componentes, conectores e periféricos usados;
- Possuem restrições quanto ao consumo de energia: devem trabalhar confiavelmente por longos períodos com fonte limitada de energia;
- Operam em ambientes com condições extremas: estão em todas as partes, logo, estão sujeitos às variações ambientais (calor, frio, vibrações, umidade, etc.) e
- Requerem interfaces apropriadas para a interconexão de sensores.

Além das características citadas acima, como trata-se de sistemas embarcados com funcionalidade bastante específica, é possível destacar também as restrições de processamento, onde deve-se estimar muito bem qual microcontrolador melhor se ajusta ao projeto, uma vez que o uso indiscriminado de microcontroladores com alto poder de processamento, além de ser financeiramente desfavorável, gera desperdício de recursos.

## 2.3 Internet das Coisas

O contínuo desenvolvimento de sistemas embarcados, a facilidade de acesso a componentes e a crescente acessibilidade de serviços de rede impulsionaram o crescimento no número de dispositivos conectados à internet. Este crescimento deu suporte a este novo paradigma conhecido como Internet das Coisas, o qual, abre as portas para inovações gerando novas

formas de interação entre seres vivos e máquinas, possibilitando a construção de cidades inteligentes, infraestruturas e serviços para melhorar a qualidade de vida e a utilização dos recursos.

Para Buyya e Dastjerdi (2016), a concretização da interoperabilidade entre esses diversos sistemas embarcados e dispositivos depende das empresas entrarem em acordo quanto à pilha de protocolo usada na comunicação, porém isto envolve uma série de aspectos, tecnologias e padrões. Embora tal processo seja complexo por envolver muitas variáveis, como quantidade de empresas, economia global, interesses empresariais e custos de fabricação, vários consórcios, órgãos e grupos de pesquisas apresentam modelos que possibilitam tal interoperabilidade.

Um destes modelos é a *Web Thing Model*, apresentado pela W3C (*World Wide Web Consortium*), que busca descrever um modelo e uma API (*Application Programming Interface*) Web a serem seguidas por quaisquer fabricantes que desejem desenvolver um produto, dispositivo, serviço ou aplicação para Web das Coisas (Trifa *et al.* 2015).

## 2.4 *Web* das coisas

Distanciando-se das demais propostas que buscam gerar novos protocolos e padrões para interoperabilidade, a Web das Coisas (do inglês Web of Things - WoT) faz uso de uma estrutura amplamente difundida, reduzindo sua complexidade e expandindo sua compatibilidade por meio dos padrões *web*. Cada objeto na WoT é identificado como *Thing* ou *Web Thing* (WT), que é uma representação digital de um objeto físico acessível por meio de uma API RESTful, embarcada ou não. RESTful é um estilo arquitetônico, ou padrão de programação, que possibilita, através de uma representação de uma interface simples e o protocolo HTTP e/ou *Web Service*(WS), a interoperabilidade entre sistemas Trifa *et al.* (2015).

### 2.4.1 Requisitos para a *Web Thing*

Para que um *web server* seja considerado uma *Web Thing* é necessário que o mesmo atenda às especificações, convenções de nomes e recomendações mínimas definidas pelo modelo, tais como:

- Suportar ao menos HTTP/1.1 e, quando possível, suportar HTTP/2;

- Disponibilizar seus recursos acessíveis via URL HTTP exclusiva, sendo capaz de responder uma requisição HTTP GET para seu endereço raiz (IP ou Nome) sobre a porta padrão (80 para HTTP, 443 para HTTPS);

- Suportar requisições HTTP do tipo GET, POST, PUT e DELETE para operações de leitura, criação, atualização e remoção respectivamente;

- Implementar os códigos de status HTTP 200 (*Success*), HTTP 400 (*Bad Request*) e HTTP 500 (*Internal Server Error*);

- Suportar JSON;

- Suportar HTTP GET em sua URL raiz.

Embora este sejam os requisitos mínimos, as especificações do modelo abrange mais funcionalidades para uma *Web Thing* que permitem amplicar sua interoperabilidade, robustez e segurança.

### 2.4.2 Padrões de Integração

As *Web Things* podem se integrar através de três padrões de conectividades: *Direct Connectivity, Gateway Based Connectivity* e *Cloud Based Connectivity.* Tais padão definem como os objetos físicos se integrarão com a *web.*

O padrão *Direct Connectivity* define que cada *Web Thing* possui sua API para a qual os clientes devem enviar suas solicitações. O cliente e a WT podem estar na mesma rede ou em redes diferentes, modificando-se apenas a URL para a qual o cliente deve enviar sua requisição. A Figura 2.1 apresenta este modelo de conectividade.

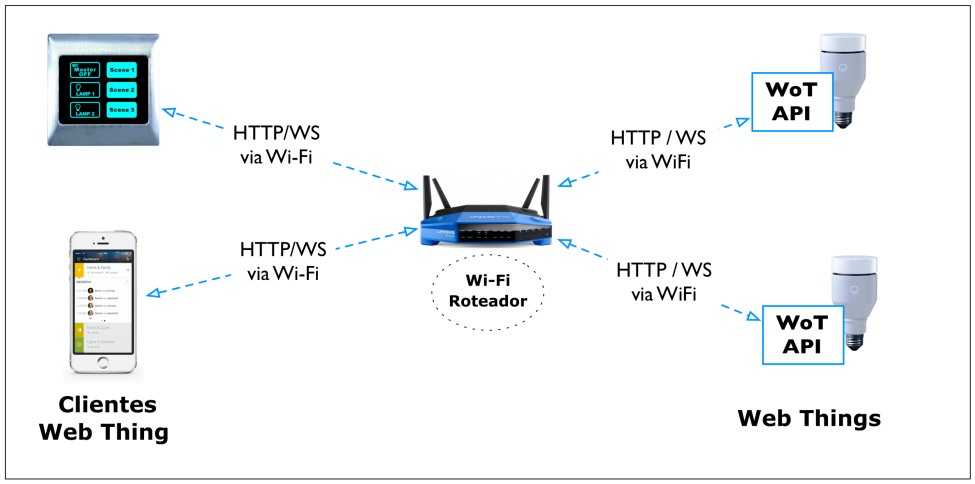


Figura 2.1: Padrão *Direct Connectivity*.
Adaptado de: https://model.webofthings.io/

O padrão *Gateway Based Connectivity* (Figura 2.2) geralmente é usado quando o dispositivo (coisa) não dispõe dos recursos necessários para prover uma API embarcada. Dessa forma uma WT intermediária expõe a API atuando como um *proxy* ou *gateway* (dependendo da complexidade do sistema), intermediando a comunicação entre a coisa e outros sistemas.

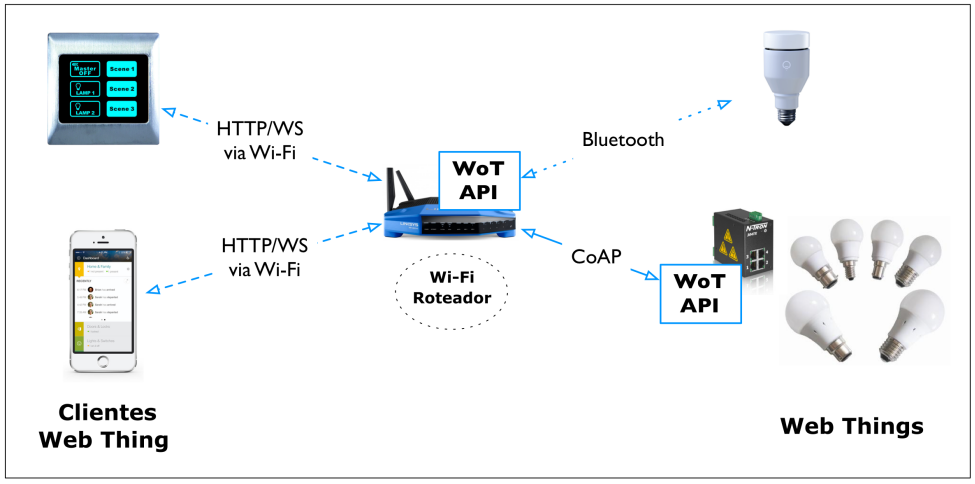


Figura 2.2: Padrão *Gateway Connectivity*.
Adaptado de: https://model.webofthings.io/

O padrão *Cloud Based Connectivity* (Figura 2.3) é similar ao *Gateway Based Connectivity*, porém, neste caso, o *gateway* é um serviço em nuvem e sua comunicação exige

conectividade com a internet.

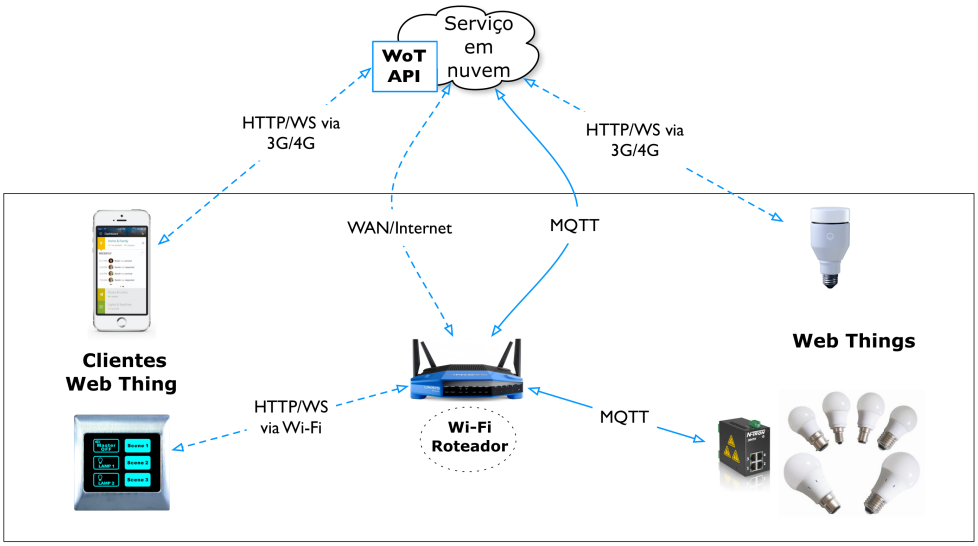


Figura 2.3: Padrão *Cloud Connectivity*.
Adaptado de: https://model.webofthings.io/

Embora sua topologia de rede seja centralizada (*Wi-Fi/Roteador*), os serviços oferecidos por cada *Web Thing* são independentes uns dos outros, possibilitando aos clientes acessarem suas funcionalidades diretamente através da rede, local ou não. Outra característica importante é que a interrupção de um dos dispositivos não afeta aos demais.

Os três modelos de conectividade da Web das Coisas possibilitam aos desenvolvedores atender diversos cenários de aplicações as quais são essenciais para implementação de um ambiente composto de diversos agentes.

### 2.4.3 Modelo *Web Thing*

Seguidas as convenções dad Seções 2.4.1 e 2.4.2, é possível ler e trocar dados com qualquer outra entidade na WoT. Porém, ainda não é possível a compreensão desses dados. Dessa forma, a *Web Thing Model* (WTM) especifica um modelo, isto é, o conteúdo JSON e API REST que uma WT deve implementar. Atender às especificações do WTM transforma a *Web Thing* em *Web Thing Estendida* (WTE) e a implementação das exigências que definem a semântica a transforma em uma *Web Thing* Semântica (WTS).

Cada WTE deve apresentar uma URL específica para cada um dos seus recursos, conforme apresentado na Tabela 2.1. É encorajado o uso de uma estrutura lógica de árvores que permitam a navegação pelos recursos mantendo pequeno o volume de dados no pacote JSON. O conteúdo JSON pode conter campos adicionais, dependendo do tipo de recursos que está sendo considerado.

Tabela 2.1: URL's exigidas para uma *Web Thing* Estendida

| **URL** | **RECURSOS** |
|---|---|
| {wt} | Retorna um objeto que é sua representação |
| {wt}/model | Retorna um objeto contendo o modelo da *Web Thing* |
| {wt}/properties | Retorna um vetor de propriedade que o recurso inicial possui |
| {wt}/properties/{id} | Retorna uma lista de valores recentes da propriedade |
| {wt}/actions | Retorna um vetor de descrições ações que o recurso pode realizar |
| {wt}/actions/{id} | Retorna um vetor que lista as execuções recentes de uma ação específica |
| {wt}/actions/{id}/{actionId} | Retorna o status de uma ação ou 404 caso a id da ação não for encontrada |

Atendendo aos requisitos apresentados por este modelo é possível acessar informações, propriedade, ações e outros recursos da *Web Thing* além de prover uma interface simples e eficiente para comunicação entre os dispositivos IoT. Não se limitando à isto, é possível se comunicar com quaisquer sistema capaz enviar e receber requisições HTTP.

## 2.5 Mineração de Dados

Mineração de dados, também conhecida como descoberta de conhecimento a partir de dados, é o processo, automatizado ou por conveniência, de extração de padrões que representam conhecimento implicitamente armazenados ou capturados em base de dados (Han *et al.* 2012).

Esse processo, como apresentado por Chen *et al.* (2015), é dividido nas seguintes etapas:

I. **Preparação de dados:** que consiste em realizar a limpeza de ruídos, seleciona um conjunto de dados dentro de uma base maior ou integrar esses dados com outros dados;

II. **Mineração de dados:** neste processo são executados os algoritmos que buscam encontrar e validar padrões de conhecimentos descobertos; e

III. **Apresentação de dados:** visualizar os dados e representar o conhecimento extraído para o usuário.

As funcionalidades da mineração de dados incluem classificação, agrupamento, análise associativa, análise de séries temporais e análises de *outliers*. Destas funcionalidades podemos destacar a análise associativa, utilizada neste trabalho.

### 2.5.1 Análise Associativa

Segundo Chen *et al.* (2015), a análise associativa é a descoberta de regras de associação que exibem condições de atributo-valor que frequentemente ocorrem juntas em um determinado conjunto de dados. Em outras palavras, as regras de associação correlacionam itens ou conjuntos de itens (*itemset*) de uma determinada base de dados baseado no quão frequente esses itens aparecem juntos nesta base. Tais condições podem ser representadas por meio de regras expressas no formado $A \Rightarrow B$ $[métricas]$ onde A define a premissa da regra (antecedente), B é a conclusão da regra (consequente) e "$[métricas]$" são os valores que permitem quantificar a inferência. Estas expressam o quão frequente os *itemsets* A e B são em uma base de dados

Geralmente a regra de associação é considerada interessante se ela satisfaz um limiar mínimo de suporte (*support*). Esta métrica, baseada nos princípios da análise probabilísticas (Seção 2.1), indica a frequência de um item em uma base de transações $D$ e pode ser expressa com a seguinte equação.

$$support(A \Rightarrow B) = P(A \cup B) = \frac{freq(AB)}{|D|} = \frac{frequência\ de\ A\ e\ B\ juntos}{num.\ de\ transações\ em\ D} \tag{2.2}$$

Outra métrica é a confiança (*confidence*), esta por sua vez representa a probabilidade condicional de B dado A, ou seja, o quão frequente é o item B, dado todas as transações que possuem o item A. Esta regra é representada pela seguinte equação:

$$confidence(A \Rightarrow B) = \frac{P(A \cup B)}{P(A)} = \frac{freq(AB)}{freq(A)} = \frac{frequência\ de\ A\ e\ B\ juntos}{frequência\ de\ A} \tag{2.3}$$

Uma terceira métrica chamada elevação (*lift*) define o grau de correlação entre o antecedente e o consequente. Tal métrica é calculada por meio do suporte e confiança como segue:

$$lift(A \Rightarrow B) = \frac{P(A \cup B)}{P(A)P(B)} = \frac{confidence(A \Rightarrow B)}{support(B)} = \frac{freq(AB) \cdot |D|}{freq(A) \cdot freq(B)} \tag{2.4}$$

Esta métrica sugere que se $P(A \cup B) = P(A)$, a ocorrência do itemset A e independente da ocorrência do itemset $B$; do contrário, $A$ e $B$ são dependentes e correlacionadas.

Diferente das métricas de *support* e *confidence* que têm suas faixa de valores de 0 a 1 (ou seja 0% a 100%), a métrica *lift* define que valores menores que 1 indicam que os itens $A$ e $B$ são inversamente correlacionados ($A \wedge \neg B$); se o valor for igual à 1, então $A$ e $B$ são independentes e não estão correlacionados; ou se o *lift* for maior de 1, $A$ e $B$ estão diretamente correlacionados ($A \wedge B$).

Primariamente, os algoritmos de análise associativa visam identificar os maiores e mais frequentes conjuntos de itens (*itensets*) em uma base de transações e, para isso, seguem as especificações apresentadas nesta seção. Alguns exemplos destes algoritmos são o *Apriori* (Agrawal e Srikant 1994), *FP-Growth* (Han 2005) e a Mineração Baseada em Restrições (Boulicaut e Jeudy 2005). Diferente destas abordagens, o método proposto visa identificar as correlações mais fortes entre dois itens, em outras palavras, uma regra que correlaciona fortemente o antecedente e o consequente combinando apenas dois itens (2-itensets).

### 2.5.2 Regra de Associação Apriori

Proposto por Agrawal e Srikant (1994), o algoritmo de Regras de Associação Apriori busca identificar *itemsets* frequentes para regras de associação booleanas. Este algoritmo adota a propriedade Apriori

na qual considera que os *k-itemsets* (*itemset* de tamanho k) são usados para explorar *(k+1)-itemsets*.

Formalmente, esta propriedade é definida pela seguinte observação: *Se um itemset $I$ não satisfaz um limite mínimo de suporte então $I$ não é frequente, ou seja, $P(I) < min\_sup$. Se um itemset $A$ é adicionado ao itemset $I$ então o conjunto resultante $I \cup A$ não pode ocorrer com uma frequência maior que $I$, logo $I \cup A$ também não é um itemset frequente, ou seja, $P(I \cup A) < min\_sup$.* Esta propriedade pertence à classe de propriedades conhecidas como anti-monotonicidade: a qual define que se um conjunto $(I)$ não passar em um dado teste, então um superconjunto $(I \cup A)$ também não passará. Em síntese, um itemset é frequente "se e somente se" o subconjunto deste itemset também for frequente. Dessa forma o algoritmo de Regra de Associação Apriori executa duas operações básicas:

I. ***Join Step*:** para encontrar $L_k$, um conjunto de candidatos de $k - itens$ é gerado unindo-se $L_{k-1}$ com si mesmo.

II. ***Prune Step*:** em cada iteração todos os itens do conjunto devem atender ao limiar mínimo de suporte, caso contrário não pertencerá ao superconjunto.

Embora simples, o algoritmo de regra de associação possui um custo computacional elevado, no sentido de que o primeiro passo do algoritmo, *Join Step*, realiza a combinação de todos os itens do conjunto para formação dos superconjuntos em cada iteração. Dessa forma o Passo 2, *Prune Step*, ameniza tal complexidade reduzindo o número de candidatos para a próxima iteração excluindo itensets não frequentes. Tal processo é ilustrado na Figura 2.4.

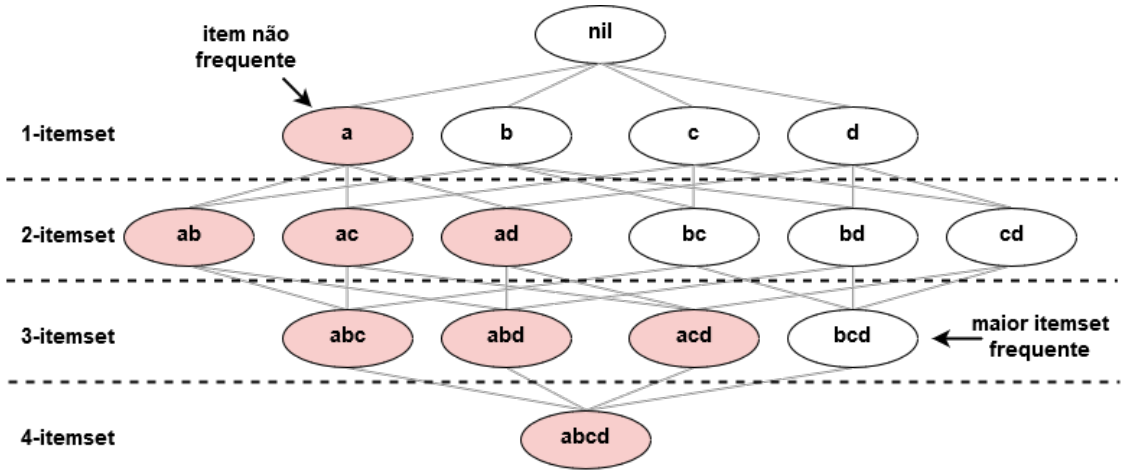


Figura 2.4: Ilustração da execução do algoritmo de regra de associação.

No exemplo, é possível identificar o processo de *Join Step*, em que cada linha representa

uma iteração do algoritmo iniciando de um conjunto vazio até a geração do maior itemset mais frequente (3-itemset). É possível observar também o impacto do processo de *Prune Step* em que todos os conjuntos que possuem o item não frequente (a) são desconsiderados, reduzindo o custo de processar mais da metade de conjuntos possíveis para as próximas iterações. O processo se repete à cada iteração até que o maior itemset frequente (bcd) seja encontrado.

## 2.6 Resumo

No Capítulo 2 foram descritos todos os conceitos, definições e teorias que dão suporte ao desenvolvimento da solução apresentada nesta dissertação. Foram apresentados os fundamentos de análise probabilística de eventos, as principais características de sistemas embarcados e suas restrições quanto ao armazenamento, processamento e consumo de energia. Também foram introduzidos conceitos sobre Web das Coisas, o qual possibilita a integração e interoperabilidade entre dispositivos na Internet das Coisas e, além destes, explanou-se sobre as técnicas de mineração de dados por meio da análise associativa que permite correlacionar conjuntos de itens frequentes em uma base de transações.

# Capítulo 3

# Revisão Sistemática

Como base para definição do algoritmo de regra de associação utilizado no método proposto (Capítulo 5), realizou-se uma revisão sistemática cujo objeto da pesquisa era avaliar estudos que aplicassem a análise associativa em sistemas embarcados no contexto da Internet das Coisas.

## 3.1 Protocolo

O protocolo foi elaborado conforme especificado em: Kitchenham (2004); Biolchini *et al.* (2005); Mafra e Travassos (2006); Kitchenham e Charters (2007) e tem suas questões de pesquisas esquematizada a partir do paradigma GQM (*goal, question and metric*) descrito em Basili e McGarry (1997) conforme apresentado na Tabela 3.1:

Tabela 3.1: Objetivo definido a partir do paradigma *goal, question and metric*

| | |
|---|---|
| **Analisar** | Algoritmos de regras de associação |
| **Com o propósito de** | Identificar métodos, mecanismos, técnicas e plataformas. |
| **No que diz respeito a** | Aplicações em ambiente embarcado e/ou com base de dados limitadas |
| **Do ponto de vista do** | Pesquisador |
| **No contexto** | Acadêmico |

## 3.2 Questões de Pesquisa

Baseado nas informações definidas na Tabela 3.1 foram levantadas 5 (cinco) questões de pesquisas além da questão principal.

- **Questão principal**: quais os principais algoritmos de regras de associação usados em ambiente embarcado ou técnicas de extração de conhecimento em base de dados limitadas?
- **Questão 1**: quais algoritmos possuem experimentos em ambiente embarcado?
- **Questão 2**: quais algoritmos possuem experimentos em base de dados limitadas?
- **Questão 3**: sobre quais as plataformas / hardwares foram aplicados estes algoritmos?
- **Questão 4**: quais os mecanismos / técnicas foram usados para otimização das regras de associação?
- **Questão 5**: quais algoritmos/mecanismos/técnicas foram usados para correlacionar estados de dispositivos?

## 3.3 Fontes e *string* de busca

A biblioteca digital usada para obtenção dos artigos para esta revisão sistemática foi a SCOPUS (Elsevier (2019)).

Os critérios para sua seleção foram:

- Consulta de artigos em biblioteca digitais;
- Disponibilidade de consulta de artigos através da web;
- Presença de mecanismos de busca através de palavras-chaves e que suportem a *string* de busca;
- Ter os estudos disponíveis na língua inglesa;

A string de busca foi definida a partir das questões de pesquisa e do padrão PICO (*population, intervention, comparison, outcomes*) (Kitchenham e Charters 2007), conforme a estrutura abaixo:

- **População**: Algoritmos de regras de associação.
- **Intervenção**: Em ambiente embarcado ou base de dados limitadas.
- **Comparação**: Não se aplica.
- **Resultados**: Algoritmos, métodos, mecanismos e técnicas.

Tabela 3.2: Número de artigos obtidos durante a definição da *string* final

| ID | ***STRING*** | ARTIGOS |
|---|---|---|
| 01 | *("association rules"OR "associative analysis"OR "associative rule mining"OR "temporal association rule"OR "temporal relation")* | 16.309 |
| 02 | *("association rules"OR "associative analysis"OR "associative rule mining"OR "temporal association rule"OR "temporal relation") AND ("embedded"OR "constrain* data*"OR "limit* data*"OR "small data*"OR "tiny data*")* | 261 |
| 03 | *("association rules"OR "associative analysis"OR "associative rule mining"OR "temporal association rule"OR "temporal relation") AND ( "embedded"OR "constrain* data*"OR "limit* data*"OR "small data*"OR "tiny data*") AND ( "algorithm*"OR "mechanism*"OR "techniq*"OR "method*")* | 245 |

A Tabela 3.2 apresenta os resultados obtidos durante execução processo iterativo para obtenção da *string* de busca ideal e a seguir é apresentada a *string* final, no padrão SCOPUS (Elsevier 2019), utilizada no dia 07/03/2019 para obtenção dos artigos avaliados nesta revisão sistemática:

*( TITLE-ABS-KEY ( "association rules"OR "associative analysis"OR "associative rule mining"OR "temporal association rule"OR "Temporal relation") AND TITLE-ABS-KEY ( "embedded"OR "constrain* data*"OR "limit* data*"OR "small data*"OR "tiny data*") AND TITLE-ABS-KEY ( "algorithm*"OR "mechanism*"OR "techniq*"OR "method*") )*

## 3.4 Critérios de Inclusão/Exclusão

Para a seleção dos artigos foram adotados os seguintes critérios de inclusão:

- Mineração de dados em ambiente embarcado;
- Extração de conhecimento em bases com poucos dados;

- Regras de associação para correlacionar dispositivos.

Para os critérios de exclusão, foram consideradas as seguintes características:

- Uso de um extenso volume de dados;
- Coletânea de publicações, exceto *surveys*;
- Trabalho não aplicável à nenhum critério de inclusão;
- Publicação não disponível.

É importante destacar que, durante o processo de revisão sistemática, os critérios de inclusão consideram estudos que são de outras áreas. Isso permitiu explorar estudos que possuem busque resolver problemas com as mesmas características às apresentadas na Definição do Problemas (Seção 1.2) e alcançar objetivos similares aos descritos na Seção 1.4.

## 3.5 Extração de Informações

Os critérios de inserção e exclusão foram aplicados em dois momentos de filtragem:

I. Avaliação do Título, Resumo e Palavras-chaves;

II. Leitura completa do artigo e extração dos seguintes itens: Título, Resumo, Palavras-chaves, Fonte de publicação, Autores, Objetivos, Ano de publicação, Número de citações, Fator de impacto, Algoritmos usados, Técnicas envolvidas, Otimização, Plataforma/Ambiente de execução, Tamanho da base de dados, Comentários, Trabalhos futuros.

## 3.6 Resultados

Após a definição da string de busca, sua aplicação na ferramenta de busca da biblioteca Scopus retornou 245 dos quais 18 eram duplicados e 23 eram coletânia de artigos que não se caracterizavam como *surveys*. A Figura 3.1 apresentam sua distribuição dos artigos válidos (204) ao longo dos anos:

É possível observar que a partir de 2004 a quantidade de artigos cresceu significativamente e apresenta uma estabilidade no período de 2004 à 2015 havendo uma queda em 2016 mas se recuperando nos anos seguintes de tal forma que seu pico de publicações é apresentado em 2018 com 19 artigos publicados.

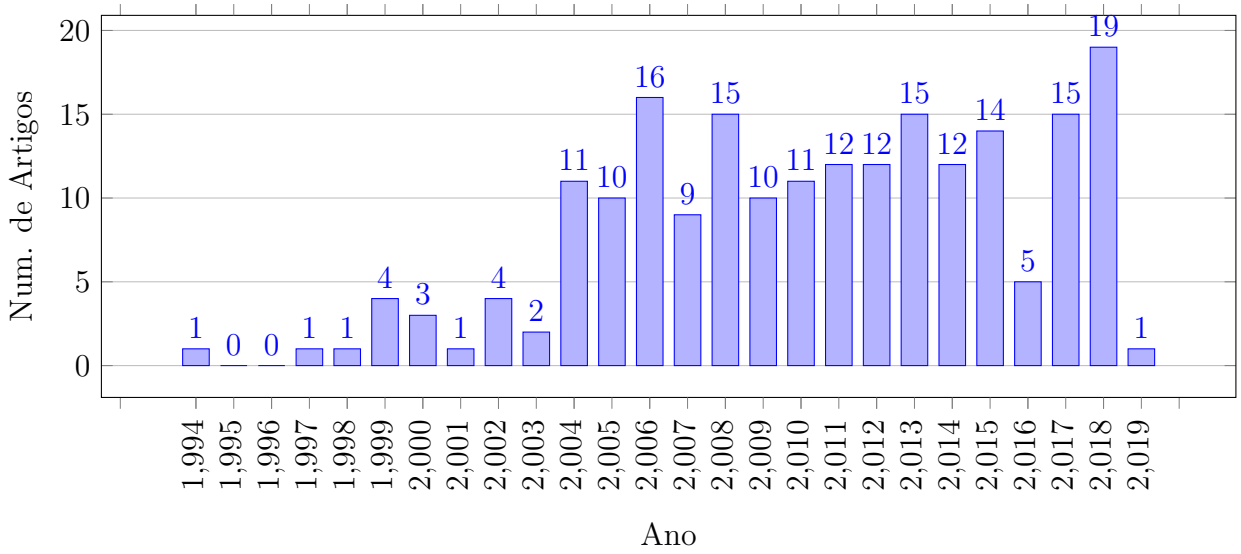


Figura 3.1: Distribuição de artigos por ano

Durante a aplicação dos critérios de inclusão e exclusão para o primeiro filtro, dos 204 artigos válidos, 120 se enquadraram nos critérios de seleção enquanto 87 foram rejeitados por se enquadrarem em algum critério de exclusão. Durante a análise mais profunda dos estudos no segundo filtro, 17 estudos apresentaram temáticas consoantes às questões de pesquisas da revisão.

A execução da revisão sistemática foi executada com auxílio da ferramenta *StArt* (Zamboni *et al.* 2010). Em Alencar (2019) é possível obter a lista de artigos (*.bib) obtida através da *string* de busca no Scopus, o arquivo de revisão sistemática (*. start), as fichas de extrações e o relatório geral da revisão gerada automaticamente pelo *StArt*.

Como principal contribuição a revisão foi capaz de responder as questões definidas na Seção 3.2:

- **Questão principal**: Quais os principais algoritmos de regras de associação usados em ambiente embarcado ou que tratam de bases de dados limitadas?

  *Apriori*, TITArl, JRip, *FP-Growth*, LFP, FARM, LSHAP, *Embedded Granular Computing*

- **Questão 1**: Quais algoritmos possuem experimentos em ambiente embarcado?

  *Apriori*, TITArl, JRip, *FP-Growth*, LFP, FARM, LSHAP, *Anomaly Detection Algorithm for Spatiotemporal Data*

- **Questão 2**: Quais algoritmos possuem experimentos em base de dados limitadas?

  *Apriori*, TITArl

- **Questão 3**: Sobre quais as plataformas / hardwares foram aplicados estes algoritmos?

  Cartão RFID, Processadores, Computador de propósito geral, MPSoC

- **Questão 4**: Quais os mecanismos/técnicas foram usados para otimização das regras de associação?

  Fila Circular, CVI (*Cluster Validity Index*), WTC (*Weighted transitive clustering*), Regressão Linear, *Random Forest*, PCA, *TreeNet*, OACCR (*Obtaining Accurate and Comprehensible Cassification Rules*), GPDCM (*Genetic Programming Data Construction Method*), *Effrom's Bootstrap*, *TinyDB*, SPIRIT, WARM, Escalonamento Circular, *Prefetching*, *K-Means Clustering*, Análise de conceito formal, *Design Space Exploration*

- **Questão 5**: Quais algoritmos/mecanismos/técnicas foram usados para correlacionar estados de dispositivos?

  Apriori, TITArl

As questões de pesquisa foram respondidas satisfatoriamente, havendo um ponto de convergência entre os estudos analisados. Dos quinze trabalhos identificados, dez utilizaram diretamente o algoritmo de Regra de Associação *Apriori*, geralmente associado a técnicas probabilísticas e/ou de reamostragem para validarem seus resultados enquanto os demais desenvolveram algoritmos baseado no *Apriori*, otimizando desempenhos e ou ajustando-o a uma instância do problema de padrões frequentes.

## 3.7 Resumo

O processo de revisão sistemática possibilitou a identificação de técnicas, métodos e mecanismos relevantes para extração de conhecimentos em base de dados limitadas. Não se limitando a isto, foi possível observar a versatilidade dos algoritmos de regras de associação os quais se estendem à solução de problemas em áreas distintas do conhecimento como exatas, humanas, biológicas e outras. Tal levantamento possibilitou a definição o algoritmo base para implementações de análise associativa, o qual foi utilizado como referência para a proposta apresentada nesta dissertação.

# Capítulo 4

# Trabalhos Correlatos

Os trabalhos correlatos apresentam os resultados obtidos durante a revisão sistemática, descrita no Capítulo 3, que possibilitou a avaliação de estudos cujos autores fizeram uso de algoritmos de regras de associação para extração de correlações em base de dados limitadas (com poucos registros), independentemente de sua área de aplicação.

Uma vez que todos os estudos tiveram por base o algoritmo de regra de associação Apriori definido por Agrawal e Srikant (1994), os mesmos foram agrupados em duas seções: estudos baseados no Apriori (conforme proposto originalmente) e estudos baseados em árvores. Em cada seção os estudos foram subdivididos em de acordo com sua otimização e técnicas envolvidas.

Nas considerações finais deste capítulo serão destacados quais métodos, técnicas e algoritmos possuem maior relevância para o objetivo geral desta pesquisa.

## 4.1 Estudos com Regras Associação *Apriori*

Usado em 10 (dez) dos 17 (dezessete) artigos identificados pela Revisão Sistemática, o Algoritmo de Regra de Associação *Apriori* (Agrawal e Srikant 1994) é explorado nessa seção de tal forma que os 7 deles foram agrupados considerando sua aplicação direta (Tabela 4.1) e os outros 3 estudos, apresentados na Tabela 4.2, apresentaram uma variação do algoritmo original.

A pesquisa conduzida por Sinaei e Fatemi (2018) descreve uma nova abordagem para otimizar o processamento de dados multimídia em plataformas MPSoC (*multi-processor system-on-chip*). Esta abordagem facilita a execução de novas aplicações, não conhecidas em tempo de projeto, por meio das regras de associação e do algoritmo de exploração do espaço. Tal abordagem identifica

Tabela 4.1: Artigos que usam Regras de Associação Apriori.

| **AUTOR(ES)** | ***TÉCNICA ENVOLVIDA*** |
|---|---|
| Mori *et al.* (2005) | *K-means Clustering* |
| Smith *et al.* (2009) | *Effron's Bootstrap* |
| McArthur *et al.* (2012) | Análise de conceito formal |
| Karimi-Majd e Mahootchi (2015) | *Clustering Validaty Index* |
| Pal *et al.* (2017) | - |
| Lynden (2017) | - |
| Sinaei e Fatemi (2018) | *Design Space Exploration Algorithm* |

o mapeamento mais eficiente para alocar as atividades nos processadores agrupando as aplicações que possuem forte correlações. Embora esta abordagem seja bem sucedida quanto à otimização da carga nos processadores, ela limita-se a definir grupos durante o tempo de design, não sendo ideal para processamento em tempo de execução.

Pal *et al.* (2017) buscaram, por meio de geração de invariantes, identificar em um sistema ciber-físico (ambiente controlado por sistemas computacionais) possíveis ataques à uma estação de tratamento d'água. Tal estudo limitou-se à correlacionar os estados dos sensores da estação de tratamento de tal forma que fosse possível identificar as invariantes durante um ataque ao sistema. Os experimentos identificaram 11.500 regras de associação para os 51 sensores existentes, logrando sucesso em responder sua questão de pesquisa ao constatar tais invariantes por meio das regras geradas em um dado espaço de tempo. No entanto é necessário ter acesso ao registro de todos os sensores do sistema ciber-físico, centralizando a solução em um só ponto de armazenamento e processamento de dados.

Lynden (2017) apresentou uma análise de URLs semânticas para suportar a vinculação automatizada de dados estruturados na *Web*. Embora seu estudo tenha usado técnicas de aprendizagem de máquina, as mesmas não foram usadas para otimizar ou complementar os resultados da aplicação das regras de associação *Apriori*, pelo contrário, o algoritmo foi aplicado para correlacionar o conhecimento obtido com uma base de dados de URLs (DBPedia) possibilitando, na *web* semântica, identificar páginas correlatas de forma mais eficiente. Os testes foram executados em uma base de dados com 5.000 URLs, uma quantidade limitada quando se comparada ao volume de URLs existentes na internet.

Karimi-Majd e Mahootchi (2015) apresentaram uma metodologia (ilustrada na Figura 4.1) que possibilita correlacionar informações de múltiplas fontes de dados para geração de novos serviços aos consumidores. Para tal, realizou-se a fusão (pré-processamento) de três bases de dados

distintas e, posteriormente, a mineração de regras usando o algoritmo *Apriori*. Finalmente, as regras extraídas são agrupadas usando a técnica *Cluster Validity Index*.

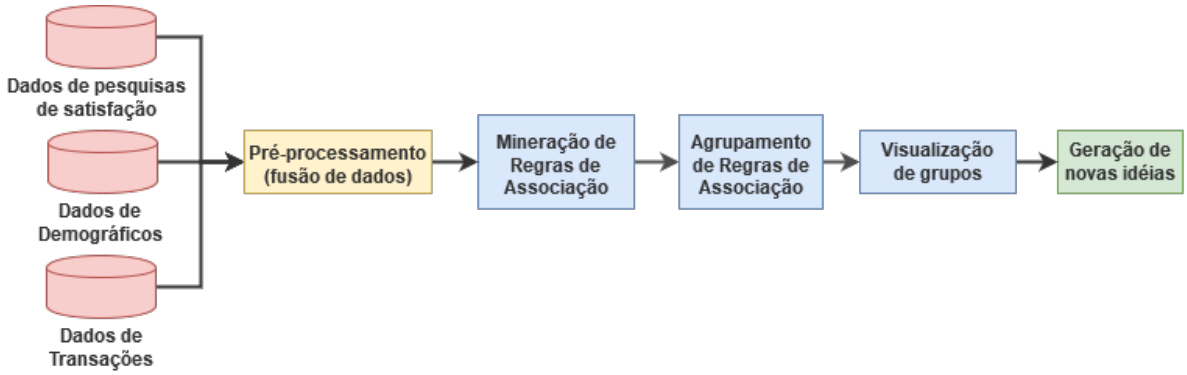


Figura 4.1: Visão esquemática da metodologia.
Adaptado de Karimi-Majd e Mahootchi (2015)

Os resultados obtidos por Karimi-Majd e Mahootchi (2015) são grupos de interesses em comum que correlacionam pesquisas de satisfação, dados demográficos e transações (produtos adquiridos pelos clientes). Com base nesses grupos é possível identificar quais os interesses mais relevantes dos consumidores por região, possibilitando novos serviços com base nos *clusters* de interesse.

McArthur *et al.* (2012) exploram o uso de regras de associação *Apriori* em base de dados pequenas, com o intuito identificar correlações entre a taxa de desemprego e fatores socioeconômico em regiões do sudoeste da Noruega. O auxílio da técnica de Análise de Conceito Formal (*Formal Concept Analysis*) possibilitou o estudo da correlação entre os atributos e as regiões sob um aspecto sistemático, validando seus estudos. A Análise de Conceito Formal é um método baseado em princípios de derivar uma hierarquia conceitual, ou ontologia formal, de uma coleção de objetos e suas propriedades. Cada conceito na hierarquia representa o conjunto de objetos que compartilham os mesmos valores para um determinado conjunto de propriedades; e cada subconjunto na hierarquia contém um subconjunto dos objetos nos conceitos acima dele.

Smith *et al.* (2009) apresentam um estudo no qual avaliaram a sensibilidade e confiabilidade das regras de associação geradas a partir de base de dados pequenas. Para validação do modelo, as regras foram avaliadas pelo método *Effron's Bootstrap*. Durante os experimentos foram correlacionados os dados obtidos pelas mamografias e suas respectivas avaliações laboratoriais (biopsia das massas/nódulos identificados pelas mamografias). A Figura 4.2 ilustra como é feita a avaliação das regras geradas.

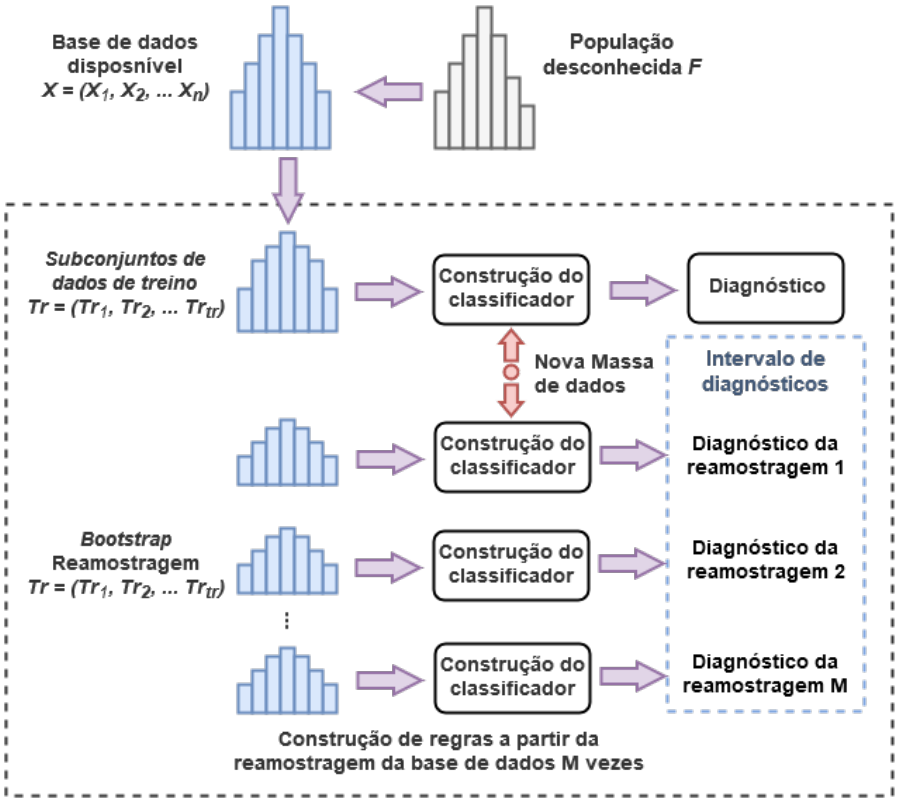


Figura 4.2: Visão geral da geração de modelos com *Effrom's Bootstrap*.
Adaptado de Smith *et al.* (2009)

Inicialmente um modelo é gerado usando uma amostra da base de dados (base de treino). Posteriormente são gerados novos modelos a partir de novas amostragens da base de treino. A quantidade de diagnósticos gerados pelos modelos obtidos com as amostragens determina o quão confiável o modelo/diagnóstico inicial é, ou seja, quanto menor a quantidade de diagnósticos diferentes, mais confiável.

Mori *et al.* (2005) buscaram, através de regras de associação temporal, que consistem em agrupar transações individuais em uma única transação considerando um espaço de tempo pré-determinado, identificar correlações entre os estados de um conjunto de sensores para determinar (ou antecipar) uma ação em ambiente residencial. Como metodologia, os autores definiram que a mudança de estados de uma sequência de sensores em um dado espaço de tempo determina um evento. Este espaço de tempo foi definido usando a técnica de agrupamento *K-Means*, a qual classifica as instâncias em grupos baseados no ponto médio entre as $K$ instâncias mais próximas ao centroide. Esse processo se repete até que não haja mais alteração de instâncias entre os grupos.

Além dos estudos descritos, durante a revisão sistemática foi possível identificar outros

3 (três) estudos (Tabela 4.2) que buscaram otimizar os resultados obtidos pelo algoritmo *Apriori* com uso de técnicas adicionais e comparar os resultados com outras técnicas de aprendizagem e mineração.

Tabela 4.2: Artigos que usam Regras de Associação Apriori e apresentaram alguma otimização

| **AUTOR(ES)** | **TÉCNICA ENVOLVIDA** | **OTIMIZAÇÃO** |
|---|---|---|
| Paul *et al.* (2012) | *Dempster-Shaffer* | Desempenho melhor que KNN, SVN, *Naive Bayes* e *Random Forest*. |
| Wang *et al.* (2013) | - | Redução/Compactação da Base de Dados. |
| Karthik (2015) | Fila Circular | Redução da complexidade. |

Paul *et al.* (2012) combinaram as regras de associação apriori com a Teoria de *Dempster-Shafer* para identificar associações probabilísticas entre um conjunto de características clínicas e o diagnóstico de displasia óssea. A teoria de *Dempster-Shafer* é uma teoria matemática que permite combinar evidências de diferentes fontes para chegar a um grau de confiabilidade (representada por uma função de credibilidade) que leva em conta todas as evidências possíveis. Durante os experimentos, para reduzir a complexidade do problema, o *itemset* foi limitado a no máximo 10 itens, no entanto não houve uma definição para o suporte mínimo uma vez que toda evidência (regra de associação) é válida para a Teoria de *Dempster-Shafer*. A metodologia proposta mostrou-se mais eficiente que árvores de decisão (ID3), *Random Forest*, *Naive Bayes*, SVM e KNN além de ser um pouco superior também aos diagnósticos clínicos.

Wang *et al.* (2013) realizaram a redução da base de dados removendo registros duplicados, o que impacta na redução do custo de processamento para a busca de *itemset* frequentes. Além disso, outra otimização proposta pelos autores é a compactação dos dados onde, considerando valores binários para representar a ausência (0) e presença (1) de um determinado item em uma transação, a compactação dá-se através da representação por uma *string* contendo uma lista de itens presentes e o seu índice em uma lista de controle (que contém todos os possíveis itens). Exemplificando, suponha-se que a lista de controle seja $X = \{pão, manteiga, queijo, presunto, café, leite, ovo...\}$, e o conjunto $D_j$ seja uma transação específica na qual possui $D_j = \{pão, manteiga, ovo\}$, logo, seus índices seriam $1_2 1_3 1_7$,de posse de tais índices é possível compactar a informação no seguinte código: “0110001”. Este mecanismo de compactação possibilita a otimização do uso do espaço de

armazenamento em sistemas embarcado durante os experimentos.

Embora os estudos apresentados por Karthik (2015) não sejam focados em extração de conhecimentos, os princípios de regras de associação foram aplicados para tornar a identificação de RFID mais segura. Sua proposta foi embarcar, nos cartões RFID, uma sequência de códigos, ordenados em uma fila circular, que pode ser identificada como um *itemset*. Quando estimulado, o cartão envia uma parte desta sequência, identificada como *footprint*, ampliando a quantidade de identificações possíveis em um cartão baseado nas possíveis combinações sequenciais de seus códigos. Além disso é possível remover um dos elementos do *itemset* como mecanismos de renovação da segurança sem disponibilizar o cartão. Tal comportamento só é possível por conta do princípio da anti-monotonicidade (definido na Seção 2.5.2).

## 4.2 Algoritmos baseados em árvores

Diferente do *Apriori*, os algoritmos apresentados nesta seção organizam seus dados em forma de árvore para identificar os *itemsets* mais frequente. Ressalta-se que dos 3 (três) estudos identificados, apenas 1 (um) não apresentou otimização em relação a outras técnicas, ou seja, é estudo não comparativo, embora tenha feito uso de técnica adicional. Os demais estudos propuseram algoritmos e/ou fizeram uso de outras técnicas e apresentaram uma análise comparativa validando a eficiência de suas propostas em relação aos outros estudos .

A Tabela 4.3 relaciona os estudos, os algoritmos utilizados, suas otimizações e as técnicas envolvidas. Assim como a sessão anterior, os estudos foram classificados inicialmente por seus algoritmos e posteriormente por sua contribuição em relação à otimização apresentada.

Tabela 4.3: Estudos com algoritmos baseados em árvores e suas contribuições e técnicas envolvidas.

| AUTOR(ES) | ALGORITMO | OTIMIZAÇÃO | TÉCNICA(S) ENVOLVIDA(S) |
|---|---|---|---|
| Ali (2012) | *FP-Growth* | Extração de conhecimento em base de dados pequenas | *Random Forest*, PCA e *TreeNET* |
| Shanmuganathan *et al.* (2014) | JRip | - | Regressão Linear |
| Gonzalez e Amft (2015) | TITArl | Desempenho melhor que HAC | *Weighted Transitive Clustering* |

Ali (2012) realizou a extração de correlações com uso do algoritmo *FP-Growth* (*Frequent Patterns Growth*). Este algoritmo faz parte de uma metodologia mais ampla (*Obtaining Accurate and Comprehensible Classification Rules* - OACCR) para extração de correlações em base de dados (pequenas e grandes) com informações médicas. A metodologia híbrida OACCR combina o uso de *Random Forest* para tratar valores ausentes, PCA (*Principle Component Analysis*) para redução de dimensionalidade da base de dados, *FP-Growth* para extração de regras de associação e, finalizando, com *TreeNet* para classificar as regras de associação geradas. A Figura 4.3 apresenta o diagrama de blocos da OACCR.

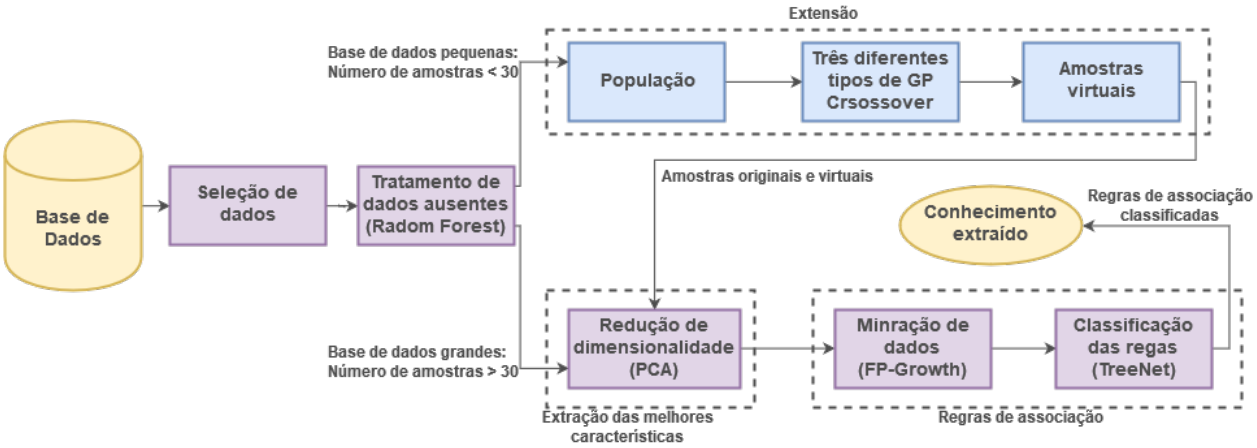


Figura 4.3: Diagrama de blocos OACCR.
Adaptado de Ali (2012)

Uma extensão é proposta no modelo para tratamento de base de dados limitadas onde um pré-processamento é permitindo a geração de instâncias virtuais baseadas na população original. Uma vez concluída, os dados (originais e virtuais) são enviados para o processo de extração, citado anteriormente

O estudo conduzido por Shanmuganathan *et al.* (2014) buscou, identificar os efeitos da variação de temperatura na produção de óleo de palma na Malásia. Através de uma abordagem híbrida, os autores analisaram uma pequena base de dados contendo informações sobre temperatura e o rendimento da produção de óleo de palma. As técnicas usadas foram árvores de decisão (J48), regressão estatística e regras de associação obtendo 78.9% de acurácia na identificação do período mais crítico para plantações, que é durante a abertura e permanência da flor totalmente aberta.

Gonzalez e Amft (2015) buscaram explorar a correlação entre grupos de sensores e atuadores em um edifício para comprovar sua hipótese de que há correlação nas alterações de estados dos

sensores de ambiente específico em um dado espaço tempo. Sua abordagem é ilustrada na Figura 4.4. O agrupamento de variáveis proposto por Gonzalez e Amft (2015), *Weighted Transitive Clustering* (WTC), baseia-se na correlação temporal existente entre a mudança de estados de sensores que monitoram um mesmo ambiente, ou seja, as variáveis agrupadas terão uma forte correlação durante suas mudanças de estado, do contrário, variáveis que pertencem a outros ambientes não serão correlacionadas. Com isso, baseado nas regras de relacionamentos, é possível inferir objetos relacionados ao mesmo espaço e que poderão ser agrupados.


Figura 4.4: Visão esquemática do agrupamento de variáveis.
Adaptado de Gonzalez e Amft (2015)

As regras de relacionamentos são obtidas através do algoritmo *Temporal Interval Tree Association Rule Learning* (TITArl), proposto por Guillame-Bert e Crowley (2012), o qual busca identificar quantas vezes um evento A implica em um evento $B$ considerando um limite de tempo máximo $r$ para que $B$ ocorra após $A$. Sua metodologia buscou comparar o algoritmo TITArl em relação aos métodos *Hierarchical Agglomerative Clustering* (HAC), *Random Choises Based* e *Mannually Rules Based*, obteve um rendimento superior a todos, de forma que foi possível agrupar 75% dos registros, considerando um intervalo de tempo $r$ de 15 segundos. Os demais algoritmos não identificaram os grupos corretamente ou criaram grupos com variáveis pertencentes a outros ambientes, apresentando correlações que não atendiam ao propósito do estudo.

## 4.3 Outras abordagens

Durante a revisão foram identificados também estudos (apresentados na Tabela 4.4) que desenvolveram seus próprios algoritmos ou realizaram alguma otimização em algoritmos existentes. Estes, por sua vez, demonstraram a eficácia de suas propostas apresentando estudos comparativos em relação a outras técnicas.

Tabela 4.4: Abordagens não baseadas no *Apriori*.

| AUTOR(ES) | ALGORITMO | TÉCNICA(S) ENVOLVIDA(S) |
|---|---|---|
| Qiu *et al.* (2006) | LSHAP | Escalonamento Circular e *Prefetching* |
| Gruenwald *et al.* (2007) | FARM | *TinyDB*, SPIRIT, WARM |
| Xu *et al.* (2008) | LFP | Algoritmos similares ao *apriori* |
| Fang *et al.* (2018) | E-GrC | *Rough Set Theory*, *Quotient space theory* |

Qiu *et al.* (2006) propuseram um algoritmo para minimizar o custo total de execução de programas para sistemas embarcados em tempo real. O *Loop Scheduling with Heterogeneous Assignment with Probability* (LSHAP), busca, por meio de análise probabilística, estimar o tempo necessário para execução de uma tarefa. O algoritmo primeiramente correlaciona as entradas aos seus tempos de execução da tabela de histórico. Posteriormente, faz-se uso do escalonamento rotativo como forma de melhorar o processo de atribuição e minimizar o custo total do processo. Finalmente, é realizado o *prefetching* para adiantar a preparação dos dados em tempo de execução.

O estudo de Gruenwald *et al.* (2007) buscou apresentar uma solução para estimar valores ausentes, corrompidos ou atrasados de leituras de um ou vários sensores. O algoritmo *Freshness Association Rule Mining* (FARM) realiza a estimativa de uma leitura de sensor ausente baseada em uma média ponderada da leitura atual dos sensores a ele correlacionados. Cada peso participante na média é derivado diretamente da força da associação correspondente do sensor. Ao atribuir a cada rodada um peso diferente, que cresce de acordo com sua ordem, é possível definir um mapeamento reversível entre um histórico inteiro de fluxo de um sensor e o conjunto de números reais. Isso permite que os dados sejam compactos e ainda suficientes para estimar.

Gruenwald *et al.* (2007) compararam o desempenho do FARM com outros três algoritmos, WARM, SPIRIT e *TinyDB* além de outros quatro métodos estatísticos, *Simple Linear Regression* (SLR), *Multiple Linear Regression* (MLR), *The Curve Regression* (CR) e Estimativas por média (AVG). As bases de dados possuíam 15% de dados ausentes. Em relação ao tempo de execução, o FARM é menos que 1 milissegundo mais lento que os demais métodos. A capacidade de estimar valores superou os 80% para FARM e WARM. Quanto a acurácia de classificação, o FARM obteve o melhor desempenho entre todos os métodos.

Xu *et al.* (2008) apresentaram uma estratégia que reduz a quantidade de itemsets inválidos,

ou seja, não frequentes. O algoritmo *Local Frequent Patterns* (LFP) pode ser aplicado à quaisquer algoritmos baseados no *Apriori* e possibilita reduzir significativamente a geração de *itemsets* desnecessários. Embora exija um consumo maior de memória, o LFP possibilita podar um espaço de busca inválido de forma eficiente. Sua estratégia está baseada na seguinte premissa: Dado um padrão frequente $p$ e seu conjunto de candidatos $C$, supõe-se que um item $a$ é o último item do padrão $p$. Para cada item em $b$ pertencente à $C$, qualquer padrão $p'$, se $b$ concatenado à $a$ não é frequente, então $b$ concatenado à $p'$ concatenado à $p$ também é não frequente.

Para validação da hipótese, Xu *et al.* (2008) implementaram dois algoritmos de regras de associação (MAFIA e SPAM), baseado no *Apriori*, usando o LFP e outras três propostas semelhantes FHUT, MHUT e PEP. Os resultados obtidos demonstraram que, para bases pequenas, o MAFIA+LFP obtiveram um desempenho 30% melhor que MAFIA+FHUT e MAFIA+MHUT. Já para base de dados densa, o desempenho foi similar, porém não tão rápido quanto o MAFIA+PEP. Comparando o SPAM+LFP com SPAM, para bases de dados pequenas, o primeiro obteve um desempenho 10 vezes melhor que o segundo, enquanto que em base de dados grandes este desempenho reduziu para 30% a 50% melhor. Xu *et al.* (2008), concluem que o LFP é eficiente quando tratando em base de dados pequena, porém seu rendimento cai quando aplicados a bases grandes.

Fang *et al.* (2018) propuseram um algoritmo para identificar itens frequentes em bases de transações baseado no processamento granular. Por meio da divisão e conquista, o *Embedded Granular Computing* (E-GrC) divide os superconjuntos em conjuntos menores para processá-los quanto a sua frequência na base de transações. Estes conjuntos menores são agregados porém são identificados com um novo rótulo (agrupando os itens) que permite reduzir o espaço de busca, já que os itens que os compõem já foram processados. A abordagem se mostrou mais eficiente que os algoritmos de mineração tradicionais como o *Apriori*, *FP-Growth*, AFOPT e DCI, porém sua aplicação se mostra mais eficiente em bases de dados grandes.

## 4.4 Resumo

Diversos estudos buscam identificar correlações implícitas em bases de dados limitadas e/ou correlacionar sensores/dispositivos em um ambiente monitorado. Alguns pesquisadores recorreram a métodos auxiliares pelos quais buscam reforçar a confiabilidade das regras obtidas pelos algoritmos de análise associativa. Dos 17 estudos apresentados, 10 fizeram o uso das regras de associação *Apriori*, 5 desenvolveram seus algoritmos baseados no *Apriori* e 4 apresentaram algoritmos de

análise associativa com abordagens diferenciada dos demais. O resultado desta pesquisa revelou a importância do algoritmo de regra de associação *Apriori* sendo este base para 15 dos 17 artigos identificados na revisão sistemática.

Pal *et al.* (2017) e Lynden (2017), fizeram a aplicação direta deste algoritmo sem uso de técnicas auxiliares, sendo este suficiente para obtenção dos resultados esperados pelos autores.

Os estudos de Mori *et al.* (2005); Karimi-Majd e Mahootchi (2015); Sinaei e Fatemi (2018); Fang *et al.* (2018) usaram métodos de agrupamento em momentos distintos no processo de extração de conhecimento. O primeiro realizou o agrupamento de registros individuais em um dado espaço de tempo para gerar transações (identificadas pelo autor como eventos) antes da extração de dados, ou seja, procedimento pré-extração e o segundo, agrupou as regras de associação para visualizar grupos de interesses comum, ou seja, procedimento pós-extração. Sinaei e Fatemi (2018) agrupam processos afins durante o design para otimizar o uso dos processadores, este processo executado antes da execução dos aplicativos. Já Fang *et al.* (2018) agrupam os dados e criam rótulos para tratar grupos de itens de forma escalável durante o processo de extração de correlações. Nos cenários expostos pelos autores o agrupamento antes, durante ou após a extração influenciou diretamente na análise associativa. Estes artigos levantaram a evidência de que o agrupamento de registros que possuíam correlações (temporais e/ou frequentes) podem ser uma ferramenta de auxílio para extração de conhecimento.

Nos estudos realizados por Ali (2012) e Smith *et al.* (2009) os autores obtiveram modelos (regras de associações) mais confiáveis por meio das técnicas de reamostragens de dados. Embora o primeiro tenha gerado reamostragem de dados reais e o segundo gerou amostras virtuais baseadas no padrão de registros dos dados originais, é possível notar que esse método se mostrou bastante eficiente uma vez que o aumento no volume de dados possibilita identificar padrões e correlações mais precisas, embora exijam mais armazenamento e processamento.

Ali (2012) destaca-se, assim como Xu *et al.* (2008); Shanmuganathan *et al.* (2014) e Gonzalez e Amft (2015), por optar em realizar o uso de um algoritmo de regra de associação baseado em árvores (FP-Growth, JRip e TITArl respectivamente). Este tipo de algoritmo, embora tenha um custo de processamento mais elevado, apresenta significativa otimização quanto ao espaço usado para armazenamento dos padrões frequentes durante o processo de extração das regras. Xu *et al.* (2008) otimizam ainda mais o uso de espaço de armazenamento realizando a podagem das árvores, retirando os itemsets menos frequentes.

Qiu *et al.* (2006) e Gruenwald *et al.* (2007) apresentaram métodos alternativos para análise

associativa, porém, seus algoritmos são baseados, assim como os demais, no *Apriori*. Ambos autores apresentaram suas próprias variações do algoritmo base (*Apriori*) e realizaram seus experimentos em base de dados embarcadas, buscando tratar problemas essenciais como baixa capacidade de armazenamento, processamento e memória. O primeiro apresentou um método de compactação e extração de conhecimento da base de dados, enquanto o segundo propôs um mecanismo para otimizar o uso de processadores em sistemas embarcados através de uma associação entre suas entradas e seus históricos de tempo de execução.

# Capítulo 5

# Método Proposto

O método proposto nesta dissertação, identificado como *eMbedded Associative Knowledge Extraction* (MAKE), ou Extração Embarcada de Conhecimento Associativo, é um mecanismo colaborativo que assume que cada dispositivo deverá comparar seu próprio padrão com os padrões dos demais dispositivos do ambiente buscando identificar as correlações mais fortes (baseado nas métricas *support*, *confidence* e *lift*) entre suas ações e as ações dos dispositivos remotos.

Cada dispositivo deverá realizar as comparações, periodicamente, em ambiente embarcado e as regras identificadas são aplicadas exclusivametne para o dispositivo que executou a análise, ou seja, cada dispositivo extrai seu próprio conjunto de regras. Por meio destas, é possível sincronizar o estado de um dispositivo remoto (consequente) de acordo com uma ação executada no dispositivo local (antecedente) caso ambas ações satisfaçam aos requisitos temporais de seus padrões, ou seja, caso ambas as ações estejam no mesmo slot de tempo.

Por meio de grupos de *multicast* (Venaas 2011) é possível definir que a busca de correlações explore um conjunto específico de dispositivos, possibilitando isolar as correlações de um determinado ambiente (e.g: dispositivos do quarto, dispositivos da sala, dispositivos da cozinha, etc)

Para melhor compreensão do método é necessário esclarecer previamente como os dispositivos deverão se comportar em relação aos seus conjuntos de estados e ações (Seção 5.1), ao armazenamento de dados e geração de padrões (Seção 5.2), ao processo de formação da base de transações (Seção 5.3) e, finalmente, ao processo de mineração descentralizada de correlações (Seção 2.5).

## 5.1 Estados e Ações do Dispositivo

Esta dissertação define que cada dispositivo deverá controlar apenas um único objeto e definir dois conjuntos: (i) $S = \{s_1, s_2, \ldots, s_k\}$ como um conjunto finito de $k$ itens que representam os possíveis estados que um dispositivo pode assumir, e (ii) $A = \{a_1, a_2, \ldots, a_i\}$ como um conjunto de $i$ ações disponíveis que permitem a transição entre os estados de $S$. Cada estado $s \in S$ representa uma interação do dispositivo com o ambiente (e.g: "lâmpada ligada"e "lâmpada desligada") e cada ação representa um estímulo, físico ou lógico, para alterar o estado do dispositivo (e.g.: "ligar"e "desligar").

Considerando que cada dispositivo atua de forma independente, os mesmos devem ser capazes de fornecer todos os recursos necessários para prover uma interface de interação pela qual serão tratados os estímulos físicos (sinais/interrupções) e/ou lógicos (requisições HTTP), bem como gerenciar seu armazenamento e processamento de dados independente dos demais dispositivos na rede.

## 5.2 Base de dados embarcada e padrões de mudanças

O método proposto também define uma forma específica de armazenar os dados brutos para mitigar a falta de recursos em dispositivos IoT. Diferentemente das tradicionais abordagens de mineração de dados e de aprendizado de máquina, o MAKE se concentra em registrar e analisar as ações que estimulam uma mudança de estado (padrão de ações), em vez de criar vários registros repetidos para mapear seus estados ao longo de um período (padrão de uso). Essa diferença reduz a quantidade de informações que devem ser armazenadas e analisadas durante a extração de conhecimento.

Assumindo que $T = \{t_1, t_2, \ldots, t_j\}$ é um conjunto finito de $j$ intervalos de tempo discretos (*slots*) e $A$ o conjunto de ações do dispositivo, conforme apresentado na Seção 5.1, é possível definir uma base de dados embarcada em uma matriz $M_{ij} = A \times T$ onde cada elemento $c_{ij} \in M_{ij}$ é um contador para cada ação $a_i \in A$ no *slot* $t_j \in T$. Na Tabela 5.1 é possível observar um exemplo desta matriz($M_{ij}$) para um dispositivo com duas ações ($A = \{"ON", "OFF"\}$) e com seis *slots* ($|T| = 6$).

Para reduzir o impacto de informações mais antigas, e excluir registros incomuns (i.e.: *outliers*), é necessário realizar uma transformação logarítmica (ver $M_{ij}$' na Tabela 5.1) em todos os contadores da matriz $M_{ij}$. Essa transformação é definida por: $c_{ij} \leftarrow \log_{|A|} c_{ij} \mid 1 \leq i \leq |A| \ \wedge \ 1 \leq j \leq |T|, \forall c_{ij} \in M_{ij}$. Além disso, se os valores de transformação forem menores que 1, o contador

assumirá o valor zero, caso contrário, ele assumirá o valor da operação logarítmica. Tal condição, representada no *slot* 5 na Tabela 5.1, evita que os contadores assumam valores negativos durante as transformações posteriores.

Tabela 5.1: Ilustração da base de dados embarcada ($M_{ij}$), Base transformada ($M_{ij}$') e Padrão de ações ($P$)

| **SLOTS (T)** | | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|---|
| **$M_{ij}$** | **ON** | 2 | 4 | 25 | 4 | 1 | 4 |
| | **OFF** | 5 | 4 | 17 | 2 | 0 | 5 |
| **$M_{ij}$'** | **ON** | 1 | 2 | **4.64** | **2** | 0 | 2 |
| | **OFF** | **2.32** | 2 | 4.08 | 1 | 0 | **2.32** |
| | **P** | OFF | - | ON | ON | - | OFF |

**$M_{ij}$** : Base de dados embarcada
**$M_{ij}$'** : Base de dados após transformação logaritmica
**P** : Padrão de ações gerado a partir de $M_{ij}$'

Uma vez que a transformação é realizada, é possível extrair um padrão confiável de ações (ver $P$ na Tabela 5.1) da seguinte forma: seja $C_{ij} = \{c_{1j}, \ldots, c_{ij}\}$ um conjunto contendo todos os $i$ contadores de coluna $j$ da matriz $M_{ij}$, e uma função $max\_action(C_{ij}, A)$ que retorna a ação (do conjunto $A$) correspondente ao maior valor de $C_{ij}$ (ou *nil* caso haja mais de um contador com o maior valor), então é possível criar um conjunto $P = \{p_1, \ldots, p_j\}$ *onde* $\forall p_j \leftarrow max\_action(C_{ij}, A) \wedge p_j \in A$ no intervalo de $1 \leq j \leq |T|$. Esta extração, baseia-se na análise probabilística (Seção 2.1) para definir o padrão de ações deste dispositivo. Caso haja ações de probabilidades iguais em um mesmo *slot*, nenhuma ação é repassada ao conjunto $P$, como pode ser observado nos *slots* 2 e 5 na Tabela 5.1.

Essa organização de armazenamento reduz a quantidade de dados que devem ser pré-processados para executar a análise associativa embarcada. É importante ressaltar que a transformação logarítmica deve ser considerada apenas caso não interfira no funcionalidade principal do dispositivo. Por exemplo, caso uma câmera de segurança tenha como principal objetivo identificar se alguém está portando uma arma de fogo, as ocorrências deste estado (arma de fogo detectada) seriam consideradas *outliers* e, durante a transformação logarítmica, seriam descartadas erroneamente. Para estes casos, recomenda-se que as transformações sejam executadas apenas para os demais contadores.

## 5.3 Base de transações

Através da combinação de pares de padrões de ações de diferentes dispositivos, é possível formar diferentes base de transações das quais é possível extrair correlações interessantes entre as ações dos dispositivos. Esta base de transações são geradas em momentos distintos (iterações) e não coexistem simultaneamente durante o processo de mineração, dadas as limitações de armazenamento e processamento dos dispositivos IoT. A fusão de um par de padrões é executada pelo dispositivo que está realizando a mineração das correlações, onde o seu padrão de ações (Padrão Local: $P_l$) se junta ao padrão de ações de um dispositivo remoto (Padrão Remoto: $P_r$) gerando a base de transação $D$ da seguinte forma: $D = \{(p_{l1}, p_{r1}), \ldots, (p_{lj}, p_{rj})\}$ onde $p_{lj} \in P_l$, $p_{rj} \in P_r$, e $1 \leq j \leq |T|$;. A Tabela 5.2 ilustra como um Dispositivo 1 combina seu padrão de ações ($P_1$) com outros padrões ($P_2$ e $P_3$) para formar as bases de transações ($D_{12}$ e $D_{13}$).

Tabela 5.2: Formação de Bases de Transações

| | **SLOTS** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **PADRÕES** | **$P_1$** | OFF | OFF | - | ON | OFF | OFF | ON | - | - | OFF |
| | **$P_2$** | - | OPEN | OPEN | - | OPEN | CLOSE | CLOSE | CLOSE | - | OPEN |
| | **$P_3$** | HIGH | - | LOW | LOW | - | HIGH | LOW | - | - | |
| **BASES DE TRANSAÇÕES** | **$D_{12}$** | OFF | OFF, OPEN | OPEN | ON | OFF, OPEN | OFF, CLOSE | ON, CLOSE | CLOSE | - | OFF, OPEN |
| | **$D_{13}$** | OFF, HIGH | OFF | LOW | ON, LOW | OFF | OFF, HIGH | ON, LOW | - | - | OFF |

O processo de criação da base de transação combina os padrões em pares ($D_{12} = P_1P_2$ e $D_{13} = P_1P_3$), no qual é possível gerar bases com tamanhos diferentes, conforme ilustrado nos *slots* 8 e 9 na Tabela 5.2. Durante a análise associativa, as regras extraídas de ambas bases não podem ser comparadas de maneira justa, uma vez que as métricas de *support* (Equação (2.2)) e *lift* (Equação (2.4)) são diretamente dependentes do tamanho da base de transação ($|D_{12}|$ e $|D_{13}|$).

Para contornar isto, as métricas de *support* e *lift* sofreram uma pequena modificação para atender às finalidades desta dissertação. Elas consideram que todas as bases de transações, gera-

das pelas combinações de dois padrões, terão seus tamanhos iguais à quantidade máxima de *slots* possíveis de se preencher em $P$, ou seja $|T|$. Esta ligeira modificação permite a comparação justa de regras obtidas a partir de bases de transações com tamanhos diferentes sem que para isso seja necessário agregar todos os padrões ao mesmo tempo.

## 5.4 Extração de Regras

O processo de extração de correlações ocorre individualmente em cada dispositivo IoT e as regras obtidas são válidas exclusivamente para o dispositivo que está executando a análise associativa. Para iniciar o processo de mineração, o dispositivo deve conhecer todos os demais nós da rede que tenham um padrão de ação para compartilhar. Portanto, cada dispositivo deve se unir a um grupo de *multicast* específico, compartilhado por todos, que possibilita gerar uma lista de endereços de rede (alvos) a partir de um pacote *multicast echo request/response* (Venaas 2011), ou seja, ao mandar um "*multicast echo request*"para o grupo compartilhado, todos os integrantes deste grupo responderão a requisição com "*multicast echo reply*", informando seu endereço de rede ao dispositivo solicitante.

As extrações são executadas diversas diversas vezes ao longo do tempo como mecanismo de identificar/descartar correlações de acordo com as mudanças no padrão de ações dos dispositivos. Tais mudanças são ocasionadas por eventuais mudanças na forma como o usuário interage com o dispositivo. Estas extrações, identificadas como *checkpoints*, incorporam ao modelo um dinamismo nas interações entre os dispositivos permitindo também se adaptar às mudanças de hábitos do usuários.

As regras obtidas durante a extração terão como seu antecessor uma ação do dispositivo que está executando a mineração e em seu consequente uma ação de algum dispositivo alvo. O número máximo de regras de um dispositivos está associado ao seu número de ações ou seja, o antecessor irá identificar uma única correlação para uma de suas ações. Tais regras atuarão como gatilhos no dispositivo antecessor que, ao serem acionados, irão disparar um estimulo lógico ao dispositivo consequente para execução da ação correlata.

Uma vez que o dispositivo obtenha a lista de alvos (endereço de rede dos demais dispositivos do grupo de *multicast*), é iniciado um processo iterativo, como ilustra a Figura 5.1.

- **Etapa I:** o dispositivo identifica seu próprio padrão de ações ($P_l$: Padrão Local) a partir da base de dados embarcada ($M_{ij}$);

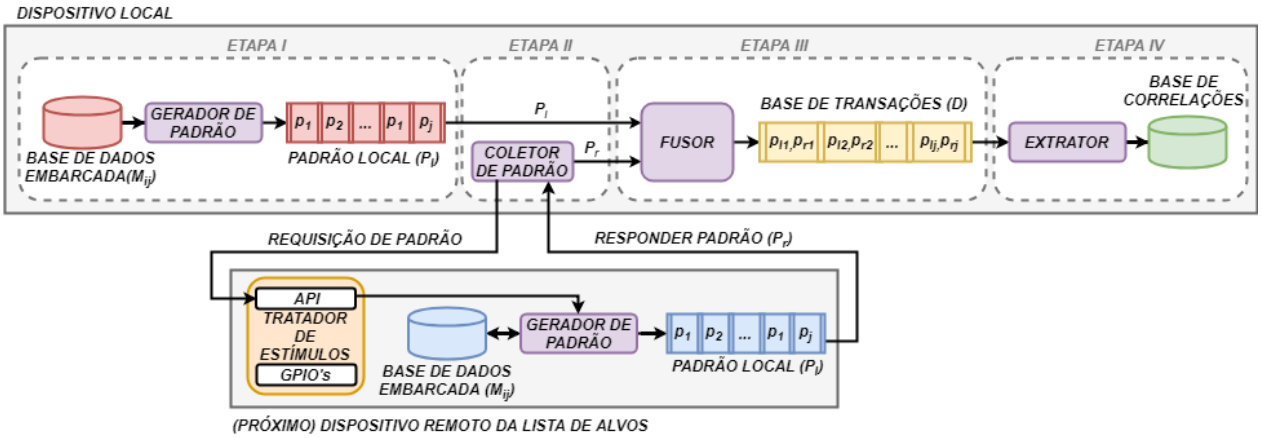


Figura 5.1: Etapas do método MAKE

- **Etapa II:** O componente *Coletor de Padrões* solicita o padrão de ações do (próximo) dispositivo na lista de alvo. O destino recebe a solicitação (por meio de sua própria API), executa a **Etapa I** em si mesmo e, em seguida, responde à origem seu padrão de ação, que é representado como $P_r$ (Padrão Remoto) na Figura; 5.1.

- **Etapa III:** o componente Fusor recebe ambos padrões ($P_l$ e $P_r$) e cria uma base de transações ($D$) unindo as ações de ambos padrões, *slot* por *slot* conforme apresentado na Seção 5.3;

- **Etapa IV:** por fim, é realizada a análise associativa na base de transações $D$ buscando identificar as correlações mais fortes para cada ação $a \in A$ do dispositivo local em relação as ações do dispositivo alvo. Então as correlações mais relevantes são comparadas com as obtidas anteriormente que estão armazenadas na Base de Correlações. As novas regras poderão substituir as já existentes ou serem descartadas de acordo com a sua importância, mensurada pelas métricas *support*, *lift* e *confidence* (nesta ordem de relevância).

Este processo se repete até que todos os alvos da lista sejam analisados e apenas a regra mais relevante para cada ação do dispositivo seja armazenadas na Base de Correlações.

## 5.5 Arquitetura do Dispositivo

Este seção descreve a arquitetura proposta para os dispositivo, seus componentes e como suas interações possibilitam a extração de conhecimento associativo em ambiente embarcado.

A Figura 5.2 ilustra como os componentes de um mesmo dispositivo interagem entre si e com o ambiente para registrar e extrair as informações necessárias para atender ao objetivo proposto

nesta dissertação:

- **Tratador de estímulos:** valida o estímulo de entrada, seja ele lógico, como requisições ou mensagens, seja ele físico, como interrupções nos pinos de entrada do microcontrolador (GPIO's). Se o estímulo for válido, então um sinal é encaminhado ao componente seguinte de acordo com a ação desejada, seja ela uma mudança de estado, uma requisição de informação (ex.: padrão, estado atual, identificação e outros), uma solicitação para habilitar / desabilitar / descartar uma regra de associação e outros. Também é possível implementar características adicionais, como interface do usuário, configurações gerais e funcionalidades personalizadas;

- **Controlador de estados:** realiza a alteração do estado atual do dispositivos e, no caso de um atuador, envia um sinal para GPIO's que corresponda a alteração requisitada. Uma vez que a ação é realizada, o componente consulta a base de dados para identificar qual a ação é mais provável de ocorrer e em seguida, realiza o incremento do contador correspondente. Se o estimulo de entrada for igual à ação mais provável de ocorrer no *slot* atual, o componente envia um sinal de volta ao Tratador de Estímulos com o intuito de notificá-lo que é necessário ativar a regra de associação para esta ação de entrada (se houver);

- **Sugestor:** gerencia as regras armazenadas na Base de Correlações. Provê a funcionalidade de enviar um estímulo lógico (requisição) para o consequente da regra disparada. Este componente também permite aos usuários ativarem e desativarem o disparo de requisições bem como descartá-las caso não satisfaçam aos interesses dos mesmos;

- **Base de Correlações:** armazena as regras que correlacionam as ações do dispositivo local com as ações do dispositivos remotos. Também são armazenadas as métricas de *support*, *lift* e *confidence* além do endereço de rede do consequente da regra. Cada ação do dispositivo pode possuir uma única regra;

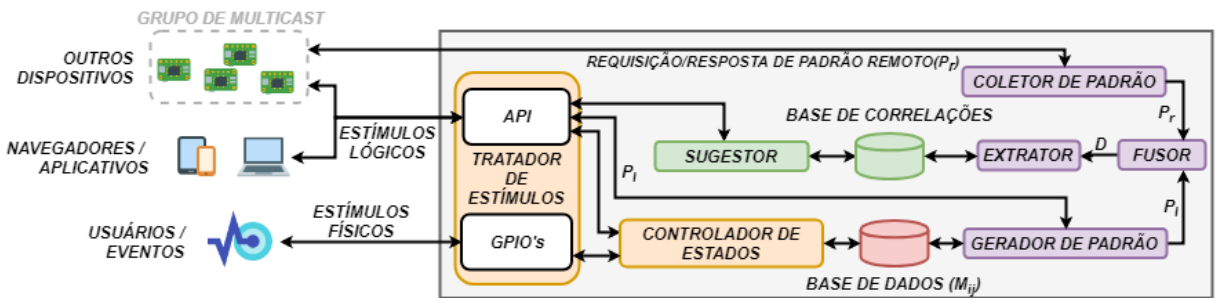


Figura 5.2: Visão geral da arquitetura

- **Base de dados** consiste em uma matriz de contadores conforme especificado na Seção 5.2;
- **Gerador de Padrões:** realiza a análise probabilística na Base de Dados para extrair o padrão de ações do dispositivo, ou seja, quais ações são mais prováveis de ocorrer em cada *slot* da matriz de contadores;
- **Coletor de Padrões:** identifica outros dispositivos na redes (lista alvos) que pertençam ao mesmo grupo de *multicast*, e coleta, iterativamente, os padrões de ações de cada dispositivo da lista;
- **Fusor:** realiza a fusão entre o padrão local (saída do Gerador de Padrões) e o padrão remoto (saída do Coletor de Padrões) para gerar a base de transações (D). Em outras palavras, une as ações mais prováveis de cada *slot* de ambos padrões;
- **Extrator:** executa a análise associativa na base de transação identificando as correlações mais relevantes entra as ações de ambos dispositivos, preservando, na base de Correlações, aquelas com maiores valores de *support*, *lift* e *confidence* nesta específica ordem.

Essa arquitetura deve ser a mesma para todos os dispositivos e as decisões de implementação devem compartilhar os mesmos parâmetros para realizar uma extração de conhecimento confiável das bases de dados distribuído que estão embarcadas nos dispositivos. É importante frisar que os valores como intervalos das extrações (checkpoints), número de itens no conjunto T (slots), valores mínimos de métricas são questões que deverão ser tratadas em fase de implementação, e não na definição do modelo, podendo variar de acordo com necessidade de aplicação.

## 5.6 Resumo

Neste capitulo foram descritos detalhadamente os componentes que integram o método proposto por esta dissertação. As Seções 5.1, 5.2 e 5.3 discorrem acerca do comportamento esperado dos dispositivos além de definir como é realizado o armazenamento, processamento e análises de dados em ambiente embarcado. Enquanto que a análise associativa, que busca identificar fortes correlações entre dois dispositivos, é apresentada na Seção 5.4 e define como as métricas tradicionais podem se adaptar para permitir que decisões locais, corroborem para formação de regras globalmente aceitas por todos os dispositivos sem a necessidade de concentração de dados.

# Capítulo 6

# Experimentos

A metodologia empregada para avaliar o desempenho da abordagem proposta é baseada em experimentos que consistiram comparar as regras de associação extraídas de uma única base de transação centralizada (abordagem tradicional) com as regras extraídas de diversas bases de transações formadas por pares de dispositivos, conforme as especificações do método proposto. Esta comparação possibilitou avaliar quão confiável é o método proposto (MAKE) ao lidar com a identificação de correlações baseadas apenas em decisões locais.

Vários experimentos foram executados para diferentes bases de dados públicas (*datasets*) e considerando diferentes intervalos de tempos entre as extrações, ou seja, para cada *dataset* a análise associativa foi executada em intervalos fixos de tempos identificados como *checkpoints* para que fosse possível observar as possíveis alterações entre as correlações a medida que novos registros foram inseridos na matriz de contadores dos dispositivos. Todos os conjuntos de dados foram pré-processados para desconsiderar registros duplicados, discretizar os valores contínuos e coletar apenas os registros que representavam uma mudanças de estado nos dispositivos.

Para executar a análise centralizada, o software de análises estatísticas R (R Development Core Team 2008) foi utilizado em conjunto da biblioteca *aRules* (Hahsler *et al.* 2011), que implementa o Algoritmo de Regras da Associação *Apriori* (Agrawal e Srikant 1994). Para cada *checkpoint*, todos os padrões dos dispositivos foram reunidos para criar uma base de transação única, a qual foi analisada pelo *aRules*. Isso nos permitiu obter todas as regras de todos os dispositivos em uma única execução do *Apriori*. As correlações identificadas pelo MAKE também foram armazenadas juntas em um único arquivo (*mrules.log*) assim como as que foram obtidas pelo *aRules* foram armazenadas no arquivo "*arules.log*". Para facilitar as comparações entre as regras, ambos arquivos armazenam

as regras em ordem decrescente pelas métricas *support*, *lift* e *confidence*.

As comparações consistiram em identificar se as correlações no arquivo *mrules.log* também eram as correlações mais relevantes no *arules.log*. Nesse caso, uma métrica chamada *hit* foi incrementada, caso contrário um segunda métrica (*miss*) foi incrementada.

Através destas métricas é possível obter a acurácia média dos experimentos, onde a taxa de *hits* e a taxa de *miss* são dadas pelas seguintes equações:

$$Taxa\ Hits = \frac{hits}{hits + miss} = \frac{num.\ de\ regras\ iguais\ para\ ambas\ abordagens}{total\ de\ regras\ extraídas} \quad (6.1)$$

$$Taxa\ Miss = \frac{miss}{hits + miss} = \frac{num.\ de\ regras\ diferentes}{total\ de\ regras\ extraídas} \quad (6.2)$$

Todos os *datasets*, *códigos fontes* e resultados dos experimentos podem ser encontrados no repositório do projeto em Alencar (2019).

## 6.1 Descrição dos *Datasets*

Foram utilizados 5 (cinco) *datasets* diferentes do projeto WSU CASAS (Cook *et al.* 2013) para realizar os experimentos. Esses *datasets* contêm registros brutos de vários sensores que monitoram diferentes ambientes, tais como: níveis de bateria, sensores magnéticos de portas, interruptores de luz, sensores de luz, sensores de movimento infravermelho e sensores de temperatura. A Tabela 6.1 descreve algumas outras informações destes *datasets* tais como: ambiente em que os sensores foram colocados, número de participantes, número dispositivos, número de dias de coleta de dados e a quantidade de registros gerados.

Tabela 6.1: Descrição dos *Datasets*

| ***Datasets*** | **Ambiente** | **Participantes** | **Dispositivos** | **Dias** | **Registros** |
|---|---|---|---|---|---|
| hh107 | Residencial | 2 | 110 | 371 | 3.369.689 |
| hh123 | Residencial | 1 | 88 | 588 | 2.907.282 |
| hh129 | Residencial | 1 | 86 | 668 | 12.303.984 |
| shib009 | Residencial | n/d* | 8 | 847 | 3.187.940 |
| tokyo | Trabalho | 9 | 67 | 115 | 802.534 |

***n/d**: não definido

Estes *datasets* foram selecionados para que pudessem estressar o método proposto quanto ao número de participantes, número de dias e quantidade de dispositivos. Dessa forma foi possível observar o comportamento do método em diferentes cenários.

## 6.2 Parâmetros dos experimentos

Para se aproximar o máximo possível de um cenário real, alguns parâmetros foram pré-definidos para os experimentos.

- Todos os dispositivos possuem base de dados independentes para cada dia da semana permitindo identificar as diferentes correlações entre os dispositivo ao longo da semana;
- Cada *slot* corresponderá a um intervalo de tempo de 15 minutos. Sendo 24 horas = 1440 minutos, então $|T| = 1440/15 = 96$ *slots*;
- O limiar mínimo da métrica *support* foi definido como o mínimo necessários para preenchimento de um único *slot* no padrão de ações, ou seja, 1%, para que haja possibilidades gerar *itemsets* de tamanho 2);
- O limiar mínimo da métrica *confidence* assumiu o valor de 90% uma este valor é padrão em softwares de análise estatísticas como R;
- Para o limiar mínimo da métrica *lift*, buscou-se identificar regras que correlacionam diretamente o antecedente e o consequente, ou seja Se A então B, dessa forma seu limiar mínimo é de 1.1.

O experimento foi executado para cada *dataset* considerando diferentes intervalos de tempo para execução da extração (Tabela 6.2) que permitiram observar o comportamento do método proposto de acordo com o volume de dados armazenados armazenado nas matrizes durante os intervalos das extrações.

| **INTERVALO** | **SEMANA** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | ... |
| **I** | X | X | X | X | X | X | X | X | X | X | X | X | ... |
| **II** | | X | | X | | X | | X | | X | | X | ... |
| **III** | | | | X | | | | X | | | | X | ... |

**I** - Extrações todas as semanas **II** - Extrações em semanas alternadas

**III** - Extrações à cada três semanas de coleta **X** - Pontos de extrações (*checkpoints*)

Tabela 6.2: Intervalos entre as extrações de correlações

## 6.3 Resultados

A Tabela 6.3 mostra os resultados da limpeza de dados antes da execução do experimento. Esse processo ignora os registros que não representam a alteração de estado de um dispositivo, conforme especificado na Seção 5.2, e transforma valores contínuos em intervalos discretos (discretização) com com base no intervalo de valores registrados.

Tabela 6.3: Pré-processamento dos dados.

| ***Dataset*** | **Registros** | | **Redução(%)** |
|---|---|---|---|
| | **Originais** | **Utilizados** | |
| hh107 | 3.369.689 | 2.811.279 | 16,57 |
| hh123 | 2.907.282 | 2.345.775 | 19,31 |
| hh129 | 12.303.984 | 56.523 | 99,54 |
| shib009 | 3.187.940 | 90.599 | 97,16 |
| tokyo | 802.534 | 171.483 | 78,63 |

Como pode ser observado, o processo de limpeza e discretização resultou em uma redução massiva nos registros de dados para os *datasets* hh129 e shib009, mais precisamente 99,54% e 97,16%, respectivamente. No *dataset* tokyo, a redução foi de 78,63% de seu conteúdo original. Os demais *datasets*, hh107 e hh123, tiveram reduções menores, de 16,57% e 19,31%, respectivamente. Isso ocorre devido ao grande número de dispositivos que registraram valores contínuos. Como esses valores precisavam ser discretizados, muitos registros não expressavam valores que representavam mudanças em seu estado discretos.

Para exemplificar tal caso podemos citar o sensor de temperatura T105 do *dataset* hh129. O mesmo possui registros de temperatura que variam de 17,5° à 33° e tem como temperatura média o valor de 25,25°. Durante os experimento, definu-se que os registros que tinham valores acima ou iguais à média foram rotulados como "HIGH"e os registros, abaixo da média, como "LOW", conforme apresentado na Figura 6.1. Uma vez discretizados, apenas os registros que se caracterizam como mudança de estado foram contabilizados, conforme especificado no modelo proposto na Seção 5.2. Dessa forma, para os quarenta primeiros registros do sensor T105, há apenas duas mudanças de estados: A primeira, que considera o primeiro registro do dispositivos, ou seja, mudança para "HIGH"e a segunda, no registro de número 27, no qual a temperatura cai de 28° para 24° ou seja, de "HIGH"para "LOW", sendo ambos registrados na base de dados embarcada através do incremento dos contadores em seus respectivos intervalos de tempo.

O mesmo comportamento se repete para vários dispositivos para todos os *datasets*. Nesse

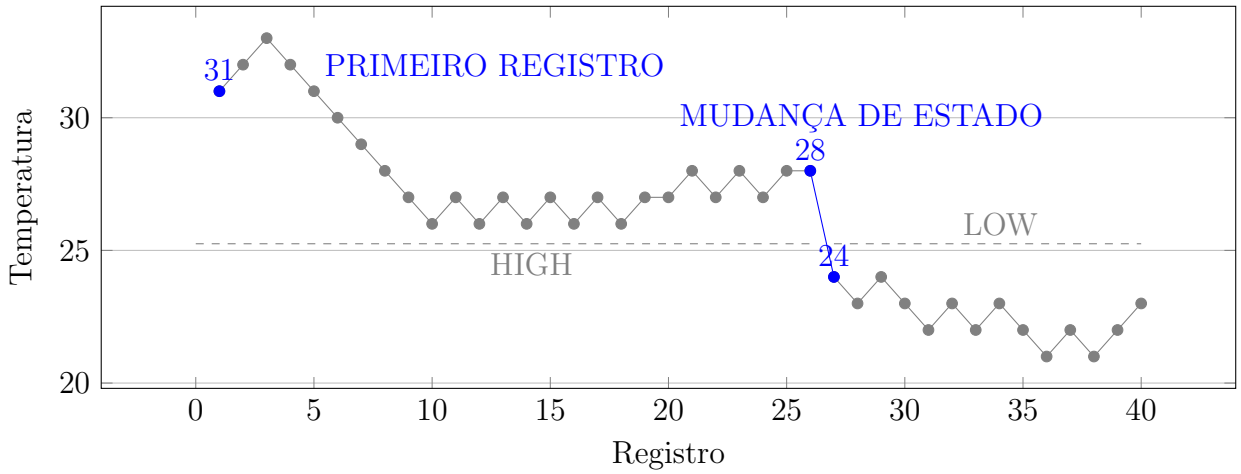


Figura 6.1: Exemplo de discretização de valores contínuos de um sensor de temperatura.

caso, o número de dispositivos que registram valores contínuos afeta diretamente a taxa de redução. Diferente de hh129, shib009 e tokyo, os intervalos de valores dos dispositivos nos *datasets* hh107 e hh123 eram menores, ou seja, havia menos variação nos valores de seus registros, reduzindo a quantidade de mudança de estados em relação ao rótulo discretizado. Além disso, há também um número menor de dispositivos que registram valores contínuos.

A Tabela 6.4 apresenta os resultados dos experimentos para todos os conjuntos de dados, apresentando o número de regras identificadas como as mais relevantes em ambas as análises (*hits*) e o número de regras identificadas apenas por MAKE (*misses*). Além disso, os valores são expressos através das Taxas Médias para cada *dataset* e também para o experimento como um todo.

Todos os experimentos para conjuntos de dados hh129, shib009 e tokyo obtiveram 100% de concordância. Em outras palavras, todas as regras identificadas pelo MAKE também foram as mais relevantes na análise do aRules.

Tabela 6.4: Resultado dos experimentos

| **Datasets** | ***Hits/Misses* por intervalo** | | | **Taxas médias (%)** | |
|---|---|---|---|---|---|
| | **I** | **II** | **III** | **Hits** | **Misses** |
| hh107 | 3.656/- | 3.493/- | 5.769/43 | 99,67 | 0,33 |
| hh123 | 1.952/- | 2.373/- | 4.040/15 | 99,82 | 0,18 |
| hh129 | 80/- | 54/- | 54/- | 100 | - |
| shib009 | 16/- | 135/- | 75/- | 100 | - |
| tokyo | 508/- | 337/- | 473/- | 100 | - |
| **Total** | **23.015 / 58** | | | **99,75** | **0,25** |

As exceções ocorreram para dois *datasets* (hh107 e hh123) considerando o Intervalo III.

O *aRules* discorda do MAKE em 43 regras para o *dataset* hh107 e 15 regras para as extrações do hh123. Nos experimentos para os Intervalos I e II, a taxa de *hits* do MAKE foi de 100%.

No total, a análise centralizada extraiu 23.015 correlações entre os dispositivos, já o MAKE identificou 23.073 regras, ou seja, 58 regras a mais. Isso representa 0,25% de todas as correlações encontradas. A taxa média de *hits* do MAKE é de 99,75% considerando todos os experimentos.

Ao analisar o relatório da experiência para identificar as razões pelas quais as regras não coincidiram/não foram identificadas em ambas abordagens, foi possível identificar que algumas regras tinham valores de métricas próximos ao limiares mínimos, definido em na Seção 6.2. Logo, a fim de verificar essa particularidade, para cada *checkpoint* que registrou valores *miss*, uma nova análise centralizada foi realizada considerando métricas mais permissivas (*support*=0.001, *lift*=1.01 e *confidence*=0.8), de tal forma que fosse possível identificar as regras em questão.

A Tabela 6.5 apresenta uma amostra desta particularidade em que as regras possuem diferentes antecedentes (LS023-HIGH, LS019-LOW e LS004-HIGH) e o mesmo consequente (LS021-LOW). Após a execução da análise com métricas permissivas, foi possível observar uma ligeira diferença em duas de suas métricas, tendo o *aRules* obtido valores superiores para *support* e inferiores para *lift* quando comparadas com as métricas obtidas pelo MAKE.

Tabela 6.5: Comparação de métricas das regras obtidas pelo MAKE e *aRules* para casos de não concordância

| **Regra** | **MAKE** | | | ***aRules*** | | | **Freq.** |
|---|---|---|---|---|---|---|---|
| | **Supp** | **Conf** | **Lift** | **Supp** | **Conf** | **Lift** | |
| $LS023(HIGH) \rightarrow LS021(LOW)$ | 0.5729 | 0.9821 | 1.1359 | 0.5978 | 0.9821 | 1.0886 | 55 |
| $LS019(LOW) \rightarrow LS021(LOW)$ | 0.7395 | 0.9861 | 1.1405 | 0.7717 | 0.9861 | 1.0930 | 71 |
| $LS004(HIGH) \rightarrow LS021(LOW)$ | 0.6250 | 0.9836 | 1.1376 | 0.6521 | 0.9836 | 1.0902 | 60 |

Embora as métricas *support* e *confidence* tenham satisfeito os limites mínimos estipulados nos parâmetros dos experimentos, a métrica *lift* apresentou valor inferior ao esperado, sendo este o motivo da não identificação da regra. A ligeira diferença entre os valores obtidos por ambas análises, ocorreu devido ao tamanho do *dataset*, o qual tinha dimensões inferiores ao tamanho do conjunto de intervalos ($T$). Sendo *support* e *lift* métricas que são diretamente afetadas pelo tamanho da base de transações, tais valores também se diferenciam à medida que o *dataset* gerado tenha valores inferiores à $|T|$. Para o *support*, conforme a Equação (2.2), quanto maior o tamanho da base de transação, menor é o valor e para o *lift*, Equação (2.4), quanto maior a base de transações, maior é seu valor.

Tal conclusão baseia-se o fato de no ambiente centralizado, todos os padrões são agrupados *slot* por *slot* e o tamanho da base de transação é igual ao número de *slots* preenchidos após o agrupamento, enquanto para a análise descentralizada, a base de transação sempre terá o mesmo tamanho que o número de *slots* da base de dados embarcada/padrão dos dispositivos, independentemente se todos os *slots* foram preenchidos, para que as métricas das regras obtidas possam ser comparadas sem a necessidade do algoritmo ter acesso à todas os padrões previamente.

A Figura 6.2 ilustra as correlações mais relevantes identificadas pelo MAKE durante a análise do *dataset* hh123 no *checkpoint* 15 às quartas-feiras para o Intervalo III.

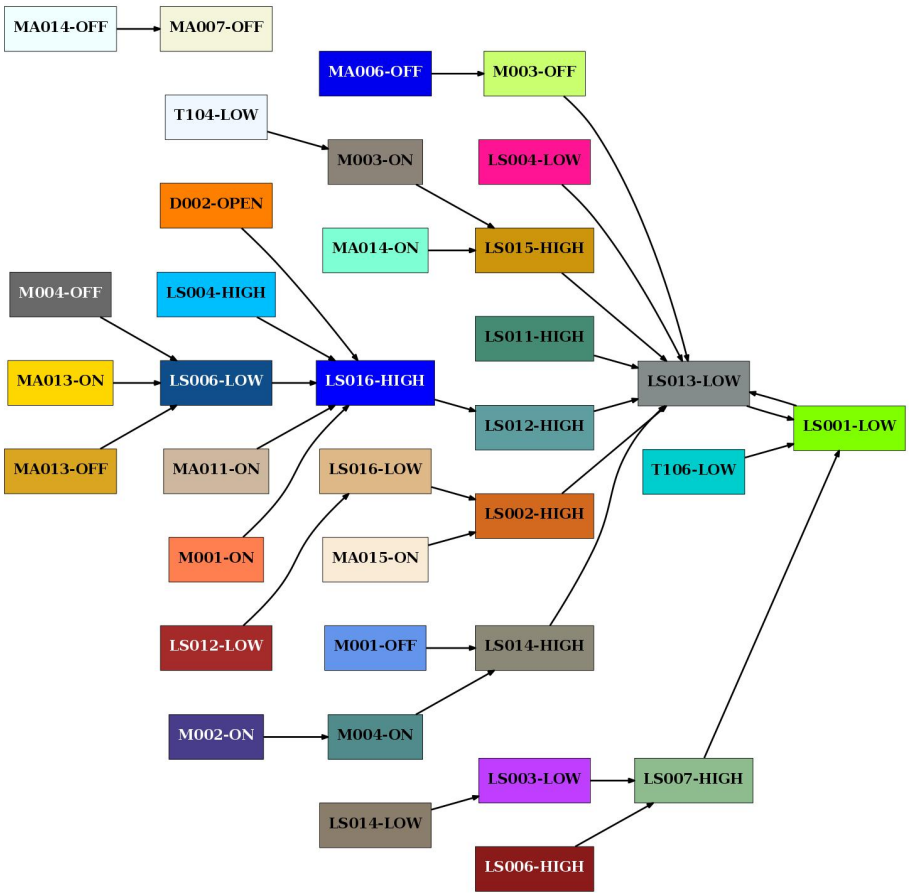


Figura 6.2: Ilustração das correlações obtidas pela mineração embarcada.

Cada par de dispositivos representa uma regra a qual correlaciona uma ação do antecedente a uma ação do consequente. As regras obtidas para cada dispositivo, permitiram gerar um ambiente que reage às interações dos usuários por meio requisições disparadas por gatilhos.

Embora a Figura 6.2 apresente uma estrutura similar a de um grafo dirigido, as correlações

entre os dispositivos devem respeitar os padrões de ações de ambos, antecedente e do consequente, para que um estímulo lógico (requisição HTTP) seja enviada pela origem e executado pelo destino. Desta forma, nem todos os caminhos observados na ilustração podem ser percorridos uma vez que as arestas (correlações) podem ser ativas em intervalos de tempos diferentes umas das outras.

Caso todos os padrões coincidam em um determinado *slots* então uma reação em cadeia pode ser iniciada em um dispositivo e ser propagada à vários outros, como por exemplo, se usuário solicitar a ação $ON$ para o dispositivo $MA013$, este pode disparar uma requisição para o $LS006$ executar a ação $LOW$, que por sua vez irá solicitar ao dispositivo $LS016$ executar a ação $HIGH$ que também irá solicitar a ação $HIGH$ em $LS012$ que irá estimuar a ação $LOW$ em $LS013$ e finalmente irá requisitar ao $LS001$ que executa a ação $LOW$.

É importante observar que esta abordagem é livre de ciclo, uma vez que, por definição, as ações mais prováveis no dispositivos são mutuamente excludentes, ou seja, um dispositivo não poderá executar uma ação A, disparar um gatilho e, consequentemente, ser estimulado (receber uma requisição proveniente de uma regra de outro dispositivo) à executar uma ação B no mesmo *slot*. Tal comportamento caracterizaria que as ações A e B são as mais prováveis de ocorrer no dado momento do disparo e recebimento das requisições.

## 6.4 Resumo

Neste capítulo foi descrito a metodologia utilizada para validação do método proposto bem como os parâmetros utilizados e os resultados obtidos.

Foram executados experimentos que consistiu em extrair as correlações de diferentes *data-sets* em diferentes intervalos de tempos para posterior comparação das regras extraídas pelo método proposto (MAKE) e a abordagem centralizada (aRules).

Os resultados obtidos foram descritos de tal forma que foi possível destacar as similaridades e diferenças para ambas análises bem como identificar características importantes quanto o método proposto.

# Capítulo 7

# Conclusão

O principal objetivo do MAKE é fornecer um mecanismo integrado para permitir que dispositivos IoT, que dispõem de recursos limitados (para armazenamento, gerenciamento, processamento de dados), extraiam conhecimentos úteis do ambiente em que estão inseridos. Esse mecanismo correlaciona pares de ações de dispositivos com o objetivo de oferecer um conjunto de sugestões de integração, por meio de solicitações HTTP, para os usuários. Essas sugestões, quando aceitas, permitem que os dispositivos controlem uns aos outros com base em seu padrão de ações, que pode estimular mudanças automáticas de estado entre dispositivos correlacionados. Além disto, o método proposto mitiga o problema do ponto único de falha evitando que os dispositivos se tornem dependente de um concentrador para identificar suas correlações.

O método proposto lida com pares de dispositivos e analisa apenas as transações que contêm informações úteis. Em um ambiente centralizado, a análise associativa considera desnecessariamente as correlações entre as ações do mesmo dispositivo, o método proposto explora apenas as correlações entre as ações de dispositivos diferentes, reduzindo a quantidade de informações que devem ser processadas. Além disto, o armazenamento de dados em forma de matriz permite reduzir a quantidade de dados que devem ser pré-processados para extração destas correlações.

Os experimentos mostraram que além das reduções massivas de dados, as correlações extraídas pelo método proposto e as extraídas pela análise associativa em ambiente centralizado coincidem, em média, 99,75%. Destaca-se ainda que este modelo limita as regas à intervalos específicos de tempo em que as ações correlacionadas devem ser as mais prováveis de ocorrer para que haja a integração entre ambos dispositivos.

Este dissertação também apresenta que o MAKE é uma alternativa interessante para im-

plementar um mecanismo embarcado para análise associativa capaz de identificar conhecimentos relevantes, válidos globalmente, sem a necessidade de agregar um grande volume de dados em um só local. O método também permite identificar mudanças no padrão de ações dos dispositivos e assim identificar novas correlações bem como descartar associações antigas.

Embora se mostre uma metodologia eficiente, o MAKE apresenta algumas limitações que devem ser exploradas em trabalhos futuros, como: (i) *dimensionalidade do dataset*, uma vez que a base de dados embarcada cresce proporcionalmente ao número de ações/estados disponíveis, o que poderia causar um alto consumo de armazenamento; (ii) intervalo de tempo discretizado, deve levar em consideração que os padrões compartilhados poderão ter dimensões de até $|T|$ itens, o que significa que cada dispositivos deverá tratar em memória duas vezes este tamanho ao realizara fusão dos padrões de ações; e (iii) *predições*, uma vez que a base de dados embarcada registra a ação em intervalos de tempo discretos (15 minutos, por exemplo), não significa que a ação esteja ativa em todo este intervalo, mas pode ser ativada a qualquer momento.

Como trabalhos futuros desta pesquisa, pretende-se criar protótipos para executar experimentos do mundo real e avaliar a experiência do usuário relacionada a sugestões para integração de dispositivos.

# Referências Bibliográficas


Agrawal, R. e Srikant, R. (1994), Fast algorithms for mining association rules, *in* Proc. of 20th Intl. Conf. on VLDB, pp. 487–499.

Alencar, M. (2019), Repositório da dissertação, https://github.com/macalencar/make. Acessado em: 2019-03-13.

Ali, S. (2012), Miner for oaccr: Case of medical data analysis in knowledge discovery, pp. 962–975.

Alvarado Moreno, J. D., Luis Garcia, L. C., Hernandez, W. C. e Barrera Obando, A. M. (2018), Embedded systems for internet of things (iot) applications: A review study, *in* 2018 Congreso Internacional de Innovación y Tendencias en Ingeniería (CONIITI), pp. 1–6.

Barr, M. e Massa, A. (2006), *Programming Embedded Systems: With C and GNU Development Tools*, O'Reilly Media, Inc.

Basili, V. e McGarry, F. (1997), The experience factory: How to build and run one (tutorial), *in* Proceedings of the 19th International Conference on Software Engineering, ICSE '97, ACM, New York, NY, USA, pp. 643–644. URL : http://doi.acm.org/10.1145/253228.253850

Berger, A. e Berger, A. (2002), *Embedded Systems Design: An Introduction to Processes, Tools, and Techniques*, CMP Books, Taylor & Francis. URL : https://books.google.com.br/books?id=3vY35UkvXrAC

Biolchini, J., Mian, P. G., Natali, A. C. C. e Travassos, G. H. (2005), Systematic review in software engineering, *System Engineering and Computer Science Department COPPE/UFRJ, Technical Report ES* 679(05).

Boulicaut, J.-F. e Jeudy, B. (2005), *Constraint-Based Data Mining*, Springer US, Boston, MA, pp. 399–416.

Buyya, R. e Dastjerdi, A. V. (2016), *Internet of Things: Principles and Paradigms*, 1st edn, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Chen, F., Deng, P., Wan, J., Zhang, D., Vasilakos, A. e Rong, X. (2015), Data mining for the internet of things: Literature review and challenges, *International Journal of Distributed Sensor Networks* 2015.

Cook, D. J., Crandall, A. S., Thomas, B. L. e Krishnan, N. C. (2013), Casas: A smart home in a box, *Computer (Long Beach Calif)* 46(7), 10.1109/MC.2012.328.

DeGroot, M. e Schervish, M. (2012), *Probability and Statistics*, Addison-Wesley. URL : https://books.google.com.br/books?id=4TlEPgAACAAJ

Elsevier (2019), Scopus elsevier indexing tools, https://www.scopus.com/. Acessado em: 2019-01-27.

Fang, G., Wang, J. e Ying, H. (2018), A novel model for mining frequent patterns based on embedded granular computing, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 26.

Gonzalez, L. e Amft, O. (2015), Mining relations and physical grouping of building-embedded sensors and actuators, *2015 IEEE International Conference on Pervasive Computing and Communications, PerCom 2015* pp. 1–10.

Gruenwald, L., Chok, H. e Aboukhamis, M. (2007), Using data mining to estimate missing sensor data, *in* Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007), pp. 207–212.

Guillame-Bert, M. e Crowley, J. L. (2012), Learning temporal association rules on symbolic time sequences, *in* S. C. H. Hoi e W. Buntine, eds, Proceedings of the Asian Conference on Machine Learning, Vol. 25 of *Proceedings of Machine Learning Research*, PMLR, Singapore Management University, Singapore, pp. 159–174. URL : http://proceedings.mlr.press/v25/guillame-bert12.html

Hahsler, M., Chelluboina, S., Hornik, K. e Buchta, C. (2011), The arules r-package ecosystem: Analyzing interesting patterns from large transaction datasets, *Journal of Machine Learning Research* 12, 1977–1981.

Han, J. (2005), *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Han, J., Kamber, M. e Pei, J. (2012), Data mining concepts and techniques, third edition.

Han, J., Pei, J., Yin, Y. e Mao, R. (2004), Mining frequent patterns without candidate generation: A frequent-pattern tree approach, *Data Mining and Knowledge Discovery* 8(1), 53–87. URL : https://doi.org/10.1023/B:DAMI.0000005258.31418.83

Health, S. (2003), *Embedded Systems Design*, Elsevier, Burlington, MA, USA.

Karimi-Majd, A.-M. e Mahootchi, M. (2015), A new data mining methodology for generating new service ideas, *Information Systems and e-Business Management* 13(3), 421–443. URL : https://doi.org/10.1007/s10257-014-0267-y

Karthik, K. (2015), Key search and adaptation based on association rules for backward secrecy, *in* 2015 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1–6.

Kireev, V. S., Guseva, A. I., Bochkaryov, P. V., Kuznetsov, I. A. e Filippov, S. A. (2019), Association rules mining for predictive analytics in iot cloud system, *in* A. V. Samsonovich, ed., Biologically Inspired Cognitive Architectures 2018, Springer International Publishing, Cham, pp. 107–112.

Kitchenham, B. (2004), Procedures for performing systematic reviews, *Keele, UK, Keele Univ.* 33.

Kitchenham, B. e Charters, S. (2007), Guidelines for performing systematic literature reviews in software engineering, 2.

Li, L., Li, Q., Wu, Y., Ou, Y. e Chen, D. (2018), Mining association rules based on deep pruning strategies, *Wireless Personal Communications* 102(3), 2157–2181. URL : https://doi.org/10.1007/s11277-017-5169-0

Lynden, S. (2017), Analysis of semantic urls to support automated linking of structured data on the web, pp. 1–6.

Mafra, S. N. e Travassos, G. H. (2006), Estudos primários e secundários apoiando a busca por evidências em engenharia de softwares, Technical report, COPPE/UFRJ. http://www.cin.ufpe.br/ in1037/leitura/EBSE-MafraTravassos-COPPE-2006.pdf.

McArthur, D., Encheva, S. e Thorsen, I. (2012), Exploring the determinants of regional unemployment disparities in small data sets, *International Regional Science Review* 35(4), 442–463.

Mori, T., Takada, A., Noguchi, H., Harada, T. e Sato, T. (2005), Behavior prediction based on daily-life record database in distributed sensing space, pp. 1703 – 1709.

Nazerfard, E. (2018), Temporal features and relations discovery of activities from sensor data, *Journal of Ambient Intelligence and Humanized Computing* . URL : https://doi.org/10.1007/s12652-018-0855-7

Pal, K., Adepu, S. e Goh, J. (2017), Effectiveness of association rules mining for invariants generation in cyber-physical systems, *in* 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE), pp. 124–127.

Paul, R., Groza, T., Hunter, J. e Zankl, A. (2012), Decision support methods for finding phenotype — disorder associations in the bone dysplasia domain, *PLOS ONE* 7(11), 1–10. URL : https://doi.org/10.1371/journal.pone.0050614

Perez, M., Jaramillo, D., Pinzón, D. e Herrera, F. (2018), Spectrum forecasting model for iot services, *in* 2018 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), pp. 877–881.

Qiu, M., Jia, Z., Xue, C., Shao, Z., Liu, Y. e Sha, E. (2006), Loop scheduling to minimize cost with data mining and prefetching for heterogeneous dsp.

R Development Core Team (2008), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Rose, K., Eldridge, S. e Chapin, L. (2015), The internet of things: How the next evolution of the internet is changing everything. URL : https://www.internetsociety.org/wp-content/uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf

Shanmuganathan, S., Narayanan, A., Mohamed, M., Ibrahim, R. e Khalid, H. (2014), A hybrid approach to modelling the climate change effects on malaysia's oil palm yield at the regional scale, *in* T. Herawan, R. Ghazali e M. M. Deris, eds, Recent Advances on Soft Computing and Data Mining, Springer International Publishing, Cham, pp. 335–345.

Sinaei, S. e Fatemi, O. (2018), Run-time mapping algorithm for dynamic workloads using association rule mining, *Journal of Systems Architecture* 91.

Smith, M., Wang, X. e Rangayyan, R. (2009), Evaluation of the sensitivity of a medical data-mining application to the number of elements in small databases, *Biomedical Signal Processing and Control* 4(3), 262–268.

Soong, T. T. (2004), *Fundamentals of Probability and Statistics for Engineers*, Wiley, Chichester; Hoboken, NJ.

Tan, P.-N., Steinbach, M. e Kumar, V. (2006), *Introduction to Data Mining*, Pearson Education.

Trifa, V., Guinard, D. e Carrera, D. (2015), Web things model thing, W3C proposed reccommendation, W3C. https://www.w3.org/Submission/wot-model/.

Venaas, S. (2011), Multicast Ping Protocol, RFC 6450. URL : https://rfc-editor.org/rfc/rfc6450.txt

Verhelst, M. e Moons, B. (2017), Embedded deep neural network processing: Algorithmic and processor techniques bring deep learning to iot and edge devices, *IEEE Solid-State Circuits Magazine* 9(4), 55–65.

Wang, X., Chen, M. e Chen, L. (2013), Research of the optimization of a data mining algorithm based on an embedded data mining system, *Cybernetics and Information Technologies* 13(SPECIALISSUE), 5–17.

Xu, Y., Ma, Z., Chen, X., Li, L. e Dillon, T. S. (2008), Improving frequent patterns mining by lfp, *in* 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1–4.

Yazici, M., Basurra, S. e Gaber, M. (2018), Edge machine learning: Enabling smart internet of things applications, *Big Data and Cognitive Computing* 2, 26.

Zamboni, A. B., Thommazo, A. D., Hernandes, E. C. M. e Fabri, S. C. P. F. (2010), Start uma ferramenta computacional de apoio à revisão sistemática, *in* 2010 Brazilian Conference on Software: Theory and Practice - Tools.