



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

Usando Regras de Associação para a Coordenação de Objetos Autônomos em Internet das Coisas

Lesly Delgado Pérez

MANAUS-AM

2026

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



Usando Regras de Associação para a Coordenação de Objetos
Autônomos em Internet das Coisas

Lesly Delgado Pérez

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, Dr.

MANAUS-AM

2026

Ficha Catalográfica

Elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

D352u Delgado Pérez, Lesly
Usando regras de associação para a coordenação de objetos autônomos
em internet das coisas / Lesly Delgado Pérez. - 2026.
68 f. : il., color. ; 31 cm.

Orientador(a): Raimundo da Silva Barreto.
Dissertação (mestrado) - Universidade Federal do Amazonas, Programa
de Pós-Graduação em Informática, Manaus, 2026.

1. Internet das coisas. 2. Regras de associação. 3. Coordenação de
objetos. 4. Mineração de dados. 5. Personas sintéticas. I. Barreto,
Raimundo da Silva. II. Universidade Federal do Amazonas. Programa de
Pós-Graduação em Informática. III. Título



Ministério da Educação
Universidade Federal do Amazonas
Coordenação do Programa de Pós-Graduação em Informática

FOLHA DE APROVAÇÃO

"USANDO REGRAS DE ASSOCIAÇÃO PARA A COORDENAÇÃO DE OBJETOS AUTÔNOMOS EM INTERNET DAS COISAS"

LESLY DELGADO PÉREZ

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

Prof. Dr. Raimundo da Silva Barreto - **Presidente**

Prof. Dr. Horácio Antônio Braga Fernandes de Oliveira - **Membro Interno**

Prof. Dr. Ruiter Braga Caldas - **Membro Externo**

Dr. Márcio André da Costa Alencar - **Membro Externo**

Manaus, 18 de setembro de 2026.



Documento assinado eletronicamente por **Raimundo da Silva Barreto, Professor do Magistério Superior**, em 21/09/2026, às 17:21, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Horácio Antônio Braga Fernandes de Oliveira, Professor do Magistério Superior**, em 22/09/2026, às 07:33, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Ruiter Braga Caldas, Coordenador de Curso**, em 22/09/2026, às 08:54, conforme horário oficial de Manaus, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufam.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **3334498** e o código CRC **6EECD65E**.

Avenida General Rodrigo Octávio, 6200 - Bairro Coroado I Campus Universitário Senador Arthur Virgílio Filho, Setor Norte - Telefone: (92) 3305-1181 / Ramal 1193
CEP 69080-900, Manaus/AM, coordenadorppgi@icomp.ufam.edu.br

Referência: Processo nº 23105.035230/2026-64

SEI nº 3334498

Agradecimentos

Aos meus pais e à minha família, por acreditarem em mim antes de qualquer resultado e por se fazerem presentes sempre. Nada do que está escrito aqui existiria sem vocês.

Aos meus amigos que ficaram longe, mas nunca ausentes: obrigada por cada mensagem, cada conversa e cada palavra de ânimo que atravessou a distância nos dias em que ela pesou mais.

Ao meu orientador, Prof. Dr. Raimundo da Silva Barreto, pela orientação atenta, pela confiança depositada neste trabalho e pelo rigor com que acompanhou cada uma de suas etapas.

Aos professores do Programa de Pós-Graduação em Informática do ICOMP/UFAM, pela formação recebida ao longo destes anos, e aos membros da banca, pelo tempo dedicado à leitura deste texto e pelas contribuições que o tornaram melhor.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Este trabalho foi parcialmente financiado pela Fundação de Amparo à Pesquisa do Estado do Amazonas – FAPEAM – por meio do projeto POSGRAD.

Você conseguirá tudo o que quiser!

Resumo

A coordenação eficiente de múltiplos dispositivos autônomos é um desafio central em ambientes de Internet das Coisas (IoT). Uma abordagem promissora é a mineração descentralizada de regras de associação, na qual cada dispositivo extrai localmente, a partir de seu histórico de uso, as relações que indicam quais ações tendem a ocorrer em conjunto, coordenando objetos sem depender de um servidor central. Entretanto, ao aplicar tal método sobre dados reais, não se dispõe de uma referência das coordenações verdadeiras, o que impede medir diretamente quanto da coordenação real é recuperado. Este trabalho implementa esse método de mineração de regras de associação em um simulador de ambiente residencial inteligente e o avalia sob uma verdade de referência (*ground truth*) controlada: modelam-se personas sintéticas por cadeias de Markov não homogêneas, com estrutura de coordenação conhecida, e mede-se, por precisão, *recall* (cobertura) e F1, a capacidade do método de recuperar as relações reais entre os dispositivos. Os experimentos, conduzidos sobre 5 residências com 4 moradores cada, com repetições e intervalos de confiança, mostraram precisão elevada e uma cobertura que decresce de forma sistemática com o tamanho do grupo coordenado, limite intrínseco à coordenação um para muitos por encadeamento, tornado mensurável pela avaliação com *ground truth*.

Palavras-chave: Internet das Coisas; Regras de Associação; Coordenação de Objetos; Mineração de Dados; Personas Sintéticas; *Ground truth*; Cadeias de Markov; Simulação.

Abstract

The efficient coordination of multiple autonomous devices is a central challenge in Internet of Things (IoT) environments. A promising approach is decentralized association rule mining, in which each device locally extracts, from its usage history, the relations indicating which actions tend to occur together, coordinating objects without relying on a central server. However, when such a method is applied to real data, no reference of the true coordinations is available, which prevents directly measuring how much of the real coordination is recovered. This work implements this association rule mining method in a smart residential environment simulator and evaluates it under a controlled ground truth: synthetic personas are modeled by time inhomogeneous Markov chains, with a known coordination structure, and the method's ability to recover the real relations between devices is measured through precision, recall (coverage), and F1. The experiments, carried out over 5 households of 4 residents each, with repetitions and confidence intervals, showed high precision and a coverage that systematically decreases with the size of the coordinated group, an intrinsic limit of one to many coordination by chaining, made measurable by the ground truth evaluation.

Keywords: Internet of Things; Association Rules; Object Coordination; Data Mining; Synthetic Personas; Ground Truth; Markov Chains; Simulation.

Sumário

1	Introdução	15
1.1	Contexto	16
1.2	Definição do Problema	18
1.3	Motivação	19
1.4	Objetivos	20
1.5	Organização do Trabalho	21
2	Fundamentação Teórica	22
2.1	Análise Probabilística e Cadeias de Markov	22
2.2	Sistemas Embarcados	24
2.3	Mineração de Dados	25
2.3.1	Análise Associativa	27
2.3.2	Regras de Associação: Algoritmos Fundamentais	28
2.4	Resumo	29
3	Trabalhos Correlatos	31
3.1	Mineração de Regras de Associação em Ambientes IoT	31
3.2	Arquiteturas Distribuídas e Múltiplos Objetos em MRA	32
3.3	Integração entre Regras de Associação e Coordenação de Objetos	33
3.4	DARE	34
3.5	Avaliação: Dados Reais e <i>Ground Truth</i>	35
3.6	Síntese Comparativa	35
3.7	Resumo	37

4	Método Proposto	38
4.1	Visão Geral do Método Proposto	38
4.2	Estados e Ações do Dispositivo	38
4.3	Registro de Eventos e Geração de Padrões	39
4.4	Formação da Base de Transações	40
4.5	Mineração de Regras Associativas	41
4.6	Coordenação por Encadeamento	43
4.7	Escopo por Cômodo e Combinação de Moradores	44
4.8	Modelagem de Personas por Cadeias de Markov	44
4.9	Definição Operacional do <i>Ground Truth</i> e das Métricas	48
4.10	Protocolo de Avaliação em Duas Fases	49
4.11	Reprodutibilidade e Desenho Experimental	49
4.12	Resumo	50
5	Resultados e Discussões	51
5.1	Precisão e o Efeito do Ruído	51
5.2	Cobertura e o Tamanho do Cluster Coordenado	52
5.3	Cadeia Cortada por Ciclos e Pares Mútuos	54
5.4	Sensibilidade à Confiança	55
5.5	Estabilidade entre Casas e Mineração por Cômodo	55
5.6	Convergência e Escolha de N	56
5.7	Resumo	57
6	Conclusão	59
6.1	Contribuições	59
6.2	Limitações	60
6.3	Trabalhos futuros	61
6.4	Considerações Finais	61
	Referências Bibliográficas	62

Lista de Figuras

4.1	Etapas do método (Alencar 2019)	42
5.1	<i>Recall</i> em função do número de coordenações verdadeiras por cômodo (confiança 0,75).	53
5.2	Sensibilidade das métricas (precisão, <i>recall</i> e F1) ao nível de confiança. . . .	55
5.3	Convergência da média acumulada de F1 em função do número de execuções.	57

Lista de Tabelas

3.1	Comparação dos trabalhos correlatos	36
4.1	Base de dados embarcada, base transformada e padrão de ações extraído . .	40
4.2	Formação de bases de transações a partir de padrões de múltiplos dispositivos	41
5.1	Efeito da confiança (média sobre as 29 instâncias)	51
5.2	Compartilhados vs. privados (média por confiança)	52
5.3	<i>Recall</i> médio por faixa de coordenações verdadeiras (confiança 0,75)	53
5.4	Diagnóstico (confiança 0,75, casos ilustrativos)	54
5.5	Média por casa (confiança 0,75)	56

Capítulo 1

Introdução

A Internet das Coisas (IoT) é uma infraestrutura de rede que conecta objetos físicos (dotados de sensores, atuadores e capacidade de processamento) à Internet, permitindo o compartilhamento ininterrupto de informações. Introduzido por Kevin Ashton em 1999, o conceito descreve sistemas em que dispositivos se comunicam sem intervenção humana ([Butun *et al.* 2019](#); [Sen *et al.* 2018](#)). Hoje, a IoT integra tecnologias como redes de sensores sem fio, sistemas embarcados e computação em nuvem, viabilizando aplicações que vão de automação residencial a cidades inteligentes. A arquitetura de IoT unifica dispositivos físicos e virtuais em uma malha integrada de comunicação, possibilitando a interoperabilidade eficiente entre o mundo real e o digital ([Abdmeziem e Tandjaoui 2014](#)).

Dispositivos IoT reúnem sensores, atuadores e microcontroladores que devem ser identificados e gerenciados remotamente, muitas vezes com processos automatizados de identificação e reação a eventos do ambiente. Tal heterogeneidade abrange desde eletrodomésticos até organismos biológicos monitorados, reforçando a necessidade de métodos eficientes de ordenação e mineração de padrões ([Sen *et al.* 2018](#)).

Entre as características da IoT estão o sensoriamento ubíquo, a computação pervasiva, a análise inteligente e a conectividade contínua ([Sen *et al.* 2018](#)). Com uma intervenção humana mínima, é possível coletar dados em larga escala, favorecendo ecossistemas computacionais autônomos, como residências inteligentes ou sistemas de irrigação automatizados para registrar, monitorar e ajustar a interação entre dispositivos ([Rani 2021](#)). Ademais, ao combinar computação em nuvem, análise de big data e tecnologias móveis, a IoT torna fac-

tível o desenvolvimento de serviços e aplicações inovadoras, capazes de extrair valor prático dos dados gerados pelos objetos conectados (Lynn *et al.* 2020).

1.1 Contexto

A era da IoT revolucionou os processos de geração e coleta de dados em diversos domínios como saúde, transporte, agricultura e comércio varejista (Lin, Zhou, You, Lin, An e Lü 2019; Mai *et al.* 2020). A propagação de dispositivos conectados tem impulsionado um crescimento de informações sem precedentes, estimando-se que o volume global de dados atinja cerca de 175 zettabytes até 2025 (Nomura *et al.* 2020).

Modelos centralizados concentram todo o processamento em um servidor único, o que simplifica a integração, mas cria pontos únicos de falha e aumenta a latência. Em contraste, modelos descentralizados distribuem decisões e processamento na borda da rede (*edge computing*, ou computação de borda), próximas aos dispositivos de origem dos dados, reduzindo a latência, reforçando a privacidade e permitindo que o sistema opere mesmo com conectividade intermitente.

No modelo centralizado, cada dispositivo apenas encaminha seus dados a um único servidor ou plataforma de nuvem, onde são integrados e disponibilizados aos usuários. Isso simplifica o gerenciamento e a integração de serviços, especialmente em ambientes com recursos restritos. Contudo, esse esquema apresenta diversos riscos. Entre eles estão o ponto único de falha, gargalos de comunicação, latência elevada em aplicações críticas, além da concentração de dados, que compromete a privacidade e a segurança. Também se destacam o alto consumo energético para envio contínuo de dados, a dificuldade de escalabilidade para bilhões de nós e a ausência de comunicação direta entre dispositivos, o que limita a interoperabilidade.

Nas arquiteturas descentralizadas, a hierarquia tradicional é rompida: os nós da rede operam de forma autônoma, trocando dados diretamente entre si, sem depender de um mediador central. Com isso, o sistema permanece funcional mesmo em contextos de baixa conectividade ou ausência de acesso constante à Internet, uma característica essencial para aplicações com alta exigência de disponibilidade e baixa latência. Esse modelo se beneficia do

deslocamento de parte do processamento para a borda da rede (como nos paradigmas de *edge* e *fog computing*), o que permite tratar os dados mais próximos de sua origem, reduzindo a latência e aumentando a escalabilidade. Em particular, o *edge computing* minimiza a dependência de servidores externos, reforça a privacidade ao evitar transferências desnecessárias de dados sensíveis e mitiga impactos de falhas na infraestrutura central.

Além disso, a descentralização favorece a interoperabilidade e permite interações inteligentes e em tempo real em cenários variados (de residências conectadas a cidades inteligentes), preservando a privacidade dos usuários. Com a evolução dos dispositivos IoT, espera-se que essa ênfase em arquiteturas distribuídas se intensifique, moldando um ecossistema cada vez mais autônomo, semelhante a uma web distribuída, sem pontos centralizados de controle ou processamento.

A mineração de Regras de Associação (MRA) emerge como ferramenta-chave para extrair padrões de interação em IoT, com aplicações que vão de automação residencial a monitoramento de tráfego. No entanto, algoritmos clássicos como *Apriori* demandam múltiplas varreduras de dados e recursos intensivos, o que os torna impraticáveis em dispositivos com CPU, memória e energia limitados (Khedr *et al.* 2019; Sarker *et al.* 2022).

A implementação de MRA em ambientes de IoT normalmente segue abordagens centralizadas ou distribuídas. Técnicas centralizadas, como a árvore de padrões de sensores, criam estruturas de dados baseadas em árvores que derivam regras de associação a partir de dados de sensores, varrendo o banco de dados apenas uma vez, o que reduz significativamente a sobrecarga computacional (Khedr *et al.* 2020). Para implementação distribuída, técnicas baseadas em *MapReduce* surgiram como soluções eficazes. Por exemplo, o *MapReduce-Apriori* aproveita recursos computacionais em múltiplos nós para melhorar o desempenho do processamento, alcançando uma **aceleração quase ideal**¹ quando comparado a técnicas de mineração centralizadas (Rashid *et al.* 2020). Da mesma forma, um algoritmo de otimização de enxame de partículas discretas autoadaptativo com operadores de algoritmos genéticos (GA-DPSO) foi proposto para otimizar o tempo de transmissão de dados ao posicionar da-

¹No contexto de mineração de dados distribuída, *aceleração* refere-se à razão entre o tempo de execução de um algoritmo em um único nó e o tempo de execução paralelo/distribuído com múltiplos nós. Uma aceleração ideal indicaria que o tempo de execução é dividido proporcionalmente ao número de nós, refletindo uma escalabilidade linear.

dos para fluxos de trabalho científicos em ambientes combinados de computação de borda e nuvem (Lin *et al.* 2021; Lin, Zhu, Zhang, Chen, Chen, Xiong e Mauri 2019).

No contexto de casas inteligentes, a extração descentralizada de conhecimento foi implementada para acomodar as rígidas restrições de armazenamento e processamento de dispositivos de IoT. Esses métodos identificam associações significativas entre ações de diferentes dispositivos de IoT por meio de análise associativa incorporada, utilizando métricas como suporte, confiança e elevação (*lift*) para determinar as relações mais relevantes (Alencar *et al.* 2020). Essas abordagens permitem a tomada de decisões autônomas no nível do dispositivo sem exigir os enormes recursos de processamento de dados normalmente associados a técnicas como o aprendizado profundo.

Visando explorar as vantagens de uma arquitetura distribuída aliadas à extração de conhecimento por meio de Regras de Associação, esta dissertação propõe a implementação de um mecanismo descentralizado. Tal mecanismo visa coordenar objetos autônomos em ambientes de Internet das Coisas. Esse mecanismo identifica associações implícitas entre ações e estados de dispositivos em um ambiente inteligente e, com base em padrões de associação extraídos, gera sugestões de coordenação, por exemplo, "ao perceber que a porta da sala se fecha, acionar o bloqueio eletrônico" ou "quando o nível de luminosidade externa cai, ajustar o brilho das lâmpadas internas" de modo a sincronizar automaticamente o comportamento dos objetos e satisfazer os fluxos de interação definidos pelo usuário.

1.2 Definição do Problema

Embora a mineração descentralizada de regras de associação permita que cada dispositivo descubra localmente relações de coordenação, extrair regras não é o mesmo que extrair as regras corretas. Ao aplicar o método sobre dados reais de uso, não se dispõe de uma referência que indique quais coordenações de fato existem no ambiente; sem essa referência, não é possível quantificar quanto da coordenação real o método recupera, apenas inspecionar as regras produzidas. Falta, portanto, uma forma de medir objetivamente a capacidade de recuperação do método, inclusive seus limites.

Assim, o problema investigado nesta dissertação é: como avaliar, sob uma verdade

conhecida (*ground truth*) e de forma reproduzível, a capacidade de um método de mineração de regras de associação de recuperar as relações reais entre os dispositivos de um ambiente residencial inteligente? Responder a essa questão exige:

- Um ambiente com estrutura de coordenação conhecida a priori.
- Métricas que quantifiquem não só o acerto (precisão), mas também a cobertura (*recall*) das coordenações verdadeiras.
- Um procedimento reproduzível para gerar os dados e minerar as regras.

1.3 Motivação

A crescente adoção da IoT em contextos domésticos, industriais e urbanos impõe desafios substanciais à coordenação eficiente de múltiplos dispositivos autônomos. Em ambientes onde decisões precisam ser tomadas em tempo real e com recursos computacionais limitados, modelos centralizados tornam-se ineficazes, pois introduzem gargalos de comunicação, aumentam a latência e comprometem a privacidade dos dados. Nesse cenário, mecanismos que permitam a extração local de conhecimento e a tomada autônoma de decisões revelam-se não apenas desejáveis, mas essenciais para garantir a escalabilidade, a resiliência e a adaptabilidade de sistemas distribuídos.

Ao permitir que cada dispositivo descubra padrões comportamentais a partir de seus próprios dados e das interações com vizinhos próximos, cria-se um ecossistema capaz de operar de forma autônoma, mesmo diante de falhas de conectividade ou da ausência de infraestrutura central. Isso contribui para uma redução significativa da dependência de servidores externos, ao mesmo tempo em que reforça a segurança e a privacidade, dado que os dados sensíveis permanecem localmente armazenados e processados. Além disso, a extração embarcada de regras de associação possibilita que os dispositivos atuem de forma proativa, antecipando eventos com base em padrões previamente observados e sincronizando ações de maneira inteligente.

Tais capacidades favorecem a criação de ambientes mais responsivos e personalizados, capazes de adaptar suas funções conforme os hábitos dos usuários. Por exemplo, é

possível ajustar automaticamente a iluminação e a climatização de um cômodo com base nos comportamentos observados em momentos anteriores, promovendo conforto, economia de energia e autonomia funcional. Assim, a presente proposta se insere em uma tendência contemporânea que busca transformar dispositivos passivos em objetos inteligentes e colaborativos, capazes de aprender, decidir e agir em coordenação com outros objetos do ambiente, um passo fundamental rumo à construção de sistemas IoT verdadeiramente distribuídos e inteligentes.

1.4 Objetivos

O objetivo principal é implementar e avaliar o método de extração de regras de associação para coordenação de objetos autônomos em IoT, por meio de um simulador de ambiente residencial inteligente baseado em personas, validando a capacidade do método de recuperar as relações reais entre dispositivos sob um *ground truth* controlado.

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

- Implementar, em um simulador de ambiente residencial inteligente, o método de mineração de regras de associação entre dispositivos;
- Modelar personas sintéticas por meio de cadeias de Markov não homogêneas no tempo (transições de estado por faixa horária), com estrutura de coordenação conhecida (*ground truth*), organizadas por cômodo e por casa (com múltiplos moradores);
- Gerar os dados de uso e minerar as regras por cômodo, seguindo um protocolo de duas fases, treinamento (geração das interações, sem mineração) e teste (mineração das regras e avaliação);
- Definir e aplicar métricas de avaliação baseadas no *ground truth*, precisão, *recall* e F1, para caracterizar a capacidade do método de recuperar as coordenações reais entre dispositivos, bem como seus limites; e

- Analisar a sensibilidade dos resultados aos parâmetros de mineração (em especial à confiança) e assegurar a reprodutibilidade por meio de semente determinística e N repetições, reportando média e intervalo de confiança de 95%.

1.5 Organização do Trabalho

No **Capítulo 1** a introdução deste trabalho é apresentada por meio da contextualização do tema, incluindo as motivações, a definição do problema, os objetivos da pesquisa e a estrutura da dissertação.

O **Capítulo 2** enfatiza a fundamentação teórica, revisando os conceitos necessários à compreensão do trabalho: a análise probabilística e as cadeias de Markov, que embasam a modelagem do comportamento das personas; os sistemas embarcados e a comunicação descentralizada, que contextualizam o cenário de aplicação do método; e a mineração de dados, com ênfase na análise associativa e nas métricas de suporte, confiança e *lift*.

O **Capítulo 3** descreve a apresentação e discussão dos trabalhos correlatos referentes ao objeto de pesquisa nesta dissertação. O capítulo possibilita uma visão geral dos principais modelos e técnicas usados para solucionar problemas similares aos apresentados durante a definição do problema.

O método proposto é detalhado no **Capítulo 4**. Primeiro, descreve o mecanismo de mineração de regras de associação para coordenação de dispositivos. Em seguida, apresenta a metodologia experimental adotada para avaliá-lo.

Os **Capítulos 5 e 6** apresentam, respectivamente, os resultados e as discussões e as conclusões deste trabalho. No **Capítulo 5** são reportadas as métricas de precisão, *recall* (cobertura) e F1, a análise de sensibilidade aos parâmetros de mineração e a interpretação dos achados, incluindo os limites de cobertura da coordenação. Já o **Capítulo 6** traz a conclusão, com a síntese das contribuições.

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, e de como se adéquam ao desenvolvimento dos experimentos e contribuem para o alcance dos objetivos da pesquisa.

2.1 Análise Probabilística e Cadeias de Markov

A teoria da probabilidade fornece os instrumentos fundamentais para a elaboração e a avaliação de modelos matemáticos destinados ao estudo de fenômenos aleatórios. Ao estudar um evento aleatório, lidamos com um procedimento cujo resultado não pode ser antecipado com precisão. Exemplos clássicos desse tipo de situação são encontrados em jogos de azar, um campo que, aliás, estimulou os primeiros avanços da teoria probabilística nos séculos XV e XVI (Soong 2004).

De acordo com DeGroot e Schervish (2012), chama-se **experimento** a qualquer procedimento, real ou hipotético, cujo conjunto de resultados possíveis seja identificável de antemão, e **evento** a qualquer subconjunto bem definido desses resultados. Sob essa perspectiva, a relevância do estudo de eventos aleatórios reside na capacidade de inferirmos, com base em tais eventos, diferentes conclusões sobre sua ocorrência, tais como:

E_1 : *O lançamento de uma moeda e a observação da face voltada para cima*

O espaço de resultados (Ω) é discreto se há um número, finito ou infinito, numerável de elementos. Se Ω 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, neste caso, admite apenas dois eventos: *cara* ou *coroa* cada qual correspondendo a um ponto amostral. Em muitos contextos, interessa quantificar a probabilidade de ocorrência de um dado evento A. 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:

$$P(A) = \frac{N_A}{N} = \frac{\text{número de elementos em } \Omega \text{ iguais a } A}{\text{número de elementos em } \Omega} \quad (2.1)$$

Ao combinar dois experimentos, também combinamos seus espaços de resultados. Por exemplo, seja E_3 o experimento que consiste no lançamento de duas moedas, com registro das faces voltadas para cima. Aqui definimos E_1 como o lançamento e a observação da primeira moeda, e E_2 como o lançamento e a observação da segunda. Dessa forma, o espaço amostral de E_3 é obtido pelo produto cartesiano dos espaços amostrais de E_1 e E_2 , resultando em todos os pares possíveis de resultados:

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

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

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

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:

$$\mathbf{E}_3 : P_{E_3}(\text{cara}, \text{cara}) = 1/4 = 0,25 = 25\%$$

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

A probabilidade de obter “cara, cara” é de 25 %, pois, entre os quatro resultados possíveis no espaço amostral, apenas um corresponde a esse evento. Outra maneira de chegar a esse valor é multiplicar as probabilidades individuais de cada lançamento; isto é, $(P_{E_1}(\text{cara}) \times P_{E_2}(\text{cara}))$. Esse procedimento ilustra como se pode calcular a chance de um determinado acontecimento num conjunto de resultados, e também evidencia que o tamanho do espaço amostral pode variar conforme as especificidades do experimento, o número de repetições e as possíveis combinações dos espaços de resultados de cada etapa.

Uma cadeia de Markov é um processo estocástico no qual a probabilidade do próximo estado depende apenas do estado atual, e não de toda a história anterior, a chamada propriedade de Markov (ausência de memória) [DeGroot e Schervish \(2012\)](#). Formalmente, para uma sequência de variáveis aleatórias $X_0, X_1, X_2, \dots$ que assumem valores em um conjunto finito de estados S ,

$$P(X_{t+1} = j \mid X_t = i, X_{t-1}, \dots, X_0) = P(X_{t+1} = j \mid X_t = i) = p_{ij},$$

em que p_{ij} é a probabilidade de transição do estado i para o estado j [Ross \(2014\)](#). Em uma cadeia homogênea, essas probabilidades são constantes ao longo do tempo; em uma cadeia não homogênea no tempo, elas variam conforme o instante t , isto é, $p_{ij}(t)$.

2.2 Sistemas Embarcados

Sistemas embarcados são sistemas computacionais especializados, destinados à execução de funções específicas, frequentemente como parte de sistemas mecânicos ou eletrônicos maiores. Ao contrário de computadores de uso geral, como computadores pessoais, voltados à flexibilidade e adaptados a uma ampla gama de necessidades do usuário, os sistemas embarcados são otimizados para tarefas específicas ([Kumar 2020](#); [Bader e Abusitta 2013](#)). Normalmente,

consistem em uma combinação de componentes de hardware (microprocessadores ou microcontroladores), software e, às vezes, *firmware* (Winarno *et al.* 2022; Anirudh e Kumar 2020).

O Instituto de Engenheiros Elétricos e Eletrônicos (IEEE) define sistemas embarcados como dispositivos usados para controlar, monitorar ou auxiliar a operação de equipamentos, máquinas ou instalações. Complementarmente, outras definições os descrevem como sistemas baseados em microprocessadores com instruções fixas para controlar equipamentos eletromecânicos, ou ainda como sistemas que integram hardware e software embarcado como elementos centrais de operação (Ibrahim *et al.* 2015). Grimheden et al. definem um sistema embarcado como "aquele que possui hardware de computador com software incorporado como um de seus componentes mais importantes" (Ibrahim *et al.* 2015).

Em termos de arquitetura, os sistemas embarcados consistem em três partes principais: uma plataforma de hardware, um sistema operacional (embora nem sempre presente) e um software aplicativo (Ding e Wang 2020). O hardware normalmente inclui microprocessadores ou microcontroladores, interfaces de entrada/saída e diversas interfaces de dispositivos (Ding e Wang 2020). Alguns sistemas embarcados utilizam sistemas operacionais em tempo real (RTOS) para monitorar a execução do software, embora sistemas de menor escala possam operar sem um RTOS (Winarno *et al.* 2022; Ishak *et al.* 2019).

As principais características dos sistemas embarcados incluem especialização para tarefas específicas, compacidade, recursos de computação em tempo real e eficiência energética (Ai 2023). Muitos sistemas embarcados devem operar sob restrições de tempo real, o que significa que devem responder dentro de limites de tempo rigorosos por razões como segurança e usabilidade (Bourahla 2009; Kelemen *et al.* 2020). Eles são normalmente projetados para serem confiáveis, consumirem menos energia, serem leves e ter um custo menor em comparação com sistemas de computação de uso geral (Fan 2020).

2.3 Mineração de Dados

Mineração de dados pode ser definida como "o processo de extração de conhecimento e informações ocultas e potencialmente úteis de grandes quantidades de dados" (Huang *et al.* 2020). Representa uma evolução natural da tecnologia da informação que combina múltiplas

disciplinas para analisar e descobrir padrões não óbvios em dados ([Athauda et al. 2009](#)). Mais formalmente, pode ser descrita como "a extração não trivial de informações ocultas, novas e potencialmente úteis de dados" ([Shankar e Purusothaman 2010](#)) ou "o processo de seleção, exploração e modelagem de grandes bancos de dados para descobrir modelos e padrões que são desconhecidos a priori" ([Anderson et al. 2010](#)).

As principais características da mineração de dados incluem: a capacidade de lidar com grandes volumes de informações e de analisá-los de forma sistemática com o objetivo de identificar padrões recorrentes, extrair conhecimento implícito e gerar informações relevantes que possam ser interpretadas e exploradas de forma vantajosa pelos seres humanos ([Gullo 2015](#)). O que distingue a mineração de dados da recuperação regular de dados é que, na mineração de dados, as relações e associações entre os dados não são conhecidas de antemão, mas são descobertas ao longo do processo ([Anderson et al. 2010](#)).

A mineração de dados emprega uma variedade de técnicas de diferentes áreas, incluindo estatística, aprendizado de máquina, tecnologia de banco de dados, inteligência artificial, reconhecimento de padrões e visualização de dados ([Karahoca et al. 2012](#); [Athauda et al. 2009](#)). Essas técnicas podem ser amplamente categorizadas em dois grupos principais: preditivas e descritivas ([Mints 2017](#); [Gnanapriya et al. 2010](#)).

Tarefas de mineração de dados preditiva concentram-se na construção de modelos que podem prever comportamentos futuros ou valores desconhecidos com base em dados existentes ([Gnanapriya et al. 2010](#)). Essas tarefas incluem técnicas como classificação, regressão e detecção de anomalias ([Cheng et al. 2021](#)). A mineração de dados descritiva, por outro lado, visa encontrar padrões que descrevam os dados de forma compreensível para humanos ([Gnanapriya et al. 2010](#)), incluindo tarefas como agrupamento, mineração de regras de associação e descoberta de padrões ([Widaningsih et al. 2023](#)).

Em sua essência, a mineração de dados consiste em um processo estruturado para identificar padrões e conexões ocultas em grandes volumes de informação. A partir dessa análise, é possível transformar dados brutos em percepções práticas que apoiam e orientam decisões fundamentadas.

2.3.1 Análise Associativa

A análise associativa, também conhecida como análise de afinidade ou mineração de regras de associação, é uma técnica fundamental de mineração de dados focada em identificar relações frequentes entre elementos de dados (song Li *et al.* 2011). Essa abordagem visa revelar associações ocultas, particularmente em bancos de dados de transações, identificando itens que frequentemente aparecem juntos (Suryana e Yulianto 2019).

O conceito central da análise associativa é a regra de associação, que assume a forma de uma declaração de implicação "se antecedente, então consequente" (ou "se-então", ex: "se leite, então pão"), em que antecedente e consequente representam conjuntos de itens diferentes sem itens em comum (Buaton *et al.* 2019). Essas regras identificam tipos específicos de associações de dados (song Li *et al.* 2011).

Geralmente a regra de associação é considerada interessante se ela satisfaz um limiar mínimo de suporte (*support*). Essa métrica indica a frequência de ocorrência de um *item* em uma base de transações D e pode ser expressa com a seguinte equação:

$$support(A \Rightarrow B) = P(A \cup B) = \frac{freq(AB)}{|D|} = \frac{num. \text{ de transações que possuem os itens } A \text{ e } B}{num. \text{ transações em } D} \quad (2.2)$$

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

$$confidence(A \Rightarrow B) = \frac{P(A \cup B)}{P(A)} = \frac{freq(AB)}{freq(A)} = \frac{num. \text{ de transações que possuem os itens } A \text{ e } B}{num. \text{ de transações que possuem o item } A} \quad (2.3)$$

Uma terceira métrica chamada elevação (*lift*) mede a força da relação entre o antecedente e o consequente de uma regra de associação. Formalmente, quantifica quantas vezes os itens ocorrem juntos com mais frequência do que seria esperado se fossem estatisticamente independentes. Tal métrica é calculada por meio do suporte e confiança como segue:

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

Essa métrica sugere que a ocorrência do *itemset* A é independente da ocorrência do *itemset* B se $P(A \cup B) = P(A)$, do contrário, A e B são dependentes e correlacionados.

Diferente das métricas de *support* e *confidence* que têm suas faixas de valores de 0 a 1 (ou seja 0% a 100%), a métrica *lift* define que valores menores que 1 indicam que existe uma correlação negativa entre os itens A e B; se o valor for igual a 1, então A e B são independentes e não estão correlacionados; ou se o *lift* for maior que 1, existe uma correlação positiva entre A e B.

2.3.2 Regras de Associação: Algoritmos Fundamentais

O algoritmo Apriori foi proposto pela primeira vez por Rakesh Agrawal e seus colegas em 1993-1994 como uma solução para o problema da cesta de compras, estabelecendo-se como um dos métodos mais influentes na mineração de dados (Jabas 2011). O *Apriori* descobre regras de associação examinando relacionamentos entre itens em bancos de dados transacionais.

O conceito fundamental por trás do *Apriori* é a propriedade antimonótona: se um conjunto de itens não é frequente, então nenhum de seus superconjuntos também não pode ser frequente (Bashir *et al.* 2009). Essa propriedade permite que o algoritmo reduza eficientemente o espaço de busca ao minerar conjuntos de itens frequentes. A mineração de regras de associação usando *Apriori* envolve duas etapas principais: primeiro, encontrar todos os conjuntos de itens frequentes que atendem aos requisitos mínimos de suporte e, em seguida, gerar regras de associação fortes a partir desses conjuntos de itens que satisfaçam os limites mínimos de confiança.

Apriori emprega uma estratégia de busca por níveis, na qual k -itemsets (itemsets de tamanho k) são usados para explorar $(k+1)$ -itemsets. O algoritmo começa examinando o banco de dados para encontrar 1-itemsets frequentes (L_1), depois usa L_1 para encontrar L_2 (2-itemsets frequentes) e continua esse processo até que nenhum outro conjunto de itens frequentes seja encontrado. Cada etapa requer uma varredura completa do banco de dados, e o algoritmo gera conjuntos de itens candidatos unindo conjuntos frequentes da iteração anterior.

Para superar as limitações do *Apriori*, especialmente sua dependência de múltiplas varreduras do banco de dados e o alto custo de geração de candidatos, foram propostos algoritmos mais eficientes como o *FP-Growth* (*Frequent Pattern Growth*) e o *Eclat* (*Equivalence Class Clustering and bottom-up Lattice Traversal*). O *FP-Growth*, introduzido por Han et al. (Han *et al.* 2000), evita a geração explícita de candidatos ao construir uma estrutura compacta denominada *FP-Tree*, que armazena as transações em um formato prefixado comprimido. A mineração de padrões frequentes é então realizada diretamente nessa estrutura, utilizando uma abordagem de divisão recursiva e projeção de padrões condicionais. Isso reduz significativamente o custo computacional, tornando o *FP-Growth* particularmente eficaz em bases densas e volumosas.

Por sua vez, o *Eclat*, proposto por Zaki (Zaki 2000), adota uma estratégia baseada em interseções de conjuntos, utilizando uma representação vertical dos dados transacionais. Em vez de associar itens a transações, o algoritmo mantém listas de transações (TID-lists) para cada item, permitindo calcular rapidamente o suporte de *itemsets* por meio da interseção dessas listas. Essa abordagem reduz o número de acessos ao disco e facilita a exploração eficiente de subconjuntos, especialmente quando os dados podem ser mantidos em memória.

Ambos os algoritmos representam avanços significativos em relação ao *Apriori* no que tange ao desempenho, escalabilidade e eficiência em diferentes tipos de bases. Estudos comparativos mostram que tanto o *FP-Growth* quanto o *Eclat* frequentemente superam o *Apriori*, sobretudo em conjuntos de dados com alta densidade de itens ou grande volume de transações.

No entanto, vale destacar que esses algoritmos foram inicialmente concebidos para cenários centralizados, nos quais todos os dados estão disponíveis em um único repositório. Em aplicações modernas como a Internet das Coisas (IoT), onde os dados são coletados de forma distribuída por dispositivos heterogêneos e geograficamente dispersos, surgem desafios adicionais. Nesses casos, torna-se necessário adaptar ou redesenhar tais algoritmos para funcionarem de maneira distribuída ou federada, reduzindo a necessidade de centralização e promovendo a eficiência na borda da rede. Esse aspecto torna-se central na proposta de soluções mais escaláveis e aderentes às restrições computacionais e de comunicação típicas de ambientes IoT.

2.4 Resumo

Este capítulo apresentou os principais fundamentos teóricos que sustentam o desenvolvimento da proposta desta dissertação. Iniciou-se com uma explanação sobre *análise probabilística e as cadeias de Markov*, que fornece o arcabouço matemático para modelar eventos aleatórios e, em particular, o comportamento das personas por meio de transições de estado, base da geração de dados adotada neste trabalho. Em seguida, foram discutidos os *sistemas embarcados*, com ênfase em suas arquiteturas, restrições operacionais e papel central na computação de borda, reforçando sua adequação a ambientes IoT.

O capítulo também abordou a *mineração de dados*, especialmente a *análise associativa*, como ferramenta-chave para a descoberta de padrões implícitos em dados coletados localmente. Foram detalhadas as principais métricas utilizadas (suporte, confiança e *lift*), bem como os algoritmos fundamentais de mineração de regras de associação, como *Apriori*, *FP-Growth* e *Eclat*, evidenciando

suas estratégias de processamento e limitações em cenários centralizados.

Por fim, reforçou-se que esses conhecimentos técnicos, da modelagem probabilística à análise de padrões em dados, formam a base conceitual do método de mineração de regras de associação avaliado nesta dissertação. Eles permitem compreender os mecanismos envolvidos na coordenação de objetos autônomos e justificam as escolhas metodológicas adotadas nos capítulos subsequentes.

Capítulo 3

Trabalhos Correlatos

Este capítulo revisa criticamente a literatura relacionada ao uso de regras de associação em sistemas de Internet das Coisas (IoT) com ênfase na coordenação de objetos autônomos. Apresentam-se trabalhos que aplicam mineração de regras de associação em ambientes IoT (especialmente para análise de padrões e otimização) e propostas de arquiteturas distribuídas e baseadas em sistemas com múltiplos objetos. Os resultados de cada estudo são relacionados diretamente ao tema da coordenação de objetos em IoT, fazendo comparações críticas entre abordagens e apontando lacunas na literatura atual. Examinam-se, ainda, as formas de avaliação empregadas nesses estudos, sobre dados reais ou sintéticos, o que evidencia a lacuna de avaliação com verdade de referência abordada nesta dissertação.

3.1 Mineração de Regras de Associação em Ambientes IoT

Diversos trabalhos exploram a mineração de regras de associação aplicada a dados de IoT, sobretudo para descobrir padrões de uso, detectar anomalias ou suportar decisões preditivas. Autores como [Jang et al. \(2021\)](#) propõem o algoritmo *RFP-Growth*, que agrupa itens em regiões densas antes de aplicar o *FP-Growth*, revelando regras que o método tradicional não identifica no conjunto completo. Essa abordagem baseada na região demonstra que diferenças locais nos dados de dispositivos IoT podem revelar padrões úteis não capturados por métodos globais. De forma análoga, [Aceron et al. \(2021\)](#) focam na associação de dispositivos IoT a partir de tráfego de rede: eles desenvolveram uma versão modificada do *Apriori* para identificar pares de dispositivos (sensores/atuadores) que frequen-

temente interagem em aplicações IoT. Em experimentos com tráfego de IoT real, o método detectou corretamente relações de comunicação típicas, evidenciando aplicações de regras de associação no mapeamento de dependências entre objetos IoT.

Outras aplicações práticas incluem IoT em cidades inteligentes e domicílios inteligentes. Por exemplo, Kireev *et al.* (2018) aplicaram regras de associação para manutenção preditiva em casas inteligentes: mineração de regras de associação aplicada a dados de sensores processados em nuvem, eles classificaram e previram falhas de dispositivos domésticos através de padrões de uso. De modo similar, Ruiz *et al.* (2023) propuseram uma arquitetura de *Big Data*+IoT para análise de consumo energético em edifícios; por meio de mineração de regras de associação em larga escala, descobriram padrões ocultos de comportamento que sugerem estratégias para melhorar a eficiência energética. Ambos os casos exemplificam como regras de associação podem extrair padrões de comportamento colaborativo de sensores e atuadores em IoT, embora, no geral, tais estudos não entrem em detalhes sobre como objetos autônomos usam essas regras para coordenar ações.

Karabulut *et al.* (2025) introduziram um pipeline de mineração de regras de associação que integra dados dinâmicos de sensores com metadados estáticos em *knowledge graphs*. Eles propõem regras de associação "semânticas" que incorporam o contexto da rede IoT, usando um método de *autoencoder* para reduzir a dimensionalidade e o número de regras. O resultado é um conjunto mais conciso de regras de alta qualidade que cobrem dados de IoT, útil para monitoramento e decisões em sistemas IoT heterogêneos. Embora este trabalho não trate explicitamente de objetos, ele mostra que incorporar ontologias e lógica de conhecimento pode enriquecer a mineração de padrões em IoT, o que potencialmente poderia beneficiar objetos autônomos na compreensão do ambiente.

3.2 Arquiteturas Distribuídas e Múltiplos Objetos em MRA

Paralelamente, há esforços consideráveis voltados a arquiteturas distribuídas e ao uso de sistemas multiagentes (MAS) para executar a mineração de regras de associação. Ogunde *et al.* (2012) estudaram agentes móveis adaptativos para mineração distribuída de regras de associação: eles descrevem uma arquitetura nos quais percorrem bases de dados remotas para coletar frequências locais e adaptar-se dinamicamente ao ambiente. O objetivo é que agentes de software de mineração otimizem a comunicação e processem dados localmente, o que dialoga com requisitos de sistemas IoT distribuídos. De modo análogo, Hingane e Patil (2014) propuseram o *framework* MAD-ARM, no qual um agente móvel percorre vários sites de dados horizontalmente particionados, coletando a lista

global de itens frequentes (GFIL) e retornando ao servidor central. Essa abordagem evidencia como agentes podem reduzir o tráfego de comunicação ao transportar uma representação resumida dos itens frequentes, contribuindo para a eficiência em bancos de dados remotos (como em muitas arquiteturas IoT). [Saleem e Prakash Raja \(2017\)](#) expandiram essa linha de pesquisa com a criação do MADPARM, que complementa o MAD-ARM utilizando agentes móveis e tabelas de bits compactas para paralelizar o acesso e a mineração em cada site, aprimorando o desempenho na geração de conjuntos frequentes. Em resumo, essas propostas destacam como estruturas baseadas em multiagentes podem dividir e coordenar eficientemente a tarefa de mineração de regras de associação em sistemas distribuídos, contribuindo para a redução da latência e do tráfego de rede.

Complementando essas arquiteturas, em [Khedr *et al.* \(2019\)](#) cada *cluster head* (CH) é modelado como objeto que executa localmente algoritmos de MRA, trocando apenas estatísticas mínimas com outros CHs; isso reduz a transferência de dados entre nós sensores. Na sequência, em [Khedr *et al.* \(2020\)](#), eles propõem um esquema distribuído de MRA onde os nós sensores realizam processamento local (sem enviar dados brutos ao servidor central), trocando apenas resumos estatísticos aos CHs. Os resultados mostraram que essa divisão de tarefas prolonga a vida útil da rede IoT ao minimizar comunicações. Tais trabalhos exemplificam uma aplicação direta de MRA na coordenação de objetos físicos em IoT: cada objeto (*cluster head* ou nó) ajusta seu comportamento (coleta e transmissão de dados) a partir dos padrões locais descobertos, evidenciando a integração entre mineração de padrões e tomadas de decisão descentralizadas.

3.3 Integração entre Regras de Associação e Coordenação de Objetos

Os trabalhos mencionados frequentemente aplicam regras de associação para fins analíticos ou de eficiência de mineração, mas pouco tratam da coordenação explícita de objetos autônomos baseada nesses padrões. Em geral, notam-se duas linhas principais: (i) *extração de padrões para análise IoT*, e (ii) *propostas de infraestrutura distribuída multi-objeto para executar mineração de regras*. A conexão direta entre uma e outra, isto é, usar os padrões descobertos para guiar comportamentos colaborativos de objetos IoT, ainda é pouco explorada. Por exemplo [Jang *et al.* \(2021\)](#), [Aceron *et al.* \(2021\)](#) e [Kireev *et al.* \(2018\)](#) fornecem algoritmos (*RFP-Growth*, *Apriori* modificado) que identificam padrões locais ou preditivos em dados de IoT, mas não descrevem como objetos usariam

essas regras em tempo de execução. Por outro lado, estruturas como MAD-ARM (Hingane e Patil 2014) e o esquema distribuído de Khedr *et al.* (2019) mostram como agentes realizam mineração de forma cooperativa, mas o foco é na eficiência da mineração e não na coordenação de tarefas de objetos aplicando os padrões.

Um exemplo relacionado à coordenação de objetos em IoT é a arquitetura fuzzy para iluminação urbana inteligente de Kouah *et al.* (2024). Embora não use regras de associação, seu sistema de múltiplos objetos IoT para controle de postes de luz destaca diversos aspectos de coordenação, objetos especializados assumem controle local, negociação e lógica *fuzzy* para equilibrar eficiência energética e segurança urbana. Esse trabalho ilustra como objetos IoT podem cooperar para atingir metas comuns, servindo como referência para como regras de associação poderiam ser incorporadas (por exemplo, *se padrões de movimento forem detectados nos sensores X e Y, então deve-se aumentar a intensidade das luzes A e B*).

O pouco acoplamento entre MRA e coordenação aponta a uma lacuna na literatura: faltam estudos que implementem diretamente regras de associação extraídas em heurísticas de coordenação de múltiplos objetos. Por exemplo, Cao *et al.* (2009) já postulavam que a sinergia entre objetos e mineração de dados, é um campo promissor, mas na prática há escassez de casos em que objetos autônomos usam padrões descobertos para autorregular suas interações. A grande maioria dos trabalhos de IoT com MRA concentra-se em análises offline (padrões de uso, recomendações, etc.), enquanto a coordenação autônoma permanece pouco abordada. Em resumo, embora existam muitos estudos sobre MRA em IoT e sobre sistemas com múltiplos objetos para MRA distribuída, poucos convergem para um cenário em que objetos comunicam e ajustam seu comportamento colaborativamente com base em regras de associação descobertas.

3.4 DARE

O DARE (Decentralized Association Rules Extraction) (Alencar et al., 2020) é o antecedente direto da estratégia aqui avaliada: distribui a extração de regras entre dispositivos IoT e utiliza mudanças de estado para construir padrões locais, comparando as regras obtidas com uma alternativa centralizada, e reportando concordância de 99,75% entre ambas.

Essa concordância, contudo, responde se duas estratégias de mineração produzem resultados semelhantes; ela não estabelece se as relações recuperadas correspondem às coordenações verdadeiras do ambiente. Este trabalho não reivindica o mecanismo de mineração como novo: sua novidade

está em introduzir uma referência externa e controlada que permite distinguir concordância entre mineradores de recuperação das coordenações verdadeiras. Com isso, torna-se possível medir falsos positivos, cobertura e F1 e, sobretudo, observar limitações representacionais do encadeamento que uma comparação entre o modelo centralizado e o descentralizado não evidencia.

3.5 Avaliação: Dados Reais e *Ground Truth*

Boa parte dos trabalhos que aplicam mineração de regras de associação em ambientes inteligentes é avaliada sobre conjuntos de dados reais de uso, como os datasets de casas inteligentes CASAS [Cook et al. \(2013\)](#). A produção desses conjuntos reais é, além disso, trabalhosa, e a própria rotulagem do *ground truth* das atividades é um desafio reconhecido na área [Köckemann et al. \(2020\)](#); [Bouchabou et al. \(2021\)](#). Nesse tipo de avaliação, mede-se o acordo entre as regras extraídas e os dados observados; contudo, tais conjuntos não trazem um gabarito rotulado das coordenações verdadeiras entre dispositivos. Por isso, é possível avaliar a precisão ou a concordância das regras, mas não medir diretamente a cobertura, o quanto da coordenação real o método recupera.

Uma alternativa é gerar dados sintéticos com estrutura conhecida, por meio de simuladores de ambientes inteligentes, como o OpenSHS [Alshammari et al. \(2017\)](#), a partir do qual se produziram datasets rotulados de atividades diárias [Alshammari et al. \(2018\)](#). Tais geradores, porém, são empregados sobretudo para reconhecimento de atividades, e não para avaliar quão bem a mineração de regras de coordenação recupera as relações entre dispositivos.

Este trabalho situa-se nessa lacuna: emprega personas sintéticas com estrutura de coordenação conhecida para medir precisão e cobertura das regras de coordenação mineradas, uma avaliação que os conjuntos de dados reais não permitem de forma direta.

3.6 Síntese Comparativa

A Tabela 3.1 sintetiza as diferenças entre os trabalhos discutidos quanto à técnica empregada, à arquitetura, ao uso das regras na coordenação, aos dados de avaliação e à disponibilidade de uma verdade de referência das coordenações. Três padrões emergem. Primeiro, a maioria dos trabalhos concentra-se na eficiência ou na aplicação da mineração, sem levar as regras extraídas à coordenação efetiva dos objetos; apenas [Kouah et al. \(2024\)](#) e o DARE atuam sobre o comportamento dos dispositivos, e o primeiro não usa regras de associação. Segundo, as abordagens descentralizadas

que aplicam regras à coordenação reduzem-se, na prática, ao DARE. Terceiro, e é a lacuna que motiva esta dissertação, nenhum dos trabalhos dispõe de uma verdade de referência das coordenações: os que usam dados reais só podem medir concordância ou acurácia de classificação, e os que usam dados sintéticos rotulam atividades, não relações entre dispositivos.

Tabela 3.1: Comparação dos trabalhos correlatos

Trabalho	Abordagem / técnica	Arquitetura	Regras aplicadas à coordenação	Dados	Verdade de referência das coordenações
Jang et al. (2021)	<i>RFP-Growth</i> (clusterização por densidade + <i>FP-Growth</i> por região)	Centralizada	Não	Espaciais de IoT (com posição)	Não
Aceron et al. (2021)	<i>Apriori</i> modificado (descobre pares sensor atuador)	Centralizada na borda (roteador)	Não (usa os pares para reencaminhar mensagens)	Tráfego de rede	Não
Kireev et al. (2018)	ARM para manutenção preditiva (prever necessidade de reparo)	Centralizada (nuvem)	Não	Casas inteligentes	Não
Ruiz et al. (2023)	ARM em larga escala (<i>Big Data</i>)	Centralizada (<i>big data</i>)	Não	Reais, consumo de edifício de escritórios (Bucareste)	Não
Karabulut et al. (2025)	ARM semântica (incorpora conhecimento/ontologias)	Centralizada	Não	IoT	Não
MAD-ARM (2014)	ARM distribuída por agentes móveis	Distribuída	Não	Bases distribuídas	Não
Khedr et al. (2019)	ARM distribuída com preservação de privacidade (bases distribuídas vertical/horizontalmente)	Distribuída (baseada em <i>clusters</i>)	Não	WSN / IoT	Não
Kouah et al. (2024)	Sistema multiagente com lógica <i>fuzzy</i>	Distribuída	Sim (sem regras de associação)	Iluminação urbana	Não
Cao et al. (2009)	<i>Agent mining</i> (posicionamento conceitual)	—	Proposta conceitual	—	—
CASAS (2013)	Conjuntos de dados de casas inteligentes	—	—	Reais	Não (apenas rótulos de atividade)

Continua na próxima página.

Tabela 3.1 — continuação

Trabalho	Abordagem / técnica	Arquitetura	Regras aplicadas à coordenação	Dados	Verdade de referência das coordenações
OpenSHS (2017) / SIMADL (2018)	Simulador / conjunto de dados sintético	—	Não	Sintéticos	Parcial (atividades, não coordenações)
DARE (2020)	ARM embarcada por transições de estado	Descentralizada	Sim	Reais	Não (concordância centralizado $\times$ descentralizado)
Este trabalho	Avaliação do DARE sob referência controlada	Descentralizada	Sim	Sintéticos (personas por cadeias de Markov)	Sim (coordenações conhecidas)

3.7 Resumo

Comparando os trabalhos, observa-se que os métodos orientados a dados tendem a explorar variações e aplicações de algoritmos clássicos sobretudo *Apriori* e *FP-Growth* (e derivados como o *Apriori* modificado ou o *FP-Growth* regional) sendo pouco exploradas alternativas como o Eclat, enquanto os métodos orientados a objetos desenvolvem arquiteturas distribuídas para melhorar a mineração em si. Personalizações do processo de mineração, agrupamento regional ou incorporação de conhecimento, geram regras mais relevantes ou concisas em contextos IoT [Jang et al. \(2021\)](#); [Karabulut et al. \(2025\)](#).

Persiste, contudo, um fraco acoplamento entre a mineração de regras e a coordenação efetiva: são escassos os estudos em que objetos autônomos usam os padrões descobertos para autorregular suas interações, predominando análises offline (padrões de uso, recomendações).

Soma-se a isso uma lacuna de avaliação: os trabalhos costumam ser avaliados sobre dados reais que não trazem um gabarito das coordenações verdadeiras [Cook et al. \(2013\)](#); [Köckemann et al. \(2020\)](#), ou sobre dados sintéticos voltados ao reconhecimento de atividades [Alshammari et al. \(2017, 2018\)](#), de modo que a capacidade de recuperar as relações reais entre dispositivos, em especial a cobertura, não é medida diretamente. É nessa lacuna que se insere esta dissertação, que avalia a mineração de regras de coordenação sobre personas sintéticas de estrutura conhecida, medindo precisão e cobertura.

Capítulo 4

Método Proposto

4.1 Visão Geral do Método Proposto

O método adotado neste trabalho coordena dispositivos autônomos por meio da mineração de regras de associação entre suas ações. Cada dispositivo observa localmente seu próprio histórico de uso, resume-o em um padrão de ações e, comparando-o com o padrão de dispositivos vizinhos, extrai regras da forma antecedente $\Rightarrow$ consequente, em que o antecedente é uma ação do dispositivo local e o consequente, uma ação de um vizinho, usando as métricas de suporte, confiança e *lift*. As regras extraídas permitem sincronizar o estado de um dispositivo em função de uma ação observada em outro, sem depender de um servidor central.

Neste trabalho o método é implementado em um simulador de ambiente residencial inteligente e avaliado sob uma verdade conhecida. As subseções seguintes descrevem o modelo de dados e as etapas da mineração (Seções 4.2–4.5), o mecanismo de coordenação por encadeamento (Seção 4.6) e o escopo por cômodo com a combinação de vários moradores (Seção 4.7).

4.2 Estados e Ações do Dispositivo

Cada dispositivo controla um elemento do ambiente (por exemplo, uma lâmpada) e opera com base em dois conjuntos principais:

- $S = \{s_1, s_2, \dots, s_i\}$: conjunto finito de estados possíveis (por exemplo, "Lâmpada Ligada", "Lâmpada Desligada").

- $A = \{a_1, a_2, \dots, a_i\}$: conjunto de ações que provocam a mudança de estado (por exemplo, "ligar", "desligar").

No simulador, os estímulos que provocam ações provêm da interação (do usuário ou da geração automática de eventos, Seção 4.8), e cada estímulo é associado à ação correspondente, que atualiza o estado do dispositivo e registra o evento.

4.3 Registro de Eventos e Geração de Padrões

Diferentemente de abordagens que realizam coletas periódicas, o método foca apenas nas mudanças de estado, otimizando o uso de recursos ao registrar apenas eventos relevantes. Ao invés de registrar continuamente o estado do dispositivo ao longo do tempo, o sistema armazena somente os momentos em que uma ação foi tomada, simplificando o histórico para eventos relevantes.

Esses eventos são organizados em uma matriz de contadores $M_{ij} = A \times T$, onde A é o conjunto de ações e T é o conjunto de *slots* temporais. Cada elemento c_{xy} representa quantas vezes a ação a_x ocorreu no *slot* t_y . Após a coleta, a matriz passa por uma transformação logarítmica para suavizar a influência de valores altos e reduzir ruídos:

$$c'_{xy} = \begin{cases} \log_{|A|}(c_{xy}), & \text{se } \log_{|A|}(c_{xy}) \geq 1, \\ 0, & \text{caso contrário,} \end{cases} \quad \text{para } 1 \leq x \leq |A|, 1 \leq y \leq |T|, \forall c_{xy} \in M_{ij}.$$

Tal condição, representada no *slot* 5 na Tabela 4.1, evita que os contadores assumam valores negativos durante as transformações posteriores.

Além de comprimir a escala das contagens, a condição $\log_{|A|}(c_{xy}) \geq 1$ elimina ocorrências abaixo do limiar implícito $c_{xy} \geq |A|$; no caso binário ligar/desligar, uma ocorrência isolada é descartada. Como o logaritmo é estritamente crescente, a transformação não altera a ordem relativa entre contagens que permanecem acima do limiar. Assim, quando o passo seguinte escolhe apenas a ação dominante, seu papel principal é filtrar ocorrências esparsas e reduzir a influência de eventos pouco recorrentes, e não redefinir qual ação possui a maior contagem.

A partir da matriz transformada M'_{ij} , é gerado um padrão local P indicando a ação dominante em cada *slot*. Caso duas ações tenham a mesma frequência máxima, nenhuma é registrada para aquele *slot*. A Tabela 4.1 ilustra esse processo:

Tabela 4.1: Base de dados embarcada, base transformada e padrão de ações extraído

SLOTS (T)	1	2	3	4	5	6
Matriz original M_{ij}						
ON	2	4	25	0	1	4
OFF	5	4	17	2	0	5
Após transformação logarítmica M'_{ij}						
ON	1	2	4.64	0	0	2
OFF	2.32	2	4.08	1	0	2.32
Padrão gerado (P)						
	OFF	–	ON	OFF	–	OFF

É possível extrair o padrão P da seguinte forma: seja $C_y = \{c_{1y}, c_{2y}, \dots, c_{iy}\}$ o conjunto contendo todos os contadores da coluna y da matriz M_{ij} , e uma função $max_action(C_y, A)$ que retorna a ação $a \in A$ correspondente ao maior valor em C_y (ou *nil* caso haja mais de um contador com o mesmo valor máximo); então, é possível criar um conjunto

$$P = \{(p_1, \dots, p_j) \in A^j \mid p_y = max_action(C_y, A), 1 \leq y \leq |T|\},$$

onde cada p_y é a ação dominante no *slot* y . Esta extração baseia-se na análise probabilística (Seção 2.1) para definir o padrão de ações do dispositivo e, em caso de probabilidades iguais em um mesmo *slot*, nenhuma ação é incluída em P (como nos *slots* 2 e 5 da Tabela 4.1).

Essa organização reduz significativamente a quantidade de dados a ser pré-processada para a análise associativa.

4.4 Formação da Base de Transações

A cada intervalo fixo, denominado *checkpoint*, os padrões locais P_l dos dispositivos trocados entre os pares. O dispositivo que executa a mineração escolhe outro dispositivo como alvo e coleta seu padrão remoto P_r . A partir disso, é formada a base de transações

$$D = \{(p_{l1}, p_{r1}), \dots, (p_{lj}, p_{rj})\},$$

onde cada par representa ações simultâneas nos mesmos *slots*.

Esse processo ocorre de forma iterativa e distribuída, gerando diversas bases de transações (D_{12} , D_{13} , ...) a partir de diferentes pares de dispositivos. A Tabela 4.2 exemplifica como um Dispositivo 1 combina seu padrão de ações (P_1) com outros padrões (P_1 e P_2) para formar as bases de transações (D_{12} e D_{13}):

Tabela 4.2: Formação de bases de transações a partir de padrões de múltiplos dispositivos

SLOTS		1	2	3	4	5	6	7	8	9	10
Padrões	P1	OFF	OFF	–	ON	OFF	OFF	ON	–	–	OFF
	P2	–	OPEN	OPEN	–	OPEN	CLOSE	CLOSE	CLOSE	–	OPEN
	P3	HIGH	–	LOW	LOW	–	HIGH	LOW	–	–	–
Base de transações	D12	OFF	OFF,OPEN	OPEN	ON	OFF,OPEN	OFF,CLOSE	ON,CLOSE	CLOSE	–	OFF,OPEN
	D13	OFF,HIGH	OFF	LOW	ON,LOW	OFF	OFF,HIGH	ON,LOW	–	–	OFF

A construção da base de transações é realizada por meio da combinação de padrões em pares (por exemplo, $D_{12} = P_1P_2$ e $D_{13} = P_1P_3$, o que possibilita a geração de bases com diferentes tamanhos, como exemplificado nos slots 8 e 9 da Tabela 4.2. No contexto da análise associativa, torna-se inviável comparar diretamente as regras derivadas dessas bases, uma vez que métricas como *suport* (Equação(2.2)) e *lift* (Equação(2.4)) dependem diretamente do número total de transações em cada base $|D_{12}|$ e $|D_{13}|$.

Para superar essa limitação, adotou-se uma adaptação nas métricas de *suport* e *lift*, de modo a torná-las compatíveis com os objetivos deste estudo. A modificação consiste em considerar que todas as bases formadas por combinações de dois padrões tenham seu tamanho padronizado com base na quantidade máxima de *slots* possíveis de preenchimento em P , representado por $|T|$. Essa adequação permite realizar comparações equitativas entre as regras extraídas, mesmo quando derivadas de bases de tamanhos distintos, sem a necessidade de integrar simultaneamente todos os padrões na base de transações.

4.5 Mineração de Regras Associativas

O processo de extração de associações ocorre individualmente em cada dispositivo, e as regras extraídas são específicas ao seu contexto e histórico: o antecedente é uma ação do dispositivo que executa a mineração e o consequente, uma ação de um dispositivo-alvo. A mineração toma como alvos os dispositivos do mesmo ambiente.

Uma vez que o dispositivo obtém a lista de alvos, é iniciado um processo iterativo, como ilustra a Figura 4.1

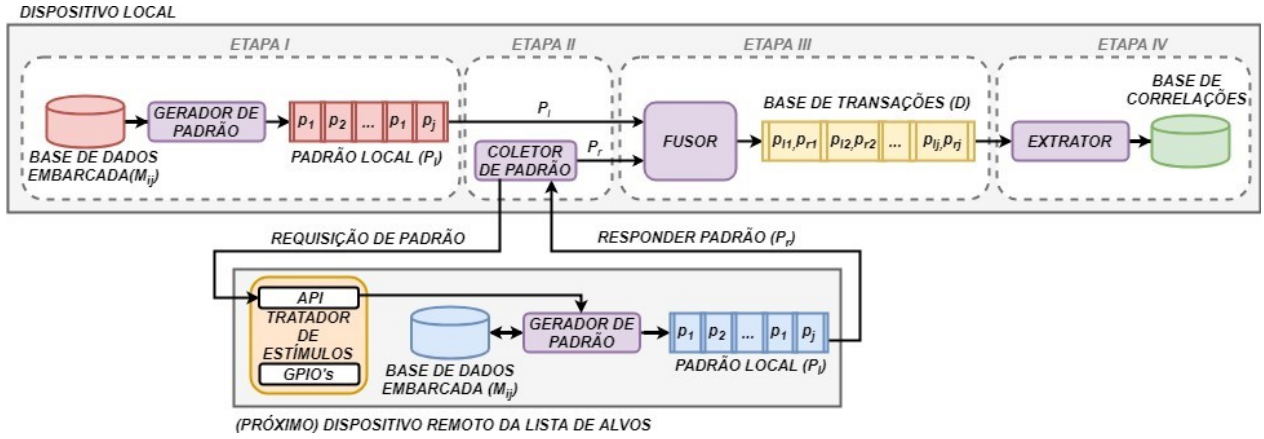


Figura 4.1: Etapas do método (Alencar 2019)

- Etapa I:** O dispositivo identifica seu próprio padrão de ações (P_l : Padrão Local) a partir da base de dados embarcada (M_{ij});
- Etapa II:** O componente *Coletor de Padrões* envia uma solicitação diretamente ao endereço MAC do próximo dispositivo alvo conhecido, solicitando seu padrão de ações. O destino recebe a solicitação (por meio de sua própria API), executa a **Etapa I** em si mesmo e, em seguida, responde a origem seu padrão de ação, que é representado como P_r (Padrão Remoto) na Figura 4.1.
- 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 4.4
- Etapa IV:** Finalmente, é realizada a análise associativa na base de transações D buscando identificar as associações mais fortes para cada ação $a \in A$ do dispositivo local em relação as ações do dispositivo alvo. Então as associações mais relevantes são comparadas com as obtidas anteriormente que estão armazenadas na *Base de Transações*. 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 ciclo de mineração é repetido a cada *checkpoint*, permitindo a adaptação contínua das regras ao perfil de uso do ambiente.

A mineração produz inicialmente as regras mineradas: todas as relações $A \Rightarrow B$ identificadas a partir dos padrões de ocorrência conjunta que satisfazem os limiares de suporte, confiança e *lift*. Assim, para um mesmo antecedente A podem existir várias candidatas, $A \Rightarrow B$, $A \Rightarrow C$, $A \Rightarrow D$, cada uma com métricas distintas.

Entretanto, nem todas são utilizadas pelo mecanismo de coordenação. Na configuração original aqui avaliada, para cada antecedente é mantida no máximo uma regra, denominada regra retida. A seleção escolhe a de melhor pontuação, priorizando *lift*, seguido de confiança e suporte. As regras retidas constituem, portanto, um subconjunto das mineradas e correspondem às arestas efetivamente utilizadas pelo mecanismo de coordenação.

4.6 Coordenação por Encadeamento

Como cada antecedente gera uma única regra (Seção 4.5), a coordenação de um para muitos não é obtida por várias regras diretas a partir de um mesmo dispositivo, e sim por encadeamento. Ao ser disparada a ação A , aplica-se a regra $A \Rightarrow B$; se o dispositivo B também for antecedente de uma regra ($B \Rightarrow C$), o disparo propaga-se a C , e assim sucessivamente, formando uma cadeia $A \rightarrow B \rightarrow C \rightarrow \dots$. Por exemplo, ao ligar a lâmpada da sala, a regra aciona a TV; como a TV, por sua vez, é antecedente de outra regra, o acionamento segue até o ar-condicionado. Dessa forma, uma única ação inicial sincroniza um conjunto de dispositivos relacionados.

Para evitar laços, o percurso registra os nós já visitados e interrompe a cadeia ao detectar um ciclo, em particular, quando dois dispositivos se apontam mutuamente (um par mútuo $A \leftrightarrow B$). A interrupção é necessária à consistência (evita disparos infinitos), mas tem uma consequência importante: os dispositivos situados além do ponto de corte não são alcançados pela cadeia. Assim, o alcance de um disparo depende da topologia das regras mineradas, e é justamente essa fração alcançada que a métrica de cobertura quantifica, o que explica por que a cobertura pode ser inferior a 100%.

O conjunto de regras retidas pode ser representado por um grafo dirigido $G = (V, E)$, em que cada vértice corresponde a uma ação e cada aresta a uma regra antecedente $\Rightarrow$ consequente. Como se retém no máximo uma regra por antecedente, o grafo tem grau de saída ≤ 1 . Consequentemente, a partir de uma ação inicial o conjunto alcançável forma uma trajetória que termina em um vértice sem sucessor ou em um ciclo: o mecanismo não produz ramificação direta a partir de um mesmo antecedente.

Segue daí que, se o grafo de referência contém um vértice com múltiplos consequentes verdadeiros, a estratégia de uma única regra por antecedente não consegue representar simultaneamente todas essas relações sem depender de arestas intermediárias adicionais. Trata-se de uma limitação representacional, e não de uma falha de implementação.

4.7 Escopo por Cômodo e Combinação de Moradores

A coordenação é buscada dentro de cada cômodo (ambiente): a mineração considera apenas os dispositivos do mesmo ambiente, não misturando dispositivos de cômodos distintos. Assim, cada cômodo é minerado de forma independente, o que mantém a coordenação fisicamente plausível e reflete a organização por vizinhança do ambiente.

Para representar uma casa com vários moradores, distinguem-se dois tipos de cômodo:

- compartilhados (sala, cozinha): existe um único por casa, e o comportamento é a combinação dos moradores que ali atuam;
- privados (quarto, escritório): cada morador tem o seu, minerado separadamente; um casal (ou um escritório de uso comum) é modelado como uma instância compartilhada explícita.

A combinação de vários moradores sobre um dispositivo compartilhado segue duas regras: o dispositivo liga se qualquer morador o liga (união das probabilidades de acionamento) e desliga apenas quando nenhum morador o mantém em uso (mínimo das probabilidades de desacionamento). Para moradores indexados por i , as probabilidades combinadas são dadas por

$$p_{\text{ligar}} = 1 - \prod_i (1 - p_{\text{acionar}, i}),$$

$$p_{\text{desligar}} = \min_i (p_{\text{desacionar}, i})$$

Dessa forma, um dispositivo permanece ativo enquanto ao menos um morador o utiliza, evitando modelagens irreais em espaços de convivência.

4.8 Modelagem de Personas por Cadeias de Markov

Na ausência de habitantes reais na avaliação, o comportamento de uso é modelado por meio de personas sintéticas, cuja estrutura de coordenação é previamente definida e conhecida. Cada persona

é descrita, por dispositivo, como uma sequência de faixas horárias; a cada faixa associam-se duas probabilidades de transição: $p_{acionar}$, probabilidade de o dispositivo ligar (passar ao estado ativo) naquela faixa, caso esteja desligado, e $p_{desacionar}$, probabilidade de desligar, caso esteja ligado.

Cada dispositivo evolui, assim, como uma cadeia de Markov não homogênea no tempo, em que a probabilidade de transição depende (i) da faixa horária e (ii) do estado atual do dispositivo. A não homogeneidade decorre de as probabilidades variarem ao longo do dia (por exemplo, nulas entre 0h e 6h, quando não há atividade). Cada dia é dividido em 96 *slots* de 15 minutos e há um conjunto de *slots* por dia da semana, totalizando $|T| = 7 \times 96 = 672$ *slots* semanais; dentro de cada dia, $slot = hora \times 4 + \lfloor minuto/15 \rfloor$.

A avaliação foi conduzida em um simulador de ambiente residencial inteligente desenvolvido como aplicação web. Ele reproduz os cômodos de uma residência, cada um com seu conjunto de dispositivos, e oferece dois modos: (i) interação manual, em que o usuário aciona os dispositivos na planta do ambiente, e (ii) geração sintética, em que uma semana completa de interações é produzida automaticamente a partir de uma persona. É o segundo modo que sustenta os experimentos.

Uma persona descreve o comportamento de um morador como uma distribuição de probabilidades por faixa horária. Para cada dispositivo, define-se uma lista de faixas; cada faixa contém o intervalo de tempo e duas probabilidades de transição: $p_{acionar}$, a probabilidade de ligar o dispositivo naquele intervalo caso esteja desligado, e $p_{desacionar}$, a probabilidade de desligar caso esteja ligado. A Listagem 4.1 exemplifica uma persona (adulto que trabalha fora, cômodo sala).

```

1  {
2    "profile": "Adulto que trabalha fora",
3    "environment": "sala",
4    "persona": {
5      "Lampara": [
6        {"inicio": "00:00", "fim": "19:00",
7         "p_acionar": 0.00, "p_desacionar": 0.40},
8        {"inicio": "19:00", "fim": "23:00",
9         "p_acionar": 0.82, "p_desacionar": 0.03},
10       {"inicio": "23:00", "fim": "24:00",
11        "p_acionar": 0.00, "p_desacionar": 0.85}
12     ],
13     "TV": [
14       {"inicio": "00:00", "fim": "19:00",
15        "p_acionar": 0.00, "p_desacionar": 0.40},
16       {"inicio": "19:00", "fim": "23:00",
17        "p_acionar": 0.80, "p_desacionar": 0.04},
18       {"inicio": "23:00", "fim": "24:00",
19        "p_acionar": 0.00, "p_desacionar": 0.85}
20     ],
21     "AireAcond": [
22       {"inicio": "00:00", "fim": "19:00",
23        "p_acionar": 0.00, "p_desacionar": 0.30},
24       {"inicio": "19:00", "fim": "23:00",
25        "p_acionar": 0.72, "p_desacionar": 0.03},
26       {"inicio": "23:00", "fim": "24:00",
27        "p_acionar": 0.00, "p_desacionar": 0.80}
28     ]
29   }
30 }

```

Listing 4.1: Exemplo de persona: ao chegar em casa (≈ 19 h), lâmpada, TV e ar-condicionado tendem a ser ligados na mesma faixa; ao dormir (≈ 23 h), tendem a ser desligados. As probabilidades nulas entre 0 h e 19 h representam ausência de atividade.

Atribuição das probabilidades. Os valores são atribuídos por arquétipo de morador (trabalhador, home office, estudante, aposentado, dona de casa, criança), refletindo a rotina esperada: p_{acionar} alto nas faixas de presença e uso, nulo nas faixas de ausência ou sono, e $p_{\text{desacionar}}$ alto ao final de cada rotina. É a coincidência de faixas com p_{acionar} elevado entre dispositivos distintos que define a estrutura de coordenação da persona, e é essa estrutura que a mineração deve recuperar.

Geração por Monte Carlo. A geração percorre a semana *slot* a *slot* ($t = 1, \dots, |T|$), partindo do estado inativo. Em cada *slot*, sorteia-se $u \sim U(0, 1)$ e aplica-se:

- se d está inativo e $u < p_{\text{acionar}}(d, t)$, passa a ativo e registra-se a transição “ligar”;
- se d está ativo e $u < p_{\text{desacionar}}(d, t)$, passa a inativo e registra-se a transição “desligar”;
- caso contrário, o estado permanece e nada é registrado (Seção 4.3).

Esse procedimento constitui, portanto, uma simulação de Monte Carlo: cada slot é um ensaio de Bernoulli condicionado ao estado atual e à faixa horária. Como as transições de dispositivos que compartilham a mesma faixa são sorteadas de forma independente, a coocorrência emerge estocasticamente, não é imposta, reproduzindo a variabilidade de um comportamento real: em algumas noites os dispositivos são acionados no mesmo slot, em outras não.

A semana é simulada $R = 14$ vezes de forma independente, acumulando as contagens na mesma matriz, de modo a estabilizar as frequências antes da mineração. Isso corresponde à fase de treinamento (Seção 4.10).

Reprodutibilidade. Toda a aleatoriedade provém de um gerador pseudoaleatório determinístico (mulberry32) [Ettinger \(2017\)](#) inicializado por uma semente. Fixada a semente, a sequência de sorteios, e portanto os dados gerados e as regras mineradas, é integralmente reproduzível. Nos experimentos, a repetição r ($r = 0, \dots, 99$) usa a semente $42 + r$.

A coordenação emerge das transições sincronizadas: quando vários dispositivos têm p_{acionar} (ou $p_{\text{desacionar}}$) alto na mesma faixa, tendem a mudar de estado juntos (por exemplo, ao chegar em casa às 19h ligam-se lâmpada, TV e ar-condicionado). São essas transições sincronizadas que constituem as coordenações a serem mineradas. Nos cômodos compartilhados de uma casa, o comportamento resulta da combinação dos moradores presentes (Seção 4.7).

4.9 Definição Operacional do *Ground Truth* e das Métricas

A verdade de referência é obtida da especificação da persona, e não dos dados gerados nem das regras mineradas. Em vez de sortear transições, calcula-se o valor esperado delas pela própria recorrência de Markov. Seja $a_d(t) = P(\text{dispositivo } d \text{ estar ativo no slot } t)$, com $a_d(0) = 0$. Para cada *slot*:

$$e_{\text{on}}(d, t) = [1 - a_d(t)] \cdot p_{\text{acionar}}(d, t),$$

$$e_{\text{off}}(d, t) = a_d(t) \cdot p_{\text{desacionar}}(d, t),$$

$$a_d(t+1) = a_d(t)[1 - p_{\text{desacionar}}(d, t)] + [1 - a_d(t)]p_{\text{acionar}}(d, t).$$

O padrão esperado P_d^* atribui a cada *slot* a transição de maior esperança: “ligar” se $e_{\text{on}} > e_{\text{off}}$, “desligar” caso contrário; e *nil* quando ambas são desprezíveis ($\max\{e_{\text{on}}, e_{\text{off}}\} < \varepsilon$) ou praticamente iguais ($|e_{\text{on}} - e_{\text{off}}| < \varepsilon$), com $\varepsilon = 10^{-3}$. O cálculo é determinístico: descreve o comportamento que a persona induz em média, sem a variabilidade de uma realização particular.

Sobre os padrões esperados de todos os dispositivos de um cômodo r , enumeram-se todas as coordenações que satisfazem os limiares:

$$C_r^* = \{(a_i, a_j) \mid a_i \in P_u^*, a_j \in P_v^*, u \neq v, \text{ e os limiares são satisfeitos}\}.$$

Diferentemente da mineração, aqui não se aplica a regra de um consequente por antecedente: C_r^* contém o conjunto completo das coordenações que a persona implica, servindo de gabarito. Como C_r^* deriva da configuração geradora, e nunca dos dados estocásticos nem das regras retidas, constitui uma referência externa ao processo de extração, evitando avaliar o minerador contra uma referência produzida por ele mesmo.

Por serem aplicados os mesmos limiares, $|C_r^*|$ depende do nível de confiança adotado; a referência corresponde às coordenações detectáveis naquele regime de limiares. As métricas devem, portanto, ser lidas como recuperação relativa a esse conjunto.

Conjunto recuperado e métricas. O conjunto recuperado $\hat{C}_r$ contém as coordenações efetivamente alcançáveis pelo mecanismo a partir das regras retidas. Se $G_r = (V_r, E_r)$ é o grafo de regras retidas no cômodo r , então

$$\hat{C}_r = \{(u, v) \mid v \text{ é alcançável a partir de } u \text{ em } G_r\}.$$

Operacionalmente, isso corresponde a, para cada gatilho, percorrer a cadeia de regras retidas, respeitando a detecção de ciclos, e registrar o conjunto de coordenações alcançadas. Assim:

$$\begin{aligned} TP &= |C_r^* \cap \hat{C}_r|, \\ FP &= |\hat{C}_r \setminus C_r^*|, \\ FN &= |C_r^* \setminus \hat{C}_r|, \\ P &= \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F_1 = \frac{2PR}{P + R}. \end{aligned}$$

Essa definição delimita também a semântica adotada: associação estatística não é causalidade; uma relação minerada pode ter suporte, confiança e *lift* elevados e ainda assim ser um falso positivo de coordenação se não pertencer a C_r^* .

Registram-se ainda duas medidas de diagnóstico da topologia, o número de ciclos cortados e de pares mútuos ($A \leftrightarrow B$), que ajudam a explicar a cobertura incompleta.

4.10 Protocolo de Avaliação em Duas Fases

A avaliação de cada cenário segue duas fases:

1. Treinamento: a partir da persona, gera-se aleatoriamente uma semana de transições por cadeia de Markov, acumulada por R repetições, sem minerar regras (equivalente a "coletar os dados de uso").
2. Teste: sobre os dados gerados, executa-se a mineração de regras e, em seguida, calculam-se as métricas contra o *ground truth*.

Essa separação garante que as regras sejam avaliadas sobre dados gerados (estocásticos), e não sobre o padrão esperado usado como gabarito.

4.11 Reprodutibilidade e Desenho Experimental

Para assegurar a reprodutibilidade, a geração usa um gerador de números pseudoaleatórios com semente determinística; fixada a semente, os resultados são replicáveis. Cada cenário é executado por $N = 100$ repetições independentes (cada uma regenerando os dados com uma semente distinta), reportando-se a média $\pm$ intervalo de confiança de 95%. Uma curva de convergência (média acumulada de F1 em função do número de execuções) justifica a suficiência de N.

Realiza-se uma análise de sensibilidade ao parâmetro de confiança (0,60; 0,75; 0,90), mantendo fixos suporte = 1% e lift = 1,1. O conjunto experimental compreende 5 casas $\times$ 4 moradores = 20 personas, das quais derivam 29 instâncias físicas de cômodo (compartilhados combinam os moradores; privados são separados por morador); com os três níveis de confiança, totalizam-se 87 pontos experimentais. As métricas resultantes (precisão, recall e F1, além dos diagnósticos de ciclos e pares mútuos) são reportadas no Capítulo 5.

4.12 Resumo

Este capítulo apresentou o método proposto e a metodologia adotada para avaliá-lo.

Na primeira parte, descreveu-se o mecanismo de mineração de regras de associação para coordenação de dispositivos. Cada dispositivo registra apenas as mudanças de estado (transições) em uma matriz de contadores por dia e slot, da qual se extrai um padrão de ações. A cada checkpoint, os padrões de pares de dispositivos formam bases de transações, sobre as quais se mineram regras antecedente $\rightarrow$ consequente seguindo as métricas de suporte, confiança e lift, conservando-se uma regra por antecedente. A coordenação um para muitos é obtida por encadeamento de regras (livre de ciclos), e a mineração é feita por cômodo, combinando o comportamento de vários moradores nos espaços compartilhados.

Na segunda parte, apresentou-se a metodologia experimental. O comportamento de uso é modelado por personas sintéticas geradas por cadeias de Markov não homogêneas (probabilidades de transição por faixa horária), cuja estrutura de coordenação é conhecida. A avaliação segue um protocolo de duas fases, treinamento (geração dos dados, sem mineração) e teste (mineração e avaliação), e mede, contra o *ground truth*, a precisão, o *recall* (cobertura) e o F1, além dos diagnósticos de ciclos e pares mútuos. Para assegurar rigor, empregam-se semente determinística, N repetições com intervalo de confiança de 95% e análise de sensibilidade à confiança, sobre um conjunto de 5 casas $\times$ 4 moradores.

Definidos o método e a forma de avaliá-lo, o próximo capítulo apresenta e discute os resultados obtidos.

Capítulo 5

Resultados e Discussões

Este capítulo apresenta e interpreta os resultados da avaliação do método por simulação sobre personas sintéticas com *ground truth*. De modo geral, observou-se um comportamento assimétrico e consistente: a precisão é alta e robusta, enquanto a cobertura é moderada e diminui de forma sistemática à medida que cresce o grupo de dispositivos coordenados. Como a estrutura de coordenação de cada cômodo é conhecida, foi possível quantificar não só o que o método acerta, mas também o que deixa de recuperar.

Todos os artefatos empregados neste trabalho, o simulador, os arquivos de persona das cinco residências, o executor experimental e os dados exportados que originam as tabelas e figuras deste capítulo, podem ser encontrados em [Delgado Pérez \(2026\)](#), juntamente com o procedimento de reprodução dos experimentos aqui reportados.

5.1 Precisão e o Efeito do Ruído

A precisão manteve-se alta e estável em todos os cenários. A Tabela 5.1 resume o efeito da confiança sobre as três métricas (média das 29 instâncias).

Tabela 5.1: Efeito da confiança (média sobre as 29 instâncias)

Confiança	Recall	Precisão	F1
0,60	0,47	0,92	0,61
0,75	0,43	0,92	0,56
0,90	0,36	0,93	0,48

O comportamento mais informativo aparece ao separar os cômodos compartilhados, que contêm dispositivos de uso independente, inseridos como ruído, dos privados (limpos), na Tabela 5.2.

Tabela 5.2: Compartilhados vs. privados (média por confiança)

Grupo	Recall (0,60 / 0,75 / 0,90)	Precisão (0,60 / 0,75 / 0,90)
Compartilhados (12)	0,40 / 0,34 / 0,22	0,81 / 0,90 / 0,97
Privados (17)	0,53 / 0,49 / 0,47	$\sim 1,00$

Nos cômodos compartilhados, a precisão sobe de 0,81 (confiança 0,60) para 0,97 (confiança 0,90): o ruído gera falsos positivos ocasionais que o limiar de confiança filtra progressivamente. Nos privados, sem ruído, a precisão já é $\sim 1,00$ em todos os níveis. Este resultado valida o desenho experimental, a inclusão de ruído tornou a métrica de precisão discriminante, e não saturada em 1,00, e evidencia que o método só produz associações espúrias em regime de confiança baixa.

5.2 Cobertura e o Tamanho do Cluster Coordenado

O achado central é a queda sistemática da cobertura à medida que cresce o número de coordenações verdadeiras do cômodo. A Figura 5.1 mostra a tendência decrescente, resumida por faixas na Tabela 5.3.

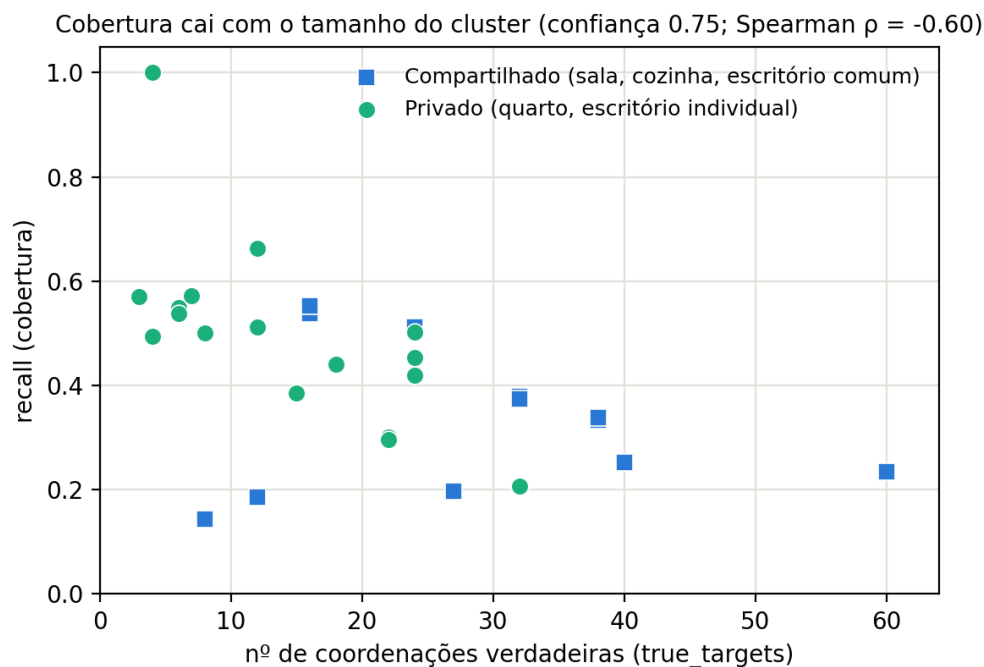


Figura 5.1: *Recall* em função do número de coordenções verdadeiras por cômodo (confiança 0,75).

Tabela 5.3: *Recall* médio por faixa de coordenções verdadeiras (confiança 0,75)

Coordenções verdadeiras	Nº de instâncias	Recall médio
≤ 8 (pequeno)	8	$\sim 0,55$
12–18 (médio)	7	$\sim 0,47$
22–38 (grande)	12	$\sim 0,36$
40–60 (muito grande)	2	$\sim 0,24$

A tendência é decrescente e o efeito é substancial: a correlação de *Spearman* entre *true_targets* e *recall* é $\rho = -0,60$ ($p < 0,001$ no cálculo exploratório, sem ajuste para o agrupamento por casa; Pearson $r = -0,55$). Os extremos ilustram o efeito: a sala (estudio) da Casa 3, com 60 coordenções verdadeiras, atinge *recall* 0,23; já o dormitório (Estudante 3) da Casa 5, com 4, atinge 1,00.

O comportamento é consistente com a limitação estrutural da Seção 4.6: ao impor grau de saída no máximo igual a 1, o mecanismo produz trajetórias sem ramificação direta; em grupos cuja referência exige múltiplos consequentes, parte das relações tende a permanecer inalcançável, e ciclos e pares mútuos podem encurtar ainda mais essas trajetórias. Trata-se de um limite intrínseco à formulação, tornado mensurável pela avaliação com verdade de referência: a cobertura não reflete uma falha de implementação, e sim até onde a topologia de regras retidas consegue propagar o dis-

paro. Note-se, contudo, que essa propriedade não implica decrescimento monotônico com o número de dispositivos (Seção 4.6); ela estabelece uma limitação que se torna mais relevante à medida que crescem a quantidade e a ramificação das coordenações verdadeiras.

5.3 Cadeia Cortada por Ciclos e Pares Mútuos

Os diagnósticos de topologia, o número de ciclos cortados e de pares mútuos, permitem examinar por que a propagação se interrompe (Tabela 5.4). Um par mútuo ocorre quando dois dispositivos se apontam reciprocamente com a mesma ação ($A \leftrightarrow B$): ao alcançá-lo, a cadeia retornaria ao ponto de partida e é interrompida pela detecção de ciclos, de modo que quaisquer dispositivos situados além desse ponto permanecem fora de alcance.

Tabela 5.4: Diagnóstico (confiança 0,75, casos ilustrativos)

Instância	<i>true_targets</i>	Pares mútuos	Ciclos cortados	<i>Recall</i>
Oficina (estúdio) – C3	60	2,04	8,85	0,23
Oficina (sala_estudo) – C5	40	3,14	9,03	0,25
Sala – C1	32	2,10	8,27	0,38
Dormitório (Estudante 2) – C5	4	0,98	1,96	0,50
Dormitório (Estudante 3) – C5	4	2,00	4,00	1,00

É importante, porém, distinguir mecanismo de preditor quantitativo. O número absoluto de pares mútuos, isoladamente, não prediz o recall entre instâncias (Spearman $\rho = -0,08$; *n.s.*). A razão é que seu efeito muda de sinal com o tamanho do grupo: em um grupo de dois dispositivos, um par mútuo $A \leftrightarrow B$ fecha a cobertura, pois ambos os sentidos tornam-se alcançáveis; em grupos maiores, o mesmo par interrompe a trajetória e deixa os demais fora de alcance. O driver quantitativo dominante é, portanto, o tamanho do cluster, e não a contagem de pares. A Tabela 5.4 torna isso visível: instâncias com contagens de pares semelhantes (~ 2) exibem *recall* muito distinto conforme o número de alvos. A cobertura depende, ainda, de como as regras de melhor pontuação se conectam: cômodos pequenos mas com rotinas escalonadas podem cobrir menos que cômodos maiores porém fortemente sincronizados, o que explica a dispersão em torno da tendência principal.

5.4 Sensibilidade à Confiança

A confiança evidencia o compromisso precisão $\times$ cobertura (Figura 5.2): 0,60 maximiza *recall* e F1 (0,61) ao custo de menor precisão nos cômodos ruidosos; 0,90 maximiza precisão mas derruba a cobertura (*recall* 0,36; algumas salas chegam a zero). A confiança 0,75 é adotada nas análises estruturais seguintes como ponto operacional intermediário, e não como ótimo global do espaço de parâmetros: como suporte e lift não foram variados, os resultados não permitem concluir que um único limiar de confiança seja ótimo para outras combinações desses parâmetros. A sensibilidade é muito mais forte nos cômodos compartilhados/ruidosos, nos privados o *recall* é praticamente insensível ao limiar, o que reforça que a confiança atua sobretudo removendo ruído.

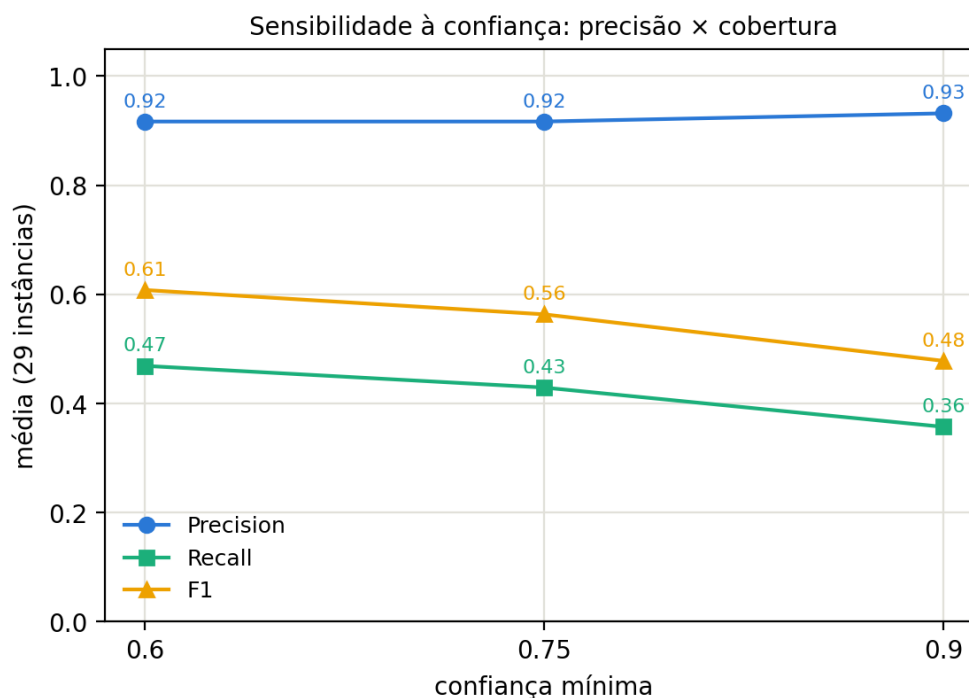


Figura 5.2: Sensibilidade das métricas (precisão, *recall* e F1) ao nível de confiança.

5.5 Estabilidade entre Casas e Mineração por Cômodo

Apesar da variação de arranjos domésticos (casais, quartos individuais, escritórios comuns e privados), os resultados são consistentes entre as cinco casas (Tabela 5.5).

Tabela 5.5: Média por casa (confiança 0,75)

Casa	<i>Recall</i>	<i>Precisão</i>	F1
1 – Família tradicional	0,43	0,93	0,58
2 – Dois trabalhadores + criança	0,41	0,95	0,56
3 – Home office duplo	0,40	0,91	0,53
4 – Casal + dois idosos	0,38	0,89	0,53
5 – República de estudantes	0,50	0,91	0,60

Essa estabilidade sugere que os efeitos observados não dependem de uma configuração particular, mas emergem da interação entre o tamanho do cluster e o mecanismo de cadeia. A mineração por cômodo, combinando moradores nos espaços compartilhados e separando os privados, mantém a coordenação fisicamente plausível; os cômodos compartilhados, por reunirem o comportamento de vários moradores, formam os maiores clusters e concentram, coerentemente, os menores valores de cobertura.

5.6 Convergência e Escolha de N

A Figura 5.3 mostra a média acumulada de F1 em função do número de execuções. As curvas apresentam estabilização visual antes de $N = 100$ execuções (variações após ~ 20 -30 repetições inferiores a $\pm 0,01$) fornecendo evidência empírica de que esse número de repetições é adequado aos cenários avaliados. Como não foi definido um critério formal de convergência, essa observação não deve ser interpretada como prova geral de suficiência de $N = 100$. Cabe lembrar que as repetições reduzem o erro de simulação, mas não constituem ambientes independentes.

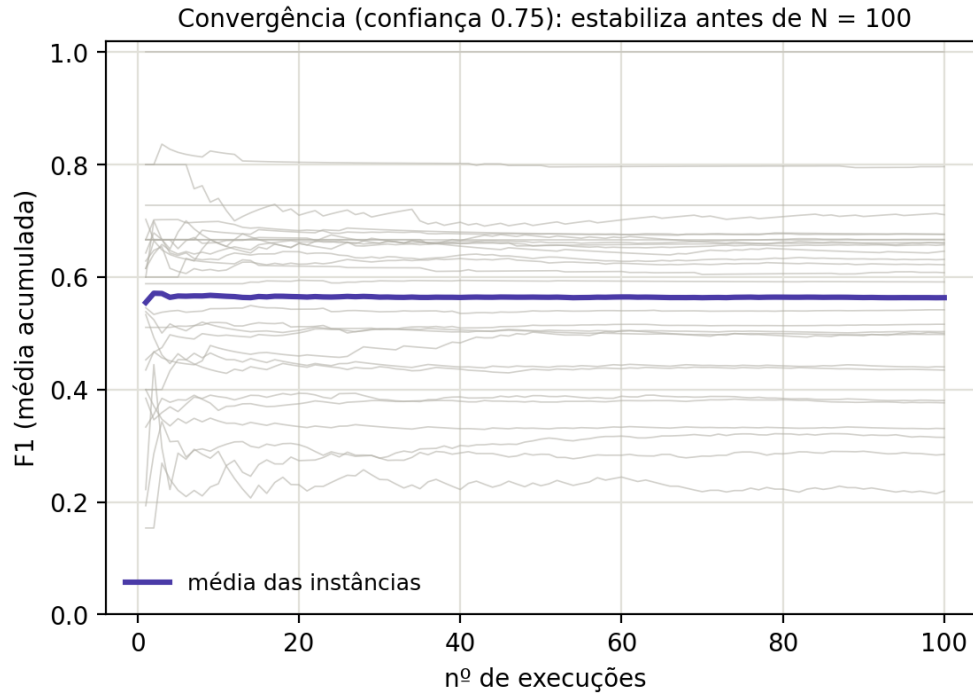


Figura 5.3: Convergência da média acumulada de F1 em função do número de execuções.

5.7 Resumo

Este capítulo apresentou e discutiu os resultados da avaliação do método por simulação. A precisão mostrou-se alta e estável ($\sim 0,92$ no geral; próxima de 1,00 nos cômodos sem ruído), e os dispositivos de ruído evidenciaram que associações espúrias só surgem em regime de confiança baixa, sendo filtradas ao elevar o limiar (de 0,81 para 0,97 nos cômodos compartilhados). A cobertura (*recall*), por outro lado, foi moderada e decresce com o tamanho do grupo coordenado (Spearman $\rho = -0,60$, evidência exploratória), comportamento consistente com a limitação representacional do mecanismo, cujo grafo de regras retidas tem grau de saída ≤ 1 e produz trajetórias sem ramificação direta, podendo ainda ser encurtadas por ciclos e pares mútuos. Distinguiu-se mecanismo de preditor: a contagem de pares mútuos, sozinha, não prediz a cobertura ($\rho = -0,08$, n.s.), pois seu efeito muda de sinal com o tamanho do grupo: fecha a cobertura em pares e interrompe-a em grupos maiores.

A análise de sensibilidade adotou a confiança 0,75 como ponto operacional intermediário (0,60 e 0,90 como limites), e os resultados mostraram-se estáveis entre as cinco casas; as curvas de média acumulada apresentaram estabilização empírica antes de $N = 100$. Os resultados devem ser lidos dentro do desenho experimental adotado: sensibilidade explorada apenas para a confiança e

instâncias aninhadas em cinco casas.

Capítulo 6

Conclusão

Esta dissertação implementou e avaliou, por simulação, o DARE, mecanismo descentralizado de coordenação de dispositivos IoT baseado em mineração de regras de associação [Alencar et al. \(2020\)](#). A avaliação foi conduzida sobre personas sintéticas geradas por cadeias de Markov, cuja estrutura de coordenação é conhecida, o que permitiu medir, sob um *ground truth* controlado e de forma reproduzível, a capacidade do método de recuperar as relações reais entre os dispositivos.

Os resultados mostraram um comportamento assimétrico e consistente. A precisão é alta ($\sim 0,92$ no geral; próxima de 1,00 nos cômodos sem ruído), e a inclusão de dispositivos de uso independente confirmou que o método só produz associações espúrias em regime de confiança baixa, filtradas ao elevar o limiar de confiança. A cobertura (*recall*), por sua vez, é moderada e diminui de forma sistemática com o tamanho do grupo coordenado (Spearman $\rho = -0,60$), efeito quantificado graças ao *ground truth*. A análise estrutural explica esse comportamento: ao reter no máximo um consequente por antecedente, o grafo de regras possui grau de saída ≤ 1 e produz trajetórias sem ramificação direta, o que restringe a representação de coordenações um para muitos mais densas; ciclos e pares mútuos podem reduzir adicionalmente o alcance, embora o tamanho e a topologia do grupo sejam os fatores centrais. Os resultados mostraram-se estáveis entre as cinco casas avaliadas, sugerindo que os efeitos não dependem de uma configuração particular.

6.1 Contribuições

As principais contribuições deste trabalho situam-se em três frentes. Uma forma de avaliar mecanismos de coordenação por regras de associação sob verdade de referência, cujo elemento central é a

derivação analítica do gabarito a partir dos próprios parâmetros que geram o comportamento. Em vez de rotular manualmente as coordenações, calcula-se a esperança das transições pela recorrência de Markov e enumeram-se as coordenações que o ambiente implica, o que evita avaliar o minerador contra uma referência produzida por ele mesmo e permite medir cobertura, e não apenas precisão ou concordância.

A caracterização quantitativa do comportamento do mecanismo sob essa referência, precisão elevada e estável, ruído filtrado pelo aumento da confiança, cobertura decrescente com o tamanho do grupo coordenado e, sobretudo, a distinção entre mecanismo e preditor, já que ciclos e pares mútuos explicam por que a cadeia é curta, mas sua contagem, isoladamente, não prediz a cobertura.

Também artefatos produzidos, reutilizáveis por outros trabalhos: o simulador de ambiente residencial inteligente, que implementa o mecanismo e admite tanto interação manual quanto geração sintética; o conjunto de personas, com cinco residências de quatro moradores cada, cobrindo diferentes arranjos domésticos; e o executor experimental, que automatiza as repetições com semente controlada, a varredura de parâmetros e a exportação dos resultados.

6.2 Limitações

Os resultados devem ser interpretados dentro do desenho experimental adotado. O comportamento é gerado por personas sintéticas, e não observado em habitantes reais; os valores absolutos devem ser lidos como propriedades do mecanismo sob estrutura conhecida, não como desempenho em campo. A verdade de referência depende dos limiares empregados em sua construção, de modo que as métricas são relativas ao conjunto de coordenações detectáveis naquele regime. A sensibilidade foi explorada apenas para a confiança, com suporte e *lift* fixos, o que torna essas conclusões condicionais a tais valores. As 29 instâncias de cômodo estão aninhadas em cinco residências, e por isso a correlação reportada tem caráter exploratório. Por fim, a combinação de moradores por união e mínimo é uma aproximação que não modela disputas de estado, e a estabilização da média acumulada antes de 100 repetições constitui evidência empírica para os cenários avaliados, não uma prova geral de convergência.

6.3 Trabalhos futuros

A limitação representacional identificada e o escopo do desenho experimental adotado apontam quatro direções de continuidade, listadas da mais diretamente decorrente dos resultados às de caráter mais amplo:

- **Extensão da coordenação para além da cadeia linear.** O principal resultado, a queda da cobertura em *clusters* grandes, motiva permitir mais de uma regra por antecedente (ou uma propagação em árvore), o que garantiria a cobertura em grupos densamente coordenados; avaliá-la com a mesma métrica seria um passo direto.
- **Validação com dados reais.** Aplicar o mesmo protocolo (duas fases + métricas) sobre traços reais de uso permitiria contrastar as personas sintéticas com o comportamento observado.
- **Modelo de moradores mais rico.** Incorporar disputas de estado entre moradores (ações opostas simultâneas) e ampliar a diversidade estrutural dos arquétipos tornariam o modelo de espaços compartilhados mais realista.
- **Ampliação do desenho experimental.** Explorar uma grade conjunta de suporte, confiança e *lift* (em vez de variar apenas a confiança) e empregar análise estatística que modele explicitamente a dependência entre cômodos de uma mesma residência.

6.4 Considerações Finais

A avaliação de mecanismos de coordenação em ambientes inteligentes esbarra, com frequência, na ausência de um gabarito: sem saber quais relações deveriam ser descobertas, resta verificar a coerência das regras produzidas. Ao construir um ambiente cuja estrutura de coordenação é conhecida por construção, esta dissertação tornou possível deslocar a pergunta de "as regras são plausíveis?" para "quanto da coordenação real elas recuperam?". A resposta obtida, precisão elevada acompanhada de cobertura parcial, limitada pela representação em cadeia, não desqualifica o mecanismo avaliado; ao contrário, delimita com precisão o seu escopo de aplicação e indica onde estão os ganhos possíveis.

Referências Bibliográficas

- Abdmeziem, R. e Tandjaoui, D. (2014), Internet of things: Concept, building blocks, applications and challenges, *ArXiv* abs/1401.6877. URL : <https://api.semanticscholar.org/CorpusID:1235952>
- Aceron, J. B. L., Teves, M. E. R. e Tan, W. M. (2021), Detecting application-level associations between IoT devices using a modified apriori algorithm, *in* 2021 International Conference on Smart Applications, Communications and Networking (SmartNets), IEEE, Glasgow, United Kingdom, pp. 1–8.
- Ai, J. (2023), The basic principles and application areas of embedded systems, *Advances in Computer, Signals and Systems* . URL : <https://api.semanticscholar.org/CorpusID:266182461>
- Alencar, M. A. d. C. (2019), Extração descentralizada de conhecimento associativo para internet das coisas, Dissertação (mestrado em informática), Universidade Federal do Amazonas, Manaus, Brasil.
- Alencar, M., Barreto, R. S., Fernandes, H., Souto, E. J. P. e Pazzi, R. W. N. (2020), Dare: A decentralized association rules extraction scheme for embedded data sets in distributed iot devices, *International Journal of Distributed Sensor Networks* 16. URL : <https://api.semanticscholar.org/CorpusID:226341709>
- Alshammari, N., Alshammari, T., Sedky, M., Champion, J. e Bauer, C. (2017), Openshs: Open smart home simulator, *Sensors* 17(5), 1003.
- Alshammari, T., Alshammari, N., Sedky, M. e Howard, C. (2018), Simadl: Simulated activities of daily living dataset, *Data* 3(2), 11.
- Anderson, G., Marwala, T. e Nelwamondo, F. V. (2010), Use of data mining in scheduler optimization, *ArXiv* abs/1011.1735. URL : <https://api.semanticscholar.org/CorpusID:13904176>

- Anirudh, V. e Kumar, M. (2020), A conceptual study on fundamentals and characteristics of embedded systems, *International Journal of Engineering Technology and Management Sciences* . URL : <https://api.semanticscholar.org/CorpusID:225368418>
- Athauda, R. I., Tissera, M. e Fernando, C. (2009), Data mining applications: Promise and challenges. URL : <https://api.semanticscholar.org/CorpusID:266561683>
- Bader, I. M. e Abusitta, A. (2013), A water-saving technique using embedded system, *Applied Mathematics-a Journal of Chinese Universities Series B* 2013, 119–123. URL : <https://api.semanticscholar.org/CorpusID:55300558>
- Bashir, S., Jan, Z. e Baig, A. R. (2009), Fast algorithms for mining interesting frequent itemsets without minimum support, *ArXiv* abs/0904.3319. URL : <https://api.semanticscholar.org/CorpusID:926574>
- Bouchabou, D., Nguyen, S. M., Lohr, C., Leduc, B. e Kanellos, I. (2021), A survey of human activity recognition in smart homes based on iot sensors algorithms: Taxonomies, challenges, and opportunities with deep learning, *Sensors* 21(18), 6037.
- Bourahla, M. (2009), Modeling and verification of real-time embedded systems. URL : <https://api.semanticscholar.org/CorpusID:17224483>
- Buaton, R., Mawengkang, H., Zarlis, M., Effendi, S., Pardede, A. M. H., Maulita, Y., Fauzi, A., Novriyenni, N., Sihombing, A. e Lumbanbatu, K. (2019), Decision tree optimization in data mining with support and confidence, *Journal of Physics: Conference Series* 1255. URL : <https://api.semanticscholar.org/CorpusID:202779235>
- Butun, I., Österberg, P. e Song, H. (2019), Security of the internet of things: Vulnerabilities, attacks, and countermeasures, *IEEE Communications Surveys & Tutorials* 22, 616–644. URL : <https://api.semanticscholar.org/CorpusID:204950321>
- Cao, L., Gorodetsky, V. e Mitkas, P. A. (2009), Agent mining: The synergy of agents and data mining, *IEEE Intelligent Systems* 24(3), 64–72.
- Cheng, K.-C., Huang, M.-J., Fu, C.-K., Wang, K.-H., Wang, H.-M. e Lin, L.-H. (2021), Establishing a multiple-criteria decision-making model for stock investment decisions using data mining techniques, *Sustainability* . URL : <https://api.semanticscholar.org/CorpusID:233666744>

- Cook, D. J., Crandall, A. S., Thomas, B. L. e Krishnan, N. C. (2013), Casas: A smart home in a box, *Computer* 46(7), 62–69.
- DeGroot, M. H. e Schervish, M. J. (2012), *Probability and Statistics*, Addison–Wesley. URL : <https://books.google.com.br/books?id=4TIEPgAACAAJ>
- Delgado Pérez, L. (2026), Simulador de coordenação de dispositivos em ambientes IoT, Software. Licença MIT. URL : <https://github.com/leslydp/simulador-coordenacao-iot>
- Ding, S. e Wang, X. (2020), Medical remote monitoring of multiple physiological parameters based on wireless embedded internet, *IEEE Access* 8, 78279–78292. URL : <https://api.semanticscholar.org/CorpusID:218564460>
- Ettinger, T. (2017), Mulberry32 PRNG, GitHub Gist. Acesso em: 21 set. 2026. URL : <https://gist.github.com/tommyettinger/46a874533244883189143505d203312c>
- Fan, Y.-H. (2020), Multiple-embedded-system optimization layout for electromagnetic wave power density in complex environments, *Energies* . URL : <https://api.semanticscholar.org/CorpusID:224863933>
- Gnanapriya, S., Suganya, R., Devi, G. e Kumar, M. S. (2010), Data mining concepts and techniques, *Data mining and knowledge engineering* 2, 256–263. URL : <https://api.semanticscholar.org/CorpusID:57146809>
- Gullo, F. (2015), From patterns in data to knowledge discovery: What data mining can do, *Physics Procedia* 62, 18–22. URL : <https://api.semanticscholar.org/CorpusID:143130873>
- Han, J., Pei, J. e Yin, Y. (2000), Mining frequent patterns without candidate generation, in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, ACM, pp. 1–12.
- Hingane, R. S. e Patil, S. (2014), Mad-arm: Distributed association rule mining using mobile agents, *International Journal of Advanced Engineering Research and Science (IJAERS)* 1(4), 52–57.
- Huang, Y., Cheng, Z., Zhou, Q., Xiang, Y. e Zhao, R. (2020), Data mining algorithm for cloud network information based on artificial intelligence decision mechanism, *IEEE Access* 8, 53394–53407. URL : <https://api.semanticscholar.org/CorpusID:214692878>

- Ibrahim, I., Ali, R., Zulkefli, M. S. H. e Elfadil, N. (2015), Embedded systems pedagogical issue: Teaching approaches, students readiness, and design challenges. URL : <https://api.semanticscholar.org/CorpusID:14659779>
- Ishak, M. K., Hwan, O. J., Jiashen, T. e Isa, N. A. M. (2019), Automated compilation test system for embedded system, *Makara Journal of Technology* . URL : <https://api.semanticscholar.org/CorpusID:68044939>
- Jabas, A. (2011), Manet mining: Mining association rules. URL : <https://api.semanticscholar.org/CorpusID:53516025>
- Jang, H.-J., Yang, Y., Park, J. S. e Kim, B. (2021), FP-Growth algorithm for discovering region-based association rule in the IoT environment, *Electronics* 10(24), 3091.
- Karabulut, E., Groth, P. e Degeler, V. (2025), Learning semantic association rules from internet of things data, *Neurosymbolic Artificial Intelligence* 1.
- Karahoca, A., Karahoca, D. e Sanver, M. R. (2012), Survey of data mining and applications (review from 1996 to now). URL : <https://api.semanticscholar.org/CorpusID:60574853>
- Kelemen, M., Miková, L., Hroncová, D., Filakovský, F. e Sinák, P. J. (2020), Embedded systems – control of power subsystems, *Acta Mechatronica* . URL : <https://api.semanticscholar.org/CorpusID:234733680>
- Khedr, A. M., Osamy, W., Salim, A. e Abbas, S. (2020), A novel association rule-based data mining approach for internet of things based wireless sensor networks, *IEEE Access* 8, 151574–151588. URL : <https://api.semanticscholar.org/CorpusID:221385412>
- Khedr, A. M., Osamy, W., Salim, A. e Salem, A.-A. (2019), Privacy preserving data mining approach for iot based wsn in smart city, *International Journal of Advanced Computer Science and Applications* . URL : <https://api.semanticscholar.org/CorpusID:202752468>
- Kireev, V. S., Guseva, A. I., Bochkaryov, P. V., Kuznetsov, I. A. e Filippov, S. A. (2018), Association rules mining for predictive analytics in IoT cloud system, in *Biologically Inspired Cognitive Architectures 2018*, Vol. 848 of *Advances in Intelligent Systems and Computing*, Springer, Cham, pp. 107–112.

- Köckemann, U., Alirezaie, M., Renoux, J., Tsiftes, N., Ahmed, M. U., Morberg, D., Lindén, M. e Loutfi, A. (2020), Open-source data collection and data sets for activity recognition in smart homes, *Sensors* 20(3), 879.
- Kouah, S., Saighi, A., Ammi, M., Naït Si Mohand, A., Kouah, M. I. e Megías, D. (2024), Internet of things-based multi-agent system for the control of smart street lighting, *Electronics* 13(18), 3673.
- Kumar, D. S. (2020), The design and analysis of char driver model for arm architecture, *International Journal Of Recent Advances in Engineering & Technology* . URL : <https://api.semanticscholar.org/CorpusID:219799332>
- Lin, B., Zhu, F., Zhang, J., Chen, J., Chen, X., Xiong, N. N. e Mauri, J. L. (2019), A time-driven data placement strategy for a scientific workflow combining edge computing and cloud computing, *IEEE Transactions on Industrial Informatics* 15, 4254–4265. URL : <https://api.semanticscholar.org/CorpusID:58981499>
- Lin, F., Zhou, Y., You, I., Lin, J., An, X. e Lü, X. (2019), Content recommendation algorithm for intelligent navigator in fog computing based iot environment, *IEEE Access* 7, 53677–53686. URL : <https://api.semanticscholar.org/CorpusID:145050612>
- Lin, J. C.-W., Djenouri, Y., Srivastava, G. e Fournier-Viger, P. (2021), Mining profitable and concise patterns in large-scale internet of things environments, *Wireless Communications and Mobile Computing* . URL : <https://api.semanticscholar.org/CorpusID:240550634>
- Lynn, T., Endo, P. T., Ribeiro, A. M. N. C., Barbosa, G. B. N. e Rosati, P. (2020), The internet of things: Definitions, key concepts, and reference architectures, *The Cloud-to-Thing Continuum* . URL : <https://api.semanticscholar.org/CorpusID:226527662>
- Mai, T., Nguyen, L. T. T., Vo, B., Yun, U. e Hong, T.-P. (2020), Efficient algorithm for mining non-redundant high-utility association rules, *Sensors (Basel, Switzerland)* 20. URL : <https://api.semanticscholar.org/CorpusID:211231397>
- Mints, A. (2017), Classification of tasks of data mining and data processing in the economy, *Baltic Journal of Economic Studies* 3, 47–52. URL : <https://api.semanticscholar.org/CorpusID:55748319>

- Nomura, K., Shiraishi, Y., Mohri, M. e Morii, M. (2020), Secure association rule mining on vertically partitioned data using private-set intersection, *IEEE Access* 8, 144458–144467. URL : <https://api.semanticscholar.org/CorpusID:221159892>
- Ogunde, A. O., Folorunso, O. e Sodiya, A. S. (2012), On the adaptivity of distributed association rule mining agents, *in* Proceedings of the Fourth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE), pp. 71–77.
- Rani, A. (2021), A proposal for architectural framework using internet of things with fog computing for an air quality monitoring system, *Journal of Informatics Electrical and Electronics Engineering (JIEEE)* . URL : <https://api.semanticscholar.org/CorpusID:240006864>
- Rashid, M. M., Kamruzzaman, J., Hassan, M. M., Shafin, S. S. e Bhuiyan, M. Z. A. (2020), A survey on behavioral pattern mining from sensor data in internet of things, *IEEE Access* 8, 33318–33341. URL : <https://api.semanticscholar.org/CorpusID:211684416>
- Ross, S. M. (2014), *Introduction to Probability Models*, 11 edn, Academic Press, Amsterdam.
- Ruiz, M. D., Fernández-Basso, C., Gómez-Romero, J. e Martín-Bautista, M. J. (2023), A big data association rule mining based approach for energy building behaviour analysis in an IoT environment, *Scientific Reports* 13, 19810.
- Saleem, A. e Prakash Raja, E. G. D. (2017), Madparm: Mobile agent based distributed and parallel association rule mining, *International Journal of Engineering Trends and Technology (IJETT)* 49(6), 375–382.
- Sarker, I. H., Khan, A. I., Abushark, Y. B. e Alsolami, F. J. (2022), Internet of things (iot) security intelligence: A comprehensive overview, machine learning solutions and research directions, *Mobile Networks and Applications* 28, 296–312. URL : <https://api.semanticscholar.org/CorpusID:247388374>
- Sen, J., Lee, M., Lee, S., Choe, Y., Domb, M., Pal, A., Rath, H. K., Shailendra, S., Bhattacharyya, A., Mihovska, A. D., Sarkar, M., Lee, H. J., Kim, M. e Averian, A. (2018), Internet of things: Technology, applications and standardization, *ArXiv* abs/1808.09390. URL : <https://api.semanticscholar.org/CorpusID:52114249>

- Shankar, S. K. e Purusothaman, T. (2010), Discovering imperceptible associations based on interestingness: A utility-oriented data mining, *Data Sci. J.* 9, 1–12. URL : <https://api.semanticscholar.org/CorpusID:18375824>
- song Li, J., Yu, H.-Y. e Zhang, X. (2011), Data mining in hospital information system. URL : <https://api.semanticscholar.org/CorpusID:13804410>
- Soong, T. T. (2004), *Fundamentals of Probability and Statistics for Engineers*, Wiley, Chichester; Hoboken, NJ.
- Suryana, A. e Yulianto, E. (2019), Application of data mining with association rules to review relationship between insured, products selection and customer behavior, *Universal Journal of Electrical and Electronic Engineering* . URL : <https://api.semanticscholar.org/CorpusID:195504537>
- Widaningsih, S., Muhamad, W., Hendriyanto, R. e Nugroho, H. (2023), An id3 decision tree algorithm-based model for predicting student performance using comprehensive student selection data at telkom university, *Ingénierie des systèmes d'information* . URL : <https://api.semanticscholar.org/CorpusID:265059530>
- Winarno, I., Barakbah, A. R., Pramadihanto, D., Sesulihatien, W. T., Harsono, T. N., Dewantara, B. S. B., Setiawardhana, S., Fariza, A., Syarif, I., Badriyah, T., Iskandariansyah, I. e Susanti, P. (2022), Embedded system training based on arduino to improve software programming knowledge for vocational students, *Abdimas: Jurnal Pengabdian Masyarakat Universitas Merdeka Malang* . URL : <https://api.semanticscholar.org/CorpusID:253561453>
- Zaki, M. J. (2000), Scalable algorithms for association mining, *IEEE Transactions on Knowledge and Data Engineering* 12(3), 372–390.