UNIVERSIDADE FEDERAL DE PELOTAS
Centro de Desenvolvimento Tecnológico
Programa de Pós-Graduação em Computação

Tese

Arquitetura DyCoRS: uma Abordagem para Ciência de Contexto Explorando Composição Dinâmica de Estratégias para Raciocínio

Roger da Silva Machado

Pelotas, 2019

Roger da Silva Machado

Arquitetura DyCoRS: uma Abordagem para Ciência de Contexto Explorando Composição Dinâmica de Estratégias para Raciocínio

Tese apresentada ao Programa de Pós-Graduação em Computação do Centro de Desenvolvimento Tecnológico da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Doutor em Ciência da Computação.

Orientador: Prof. Dr. Adenauer Corrêa Yamin
Coorientadora: Prof[a]. Dr[a]. Ana Marilza Pernas

Pelotas, 2019

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação na Publicação

M149a Machado, Roger da Silva

Arquitetura DyCoRS : uma abordagem para ciência de contexto explorando composição dinâmica de estratégias para raciocínio / Roger da Silva Machado ; Adenauer Corrêa Yamin, orientador ; Ana Marilza Pernas, coorientadora. — Pelotas, 2019.

101 f. : il.

Tese (Doutorado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2019.

1. Raciocínio de contexto. 2. Ciência de contexto. 3. Computação ubíqua. I. Yamin, Adenauer Corrêa, orient. II. Pernas, Ana Marilza, coorient. III. Título.

CDD : 005

Elaborada por Aline Herbstrith Batista CRB: 10/1737

Roger da Silva Machado

Arquitetura DyCoRS: uma Abordagem para Ciência de Contexto Explorando Composição Dinâmica de Estratégias para Raciocínio

Tese aprovada, como requisito parcial, para obtenção do grau de Doutor em Ciência da Computação, Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas.

Data da Defesa: 20 de dezembro de 2019

Banca Examinadora:

Prof. Dr. Adenauer Corrêa Yamin (orientador)
Doutor em Computação pela Universidade Federal do Rio Grande do Sul.

Profª. Drª. Ana Marilza Pernas (coorientadora)
Doutora em Computação pela Universidade Federal do Rio Grande do Sul.

Prof. Dr. Cristiano André da Costa
Doutor em Computação pela Universidade Federal do Rio Grande do Sul.

Prof. Dr. José Palazzo Moreira de Oliveira
Doutor em Informática pelo Instituto Nacional Politécnico de Grenoble.

Profª. Drª. Tatiana Aires Tavares
Doutora em Engenharia Elétrica pela Universidade Federal do Rio Grande do Norte.

AGRADECIMENTOS

Primeiramente, quero agradecer a Deus, pois sei que sem ele nada poderia ter sido realizado.

Quero agradecer a minha família, meus pais que sempre me deram todo o apoio para me dedicar aos estudos, aos meus irmãos e a minha vó Geltrudes que me criou e sei que me tornei a pessoa que sou hoje graças a ela. Um agradecimento especial para minha noiva Fernanda, por todo o incentivo para me dedicar a este trabalho e pela compreensão durante esses anos.

Agradeço aos meus orientadores Prof. Dr. Adenauer Corrêa Yamin e Prof^{a}. Dr^{a}. Ana Marilza Pernas pela parceria realizada durante todos esses anos, parceria essa que começou ainda na graduação. Muito obrigado por toda a dedicação, ensinamentos, contribuições e grande incentivo na realização deste trabalho, aprendi muito durante esses anos e espero poder continuar tendo a oportunidade de trabalhar com vocês.

Agradeço a todos os colegas do grupo de pesquisa LUPS, com os quais compartilhei longas horas, realizando diversas atividades durante os quatro anos de doutorado.

RESUMO

MACHADO, Roger da Silva. **Arquitetura DyCoRS: uma Abordagem para Ciência de Contexto Explorando Composição Dinâmica de Estratégias para Raciocínio**. Orientador: Adenauer Corrêa Yamin. 2019. 101 f. Tese (Doutorado em Ciência da Computação) – Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2019.

A computação ubíqua devido a sua elevada escalabilidade utiliza mecanismos de gerência que tenham operação mais autônoma possível, necessitando para isso de abordagens que viabilizem a ciência de contexto. A ciência de contexto pode ser estruturada em quatro etapas: aquisição; modelagem; processamento; e disseminação. Dentre estas etapas destaca-se a de processamento de contexto, a qual, dentre outras funcionalidades, é responsável por realizar o raciocínio de contexto inferindo novos conhecimentos e melhorando a compreensão dos contextos adquiridos. Diversas estratégias vêm sendo utilizadas para realizar o raciocínio de contexto, sendo que ainda não foi encontrada uma estratégia que possa ser aplicada de maneira igualmente satisfatória em diferentes cenários de aplicação. Em função disto, as abordagens ditas híbridas vêm ganhando destaque. Essas abordagens permitem a combinação de duas ou mais estratégias de raciocínio, podendo aproveitar os pontos positivos e minimizar os aspectos negativos de cada estratégia. Apesar destas abordagens já estarem sendo usadas, ainda nota-se uma falta de flexibilidade na disponibilização dos recursos oferecidos. Assim, este trabalho tem como objetivo a concepção de uma arquitetura chamada DyCoRS, a qual é dinâmica no oferecimento dos recursos necessários para obtenção da ciência de contexto. Dentre os recursos oferecidos destaca-se a abordagem híbrida para raciocínio de contexto, a qual permite uma composição dinâmica de diferentes estratégias. Além disso, a arquitetura ainda provê: (i) diferentes modelos de armazenamento; (ii) um método para correlação das informações persistidas nos diversos modelos; e (iii) um processo que permite a reconfiguração dos recursos providos pela arquitetura em tempo de execução. A arquitetura proposta foi avaliada por meio de cenários de uso, explorando diferentes funcionalidades providas pela DyCoRS em cada um deles. Os resultados obtidos mostram que é possível oferecer uma maior flexibilidade para as aplicações cientes de contexto escolherem como querem realizar o raciocínio das informações contextuais. Desta forma, os resultados alcançados estimulam a continuação da pesquisa.

Palavras-chave: Raciocínio de Contexto. Ciência de Contexto. Computação Ubíqua.

ABSTRACT

MACHADO, Roger da Silva. **DyCoRS Architecture: A Hybrid Approach to Context Awareness Exploring Dynamic Composition of Reasoning Strategies** . Advisor: Adenauer Corrêa Yamin. 2019. 101 f. Thesis (Doctorate in Computer Science) – Technology Development Center, Federal University of Pelotas, Pelotas, 2019.

Due to its high scalability, ubiquitous computing uses management mechanisms that have the most autonomous operation possible, requiring approaches that enable context awareness. Context awareness can be structured in four steps: acquisition; modeling; processing; and dissemination. Among these steps, we highlight the context processing, which, among other features, it is responsible for performing contextual reasoning inferring new knowledge and improving understanding of acquired contexts. Several strategies have been used to implement context reasoning, and a strategy that can be applied satisfactorily in different application scenarios has not yet been found. Because of this, hybrid approaches for context reasoning are gaining prominence. These approaches allow the combination of two or more strategies, being able to take advantage of the positive points and minimize the negative aspects of each strategy. Although these approaches are already being used, we still note a lack of flexibility in making available the resources offered. Thus, this work aims to design an architecture called DyCoRS that is dynamic in providing the necessary resources to obtain context awareness. Among the features offered, we highlight the hybrid approach for context reasoning, which allows the dynamic composition of different strategies. Besides, the architecture also provides: (i) different storage models; (ii) a method for correlating persisted information in the various models; and (iii) a process that allows the reconfiguration of the resources provided by the architecture at runtime. We evaluate the proposed architecture through usage scenarios, exploring in each one of them, different functionalities provided by DyCoRS. The results show that it is possible to offer greater flexibility for context-aware applications to choose how they want to perform the reasoning of contextual information. Thus, the results stimulate further research.

Keywords: Context Reasoning. Context Awareness. Ubiquitous Computing.

LISTA DE FIGURAS

1	Visão das etapas envolvidas para a obtenção da ciência de contexto.	20
2	Volume da produção cientifica retornado por ano.	35
3	Fluxo do processo de revisão sistemática.	36
4	Artigos retornados em cada base de indexação.	37
5	Visão geral do middleware CARE.	39
6	Visão geral da abordagem DynamiCC.	41
7	Visão geral do sistema COSAR.	42
8	Visão geral do middleware IMERA.	43
9	Fases da estratégia híbrida do framework Mining Minds	45
10	Abordagem do processamento de contexto em CASP.	46
11	Estrutura do raciocínio híbrido de contexto do framework CARA.	47
12	Abordagem empregada para raciocínio dos produtos de varejo.	49
13	Abordagem híbrida do middleware SWARMs.	50
14	Organização arquitetural da DyCoRS.	55
15	Diagrama ER das regras composicionais de raciocínio	59
16	Fluxo de execução do Componente de Raciocínio	61
17	Diagrama de classes do Módulo de Acesso ao Repositório	63
18	Diagrama ER das regras de correlação	64
19	Fluxo de execução do Componente de Correlação	65
20	Diagrama ER das regras de reconfiguração	66
21	Fluxo de execução do Componente de Reconfiguração	67
22	Visão geral do tratamento das informações realizado no cenário 1.	74
23	Regra composicional utilizada no cenário 1.	75
24	Regras composicionais referentes a M1 e M2	75
25	Regra M1 com a técnica referente a tag A1.	76
26	Regra M2 com a técnica referente a tag A2.	76
27	Regra aplicada para geração de alertas no cenário 1.	77
28	Visão geral do tratamento das informações realizado no cenário 2.	80
29	Regra composicional utilizada no cenário 2.	80
30	Regra aplicada no cenário 2.	82
31	Visão geral do tratamento das informações realizado no cenário 3.	83
32	Regra de correlação usada para identificar a situação de interesse.	85

LISTA DE TABELAS

1 Comparação entre abordagens para modelagem de contexto. . . . 25

2 Comparação entre as estratégias para raciocínio de contexto. . . . 33

3 Número de artigos suprimidos em cada critério de exclusão. 38
4 Comparação entre os trabalhos relacionados. 51

5 Comparação entre os trabalhos relacionados e a DyCoRS. 69

6 Escore de alerta precoce com base nos valores dos sinais vitais. . . 74
7 Instâncias presentes em cada categoria dos sinais vitais. 78
8 Precisão das estratégias no cenário 1. 78
9 Precisão das estratégias no cenário 2. 82

LISTA DE ABREVIATURAS E SIGLAS

API	*Application Programming Interface*
BSD	*Berkeley Software Distribution*
CARA	*Context-Aware Real-time Assistant*
CARE	*Context Aggregation and REasoning*
CASP	*Context-Aware Service Platform*
CEP	*Complex Event Processing*
CRF	*Conditional Random Field*
DoS	*Denial of Service*
DyCoRS	*Dynamic Composition of Reasoning Strategies*
ER	Entidade Relacionamento
EXEHDA	*Execution Environment for Highly Distributed Applications*
HMM	*Hidden Markov Models*
HTTP	*Hypertext Transfer Protocol*
HP	*Hewlett-Packard*
IoT	*Internet of Things*
IP	*Internet Protocol*
MIMIC	*Multiparameter Intelligent Monitoring in Intensive Care*
OPM	*Operator Profile Manager*
OWL	*Ontology Web Language*
R2L	*Remote to Local*
RDF	*Resource Description Framework*
RHIC	Repositório Híbrido de Informações Contextuais
RSL	Revisão Sistemática da Literatura
SMS	*Short Message Service*
SPPM	*Service Provider Profile Manager*
SVM	*Support Vector Machine*

SWARM	*Smart and Networking Underwater Robots in Cooperation Meshe*
SWRL	*Semantic Web Rule Language*
TAM	*Technical Architecture Modeling*
U2R	*User to Root*
UbiComp	*Ubiquitous Computing*
UML	*Unified Modeling Language*
UPM	*User Profile Manager*
URI	*Uniform Resource Identifier*
UTI	Unidade de Terapia Intensiva
XML	*eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO 14
1.1 Questão de Pesquisa 16
1.2 Objetivos 16
1.3 Estrutura do Texto 17

2 CIÊNCIA DE CONTEXTO NA UBICOMP 18
2.1 Aquisição de Contexto 20
2.2 Modelagem de Contexto 22
2.3 Processamento de Contexto 25
2.4 Disseminação de Contexto 26
2.5 Considerações do Capítulo 27

3 UMA REVISÃO DE ESTRATÉGIAS PARA RACIOCÍNIO DE CONTEXTO . 28
3.1 Baseadas em Aprendizagem não Supervisionada 28
3.2 Baseadas em Aprendizagem Supervisionada 29
3.3 Baseadas em Casos 29
3.4 Baseadas em Lógica Descritiva 30
3.5 Baseadas em Lógica Fuzzy 30
3.6 Baseadas em Lógica Probabilística 31
3.7 Baseadas em Regras 31
3.8 Análise das Estratégias para Raciocínio de Contexto 32
3.9 Considerações do Capítulo 33

4 ESTADO DA ARTE EM RACIOCÍNIO HÍBRIDO DE CONTEXTO 34
4.1 Explorando a RSL na Construção do Estado da Arte 34
4.2 Trabalhos Relacionados Identificados na RSL Realizada 39
4.2.1 Raciocínio Baseado Lógica Descritiva e em Regras 39
4.2.2 Raciocínio Baseado Aprendizagem Supervisionada e em Lógica Descritiva 41
4.2.3 Raciocínio Baseado em Casos e em Lógica Descritiva 45
4.2.4 Raciocínio Baseado em Casos e em Lógica Fuzzy 46
4.2.5 Raciocínio Baseado em Aprendizagem Supervisionada e Lógica Probabilística 48
4.2.6 Raciocínio Baseado em Lógica Descritiva e em Lógica Probabilística . . 49
4.2.7 Análise dos Trabalhos Relacionados a Raciocínio Híbrido de Contexto . 50
4.3 Discussão das Questões de Pesquisa da RSL 52
4.4 Considerações do Capítulo 53

5 ARQUITETURA DYCORS: EXPLORANDO COMPOSIÇÃO DINÂMICA DE ESTRATÉGIAS PARA RACIOCÍNIO DE CONTEXTO 54
5.1 Organização Arquitetural da DyCoRS 54
5.1.1 Componente de Comunicação com o Meio 56
5.1.2 Componente de Pré-processamento 57
5.1.3 Componente de Raciocínio 57
5.1.4 Componente de Armazenamento 61
5.1.5 Componente de Correlação 63
5.1.6 Componente de Reconfiguração 66
5.1.7 Componente de Disseminação 68
5.2 Análise Comparativa com Trabalhos Relacionados 68
5.3 Considerações do Capítulo 70

6 AVALIAÇÃO E DISCUSSÃO DOS RESULTADOS 71
6.1 Principais Ferramentas Utilizadas 71
6.2 Cenários de Uso 73
6.2.1 Cenário 1 - Computação Aplicada à Saúde 73
6.2.2 Cenário 2 - Segurança da Informação 79
6.2.3 Cenário 3 - Ambiente de Vivência Assistida 83
6.3 Considerações do Capítulo 86

7 CONSIDERAÇÕES FINAIS 87
7.1 Trabalhos Futuros 89
7.2 Publicações 90

REFERÊNCIAS 95

1 INTRODUÇÃO

A Computação Ubíqua (UbiComp), é um paradigma no qual os computadores são disponibilizados para a necessidade humana, tendo como premissa envolver o mínimo possível o usuário com a gerência da infraestrutura computacional (OCEGUEDA-MIRAMONTES; SANCHEZ; AGUILAR, 2019). Assim, a computação ubíqua, desde quando foi proposta no artigo seminal de Weiser (1991), vem sistematizando abordagens para integrar a computação com o mundo físico, da forma mais transparente possível.

Os últimos 20 anos têm avançado consideravelmente neste sentido com a proliferação de dispositivos conectados com diferentes perfis operacionais, e em uma escala que prevê atingir em 2020 um total de 50 bilhões (COMMISSION, 2015). Estes avanços têm inspirado pesquisas em UbiComp nas áreas de Interação Humano-Computador, Inteligência Artificial, Inteligência Ambiental, Internet das Coisas (IoT) e Sistemas Cibernéticos (WANG; ONG; NEE, 2017).

Sistemas ubíquos se tornaram uma parte importante, e até muitas vezes essencial, na vida diária moderna. Casas inteligentes são bons exemplos nos quais tais sistemas podem ser encontrados. No entanto, o projeto e a implementação de aplicações ubíquas são tarefas difíceis, pois envolvem várias áreas da computação, como engenharia de software, inteligência artificial e sistemas distribuídos (SÁNCHEZ GUINEA; NAIN; LE TRAON, 2016).

Devido ao rápido desenvolvimento da IoT enquanto uma materialização da UbiComp e sua utilização em diferentes aspectos da vida cotidiana, observa-se um aumento na quantidade de dados de contexto que continuamente são gerados, os quais representam o estado do ambiente ao longo do tempo (SEZER; DOGDU; OZBAYOGLU, 2018). Acredita-se que a utilização efetiva deste grande volume de dados de contexto pode introduzir novas possibilidades para o desenvolvimento humano, com base em soluções computacionais que sejam autônomas, responsivas e adaptativas ao contexto (LI et al., 2015).

Desta forma, a ciência de contexto torna-se um conceito chave para fornecer serviços adaptáveis, por exemplo, quando os serviços mais adequados são selecionados

de acordo com as informações de contexto relevantes para o usuário, ou quando é necessário alterar os parâmetros operacionais dos serviços, em função do contexto, durante sua execução. Deste modo, as aplicações cientes de contexto devem ser capazes de adaptar seus comportamentos em mudança com o mínimo de intervenção humana, e para tanto introduzem desafios diversos para os seus desenvolvedores (TEMDEE; PRASAD, 2018).

Os sistemas cientes de contexto na sua operação envolvem uma grande quantidade de informações de contexto que precisam ser: recuperadas constantemente; efetivamente interpretadas; rapidamente processadas; divulgadas às aplicações interessadas; e mantidas em repositórios de contexto (BELLAVISTA et al., 2012). Portanto, para minimizar os esforços de desenvolvimento deste tipo de aplicação, precisam ser adotados mecanismos robustos para gerenciamento de contexto, bem como esquemas de representação de contexto que possibilitem o tratamento da heterogeneidade dos dados envolvidos.

Investigar o gerenciamento de contexto, em termos de abordagens para modelagem de informações de contexto e estratégias de raciocínio para processamento das informações de contextuais, constitui parte de um significativo esforço de pesquisa. Além disso, é necessário investigar o emprego da ciência de contexto para o desenvolvimento de aplicações que são adaptáveis segundo os interesses definidos pelos usuários (BIBRI, 2015).

Estratégias para raciocínio de contexto vêm sendo desenvolvidas, mas mesmo as mais recentes, não se mostram versáteis para atender os requisitos de raciocínio necessários para diferentes aplicações ubíquas, no cenário da IoT (LI et al., 2015). Assim, identifica-se como uma possível solução a combinação de diferentes estratégias para raciocínio de contexto, constituindo assim, as abordagens híbridas (BETTINI et al., 2010).

O uso de abordagens híbridas para raciocínio é uma área de pesquisa em crescimento na comunidade devido a sua possibilidade de maior precisão preditiva (SUBBARAJ; VENKATARAMAN, 2016). Estas abordagens fornecem mecanismos adequados para a solução baseada em computador considerando problemas complexos, como os que são enfrentados pelas aplicações ubíquas (CABRERA; EDYE, 2010).

Assim, apesar de existirem diversas propostas híbridas na literatura e elas abordarem diferentes frentes, nota-se que ainda são necessários esforços de pesquisa relacionados ao tema. Principalmente, tendo como foco o desenvolvimento de abordagens que sejam dinâmicas, de forma que as aplicações possam escolher como querem utilizar os diferentes recursos disponíveis. Ainda, com o desenvolvimento dessas abordagens as aplicações podem escolher se estes diferentes recursos serão utilizados de forma individual ou combinada, tudo isso dependendo dos contextos que serão tratados.

1.1 Questão de Pesquisa

A partir das motivações centrais da área de ciência de contexto, identificadas a partir de uma revisão de literatura, foi concebida a questão descrita a seguir, a qual norteou os esforços de estudo e pesquisa desenvolvidos nesta Tese.

Como deve ser uma proposta para ciência de contexto que viabilize uma combinação dinâmica de diferentes estratégias para raciocínio?

Esta Tese tem como convicção que uma arquitetura de software para ciência de contexto que explore uma metalinguagem para criação de regras para processamento contextual, concomitantemente com a disponibilidade dos mecanismos necessários para prover uma abordagem híbrida para raciocínio de contexto, poderá viabilizar uma combinação dinâmica de diferentes estratégias, atendendo assim, com uma mesma arquitetura, aplicações com diferentes demandas de processamento contextual.

1.2 Objetivos

O objetivo geral desta Tese é a concepção de uma arquitetura direcionada ao provimento de ciência de contexto para as aplicações ubíquas, intitulada de *Dynamic Composition of Reasoning Strategies* (DyCoRS), que seja dinâmica no oferecimento dos recursos necessários para obtenção da ciência de contexto. Dentre os recursos oferecidos destaca-se uma abordagem híbrida para raciocínio de contexto, a qual deve permitir uma composição dinâmica de diferentes estratégias. Além disso, a arquitetura ainda deverá prover: (i) diferentes modelos de armazenamento; (ii) um método para correlação das informações persistidas nos diversos modelos; e (iii) um processo que permita a reconfiguração dos recursos providos pela arquitetura em tempo de execução.

Na perspectiva de atender o objetivo geral previsto, as seguintes metas foram sistematizadas, tendo como perspectiva a consolidação da abordagem de estudo e pesquisa a ser desenvolvida, a qual contemplou um crescimento gradativo de especificidade, e que culminou com a disseminação do trabalho em veículos qualificados.

- revisar, na perspectiva da computação ubíqua, os principais conceitos relacionados à ciência de contexto;
- identificar o estado da arte em abordagens híbridas para raciocínio de contexto, empregando uma Revisão Sistemática de Literatura;
- conceber uma arquitetura para suporte à ciência de contexto que possa ser empregada em diferentes domínios de aplicação;
- especificar componentes e funcionalidades a serem providos pela arquitetura;

- conceber uma estratégia composicional para raciocínio de contexto, a qual possua como diferencial a possibilidade de combinar diferentes estratégias de raciocínio;
- prototipar cenários de uso que demonstrem a utilização da arquitetura proposta;
- divulgar ante a comunidade científica os resultados atingidos pela pesquisa por meio de publicações em conferências e/ou periódicos especializados da área.

1.3 Estrutura do Texto

O documento desta Tese está organizado em sete Capítulos, partindo de aspectos gerais da área de Ciência de Contexto, e evoluindo para aqueles mais específicos no tópico de pesquisa desenvolvido.

No Capítulo 2 são sistematizados os principais conceitos de ciência de contexto, sendo identificadas as principais etapas envolvidas no gerenciamento das informações contextuais. Particularmente, é dada ênfase na discussão das etapas de aquisição, modelagem, processamento e disseminação de contexto.

Uma revisão sobre estratégias para raciocínio de contexto é tratada no Capítulo 3, mostrando suas principais características e vantagens na utilização de cada estratégia. É feita uma caracterização de uso das estratégias híbridas, as quais buscam combinar duas ou mais estratégias de forma a se beneficiar dos pontos fortes de cada uma delas.

No Capítulo 4 é sumarizado o estado da arte em raciocínio híbrido de contexto, sendo explorada para isso uma Revisão Sistemática de Literatura. Desta forma, são apresentados o fluxo de execução da revisão realizada, as principais características dos trabalhos selecionados e uma análise comparativa entre estes.

A arquitetura DyCoRS é especificada no Capítulo 5, sendo discutidas as suas principais características, componentes arquiteturais e funcionalidades oferecidas para a obtenção da ciência de contexto.

No Capítulo 6 é apresentado o método de avaliação da DyCoRS, sendo contemplados três cenários de uso, explorando diferentes funcionalidades oferecidas pela arquitetura concebida. Ainda, são apresentadas as principais tecnologias utilizadas na prototipação da DyCoRS.

Por último, no Capítulo 7 são apresentadas as considerações finais, destacando as principais contribuições observadas até o momento, as possibilidades de trabalhos futuros a serem desenvolvidos, bem como as publicações realizadas ao longo do curso do doutorado.

2 CIÊNCIA DE CONTEXTO NA UBICOMP

UbiComp é o termo utilizado para a terceira era da computação moderna, é a fase que se está vivendo nos dias de hoje, na qual há uma variedade de dispositivos eletrônicos com diferentes perfis computacionais, que fornecem conjuntamente serviços para o usuário (SÁNCHEZ GUINEA; NAIN; LE TRAON, 2016). A UbiComp aumenta o uso do computador, tornando os computadores disponíveis de forma distribuída em todo o ambiente físico, ao mesmo tempo, com a perspectiva de torná-los invisíveis para o usuário (MACHADO et al., 2017).

De uma forma geral, a UbiComp pode ser conceituada como sendo um ambiente repleto de dispositivos que se comunicam entre si, e que podem muitas vezes serem carregados pelo usuário, na perspectiva da computação móvel. O objetivo geral da Ubicomp é melhorar a experiência dos usuários, oferecendo algum tipo de aplicação e/ou serviço de forma transparente (PITTOLI, 2015).

Dentre as diversas demandas de um ambiente ubíquo para oferecer serviços de forma transparente, está a ciência ao contexto (TEMDEE; PRASAD, 2018). A qual pode ser vista como a capacidade de (COSTA, 2008): (i) coletar informações relevantes para o contexto de interesse das aplicações; (ii) armazenar os dados coletados em um histórico que pode ser utilizado para estabelecer tendências sobre os valores de informações do contexto; (iii) direcionar ações e comportamentos que interfiram no estado do ambiente do usuário.

Contexto é qualquer informação que possa usada para caracterizar a situação de uma entidade (pessoa, local ou objeto) considerada relevante para a interação entre o usuário e a aplicação, incluindo o próprio usuário e a aplicação (DEY, 2001; KHATTAK et al., 2014). Enquanto que a ciência de contexto é a capacidade de um sistema em usar o contexto para prover serviços e/ou informações relevantes para o usuário (TEMDEE; PRASAD, 2018). Os sistemas cientes de contexto devem ser flexíveis, adaptativos e capazes de atuar automaticamente para ajudar o usuário na realização de suas atividades (MACHADO, 2010).

As áreas da computação ubíqua e inteligência artificial foram as pioneiras em estudar e utilizar o conceito de contexto, demonstrando o potencial da aplicação desse

conceito nos sistemas computacionais com o intuito de aumentar a transparência de uso. Pesquisas recentes vêm utilizando o conceito de contexto para beneficiar sistemas ligados a outras áreas tais como:

- sistemas colaborativos: utilização de contexto para melhorar a interação e produtividade do grupo, aplicado por exemplo, na área médica;
- hipermídia adaptativa: possibilidade de personalização e adaptação do conteúdo de sites Web a partir de contextos;
- integração de dados: facilitando a resolução de conflitos semânticos com a utilização de contextos;
- interação humano-computador: o contexto é utilizado para adaptar as interfaces dos sistemas tornando mais intuitiva a sua interação com os usuários.

Algumas motivações para aplicação de ciência de contexto nos sistemas computacionais são: auxiliar na compreensão da realidade; facilitar a adaptação de sistemas; contribuir no processo de transformação dos dados em informação; apoiar a compreensão de eventos; ajudar a identificar situações de interesse.

Há uma série de funcionalidades que devem ser providas de forma a obter a ciência de contexto, dentre elas pode-se destacar (SAHU; ADANE, 2015):

- suportar a manipulação de uma variedade de tipos de sensores;
- lidar com a natureza distribuída da informação de contexto, porque os dados vêm de fontes diferentes e de naturezas de dados diferentes;
- proporcionar uma interpretação transparente para as aplicações e uma abstração para os dados de contexto;
- disponibilizar o armazenamento dos dados de contexto e realizar a manutenção desse armazenamento;
- controlar o fluxo de dados do contexto.

De forma simplificada, a construção de um sistema ciente de contexto se dá por meio de quatro etapas, sendo elas: (i) aquisição; (ii) modelagem; (iii) processamento; (iv) disseminação. A Figura 1 apresenta uma visão do processo envolvido para a obtenção da ciência de contexto,

Primeiramente pode ser visualizada a etapa de aquisição, a qual se comunica com os diversos tipos de sensores, capturando os dados contextuais. Após, pode ser visualizada a etapa de modelagem, a qual também é intitulada de representação de contexto, estando diretamente ligada às operações das outras etapas da ciência de

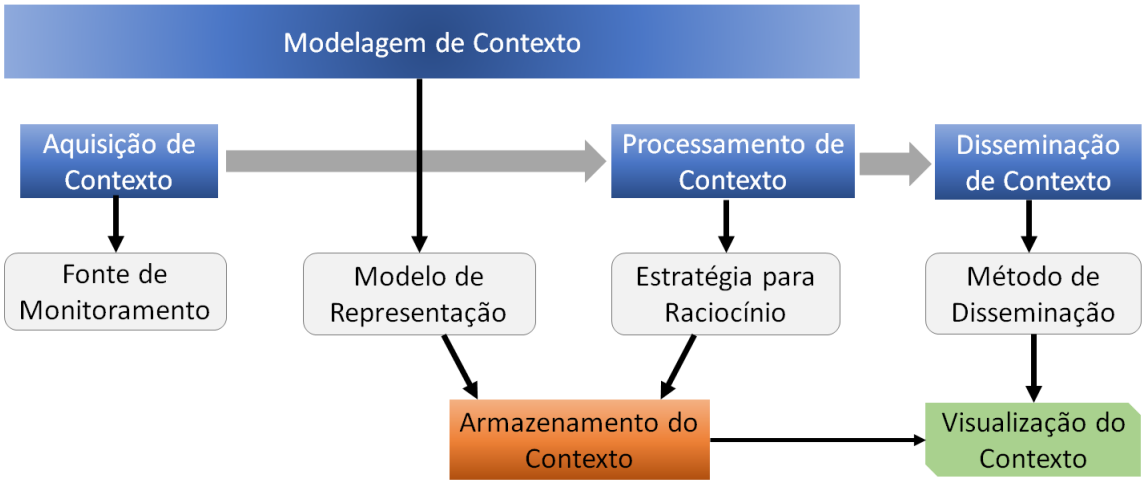

Figura 1 – Visão das etapas envolvidas para a obtenção da ciência de contexto.
Fonte: Adaptado de LI et al., 2015, p. 20577.

contexto, sendo utilizada, por exemplo, como base para a definição do modelo de armazenamento que será empregado.

Na sequência está a etapa de processamento, a qual processa os dados de contexto com o intuito de realizar alguma inferência nos mesmos, de forma a identificar determinadas situações que podem gerar uma ação a ser realizada. E por último, é apresentada a etapa de disseminação que possibilita as informações contextuais para as aplicações cientes de contexto.

Além das quatro etapas que podem ser visualizadas na parte superior da Figura 1, destaca-se as camadas de armazenamento e visualização. A primeira é responsável por realizar o armazenamento dos dados de contexto adquiridos, bem como das informações contextuais que foram inferidas pela etapa de processamento. A segunda é encarregada por disponibilizar métodos de visualização dos dados de forma a facilitar o acesso, pelas aplicações cientes de contexto, aos dados capturados e situações detectadas.

2.1 Aquisição de Contexto

A aquisição do contexto refere-se ao processo de monitorar, capturar e/ou obter informações contextuais. A etapa de aquisição visa abstrair das aplicações cientes de contexto a complexidade da coleta de dados, além de possibilitar a reutilização de sensores e a separação entre obtenção e utilização das informações contextuais (ALEGRE; AUGUSTO; CLARK, 2016).

A informação contextual pode ser adquirida a partir de diversas fontes, tais como (VIEIRA et al., 2006):

- sensores físicos: são o tipo mais comum de sensores, tangíveis e geram dados por si só. Os dados obtidos a partir destes sensores são chamados de con-

texto de baixo nível. Possuem menor significância e apresentam vulnerabilidade a pequenas mudanças. São responsáveis pela captura de informações sobre o ambiente físico, tais como, luz, temperatura, umidade, gás. Alguns pontos positivos são: a detecção de erros e a possibilidade de identificar falta de valores de forma relativamente fácil; ser eficiente, devido a ter acesso às configurações do sensor de baixo nível. Como pontos negativos, destacam-se a implantação e manutenção de hardware, que pode ser custosa e a necessidade de tratar com sensores e programação de baixo nível;

- sensores virtuais: estes sensores recuperam dados de outras fontes e publicam como dados de sensores. Podem ser utilizados para coletar informações que não podem ser medidas fisicamente, tais como, detalhes de calendário, e-mail, bate-papo, mapas, dados de redes sociais, preferências do usuário. Alguns pontos positivos são, o fornecimento de dados significativos e informações de contexto de alto nível, a possibilidade de não precisar lidar com tarefas de nível de hardware. Como pontos negativos destacam-se a dificuldade de encontrar erros nos dados e a dificuldade para o preenchimento de valores ausentes, já que em sua maioria são dados não numéricos e imprevisíveis;

- sensores lógicos: combinam sensores físicos e sensores virtuais, a fim de produzir informações mais significativas. Estes sensores são utilizados para coletar informações que não podem ser coletadas diretamente através de um único sensor físico, realizando a combinação de dados de vários sensores, como por exemplo, informações sobre o tempo, o reconhecimento de atividade, reconhecimento de localização. Alguns pontos positivos são: o fornecimento de dados altamente significativos; o provimento de informações de contexto de alto nível e mais precisas; a possibilidade de não precisar lidar com tarefas de nível de hardware. Como pontos negativos destacam-se a dificuldade de encontrar erros em dados e a falta de controle sobre processo de produção dos dados.

As informações contextuais podem ser classificadas das seguintes formas (HENRICKSEN; INDULSKA, 2006):

- percebidas: são provenientes de sensores físicos ou lógicos, possuem uma baixa persistência. As informações podem ser imprecisas, desconhecidas ou caducas, e suas fontes de imperfeição são erros na percepção, falhas do sensor ou desconexões da rede;

- provenientes de perfil: são fornecidas pelo próprio usuário, possuem uma persistência moderada. As informações tendem a caducar, e suas fontes de imperfeição são a omissão do usuário em atualizar mudanças ocorridas;

- derivadas: são aquelas obtidas por meio de mecanismos de derivação, possuem uma persistência variável. As informações estão sujeitas a erros e imperfeições, suas fontes de imperfeição são entradas imprecisas e mecanismo de derivação imaturo ou simplificado.

É importante que a tarefa de aquisição de contexto permaneça em constante execução, e é desejável que seja implementada de forma independente das aplicações que a utilizem, possibilitando assim que diversas aplicações possam fazer uso das mesmas informações contextuais.

2.2 Modelagem de Contexto

Um sistema ciente de contexto requer que informações contextuais sejam trocadas e utilizadas por diferentes entidades, como agentes humanos e de software, dispositivos e serviços. A quantidade de informação capturada e acessada num modelo de representação do contexto é significativa, com isso, os estudos de técnicas para representação de informações contextuais vêm se tornando cada vez mais importantes (KHATTAK et al., 2014).

O processo de modelagem de contexto consiste na concepção de um modelo de entidades do mundo real, suas propriedades, estado de seu ambiente e situações que podem ser usados como referência para a aquisição, interpretação e raciocínio de informações contextuais (KNAPPMEYER et al., 2013). Um bom formalismo para modelagem de contexto reduz a complexidade das aplicações cientes de contexto, facilita o acesso às informações realizando buscas de forma eficiente, melhora a capacidade de manutenção e de evolução da aplicação (STRANG; LINNHOFF-POPIEN, 2004). Alguns requisitos que devem ser levados em consideração durante a modelagem das informações de contexto são heterogeneidade, mobilidade, dependências e relações entre os dados.

Existem diversos modelos para representação do contexto, tais como (PERERA et al., 2014):

- chave-valor: esta abordagem utiliza a estrutura de dados mais simples para representar a informação contextual. O contexto é representado através de pares compostos por uma chave, responsável por identificar o atributo de contexto, e por um valor associado a essa chave. É um modelo simples de utilizar e manipular, podendo ser utilizada para modelar quantidade limitada de dados. O modelo chave-valor não permite estruturações mais sofisticadas que habilitem algoritmos eficientes de recuperação de contexto. A modelagem deste tipo não é escalável e não é adequada para armazenar estruturas de dados complexas;

- linguagem de marcação: este modelo utiliza estruturas de dados hierárquicas, consistindo de *tags* com atributos e conteúdo para representar o contexto, sendo considerada uma melhoria sobre o modelo chave-valor. A vantagem de usar *tags* de marcação é que estas permitem a recuperação de dados de forma eficiente. Em contraste, as linguagens de marcação não oferecem recursos avançados de expressão que permitam raciocínio de informações contextuais. Esta abordagem pode ser utilizada como um formato intermediário de organização de dados, bem como um modo de transferência de dados;

- gráfico: os modelos gráficos possibilitam um melhor entendimento da estrutura das informações contextuais, mas possuem um baixo grau de formalismo. Este modelo pode ser utilizado para armazenamento em longo prazo de grandes volumes de dados, e os contextos históricos podem ser armazenados em bancos de dados;

- orientado a objetos: essa abordagem modela os dados utilizando hierarquias e relações entre classes, explorando os benefícios do paradigma de programação orientado a objetos, como encapsulamento e reusabilidade. Os detalhes de processamento de contexto são encapsulados no nível de objetos e o acesso às informações de contexto é realizado apenas por meio de interfaces. Os modelos orientados a objetos são fortes em relação à possibilidade de composição distribuída, pois novos tipos de informações de contexto podem ser adicionados e instâncias podem ser atualizadas de forma distribuída. Porém, normalmente, as infraestruturas de execução que suportam esses modelos exigem muitos recursos dos dispositivos computacionais, o que nem sempre pode ser atendido. Além disso, esta estratégia não fornece capacidade de raciocínio de contexto e a avaliação de projetos é dificultada devido à falta de normas e especificações;

- lógico: os modelos baseados em lógica representam as informações contextuais como fatos, expressões e regras. Além disso, um processo de inferência pode ser utilizado para derivar novos fatos com base nas regras modeladas, com isso a informação contextual é adicionada, atualizada e removida em termos de fatos ou inferências a partir de regras. Essa estratégia possui um alto grau de formalismo, no entanto, a falta de normalização reduz a possibilidade de reutilização e aplicabilidade, podendo implicar em uma maior dificuldade de manutenção e compreensão. A modelagem baseada em lógica permite que novas informações de contexto de alto nível possam ser extraídas utilizando informações de contexto de baixo nível, adicionando assim a capacidade de melhorar outras técnicas de modelagem de contexto;

- ontológico: nesta abordagem, o contexto é organizado em ontologias. Uma on-

tologia é uma especificação explícita, formal, de uma conceitualização compartilhada em que objetos, conceitos, entidades e relacionamentos do mundo real são definidos em uma determinada área de interesse ou domínio de conhecimento (GRUBER, 1993). Um dos grandes interesses na construção e uso de ontologias é tornar o conhecimento sobre o mundo real processável por máquinas. Uma das vantagens da utilização das ontologias é a possibilidade de raciocínio lógico, o qual pode ser utilizado pelas aplicações para inferir contextos de alto nível a partir de contextos de baixo nível e para checar e resolver inconsistências no conhecimento contextual. Como desvantagens, pode-se destacar a complexidade que a representação pode alcançar e a grande utilização de recursos computacionais necessários para realizar a recuperação das informações.

Visando realizar uma comparação entre os modelos de contexto mais utilizados, os autores em Strang; Linnhoff-Popien (2004) propuseram uma série de métricas que podem ser avaliadas:

- composição distribuída (dc): a composição e administração dos modelos de contexto devem ser extremamente dinâmicas em termos do tempo, topologia da rede e recursos;
- validação parcial (pv): capacidade para validar conhecimento parcial. Em determinados momentos, devido a composição distribuída, não é possível validar todo o conhecimento de contexto;
- qualidade da informação (qua): a qualidade da informação muda de acordo com o sensor utilizado, entre outros fatores. Os métodos devem suportar o tratamento de informação com distintos níveis de qualidade;
- incompleteza e ambiguidade (inc): o método deve ser capaz de tratar informações incompletas e ambíguas;
- nível de formalidade (for): modelos com sintaxe e semântica bem definidas;
- aplicabilidade em ambientes já existentes (app): utilização de modelos em aplicações já existentes.

A Tabela 1 apresentada uma comparação entre os modelos de contexto apresentados, na qual o sinal de (-) significa limitação do modelo, (- -) limitação ainda maior, o sinal de (+) o atendimento da métrica e o (++) o atendimento de uma forma mais satisfatória.

Como pode ser observado na Tabela 1, cada modelo possui vantagens e desvantagens em relação a sua utilização, sendo que ainda não foi encontrada um modelo que

Tabela 1 – Comparação entre abordagens para modelagem de contexto.

Modelo	dc	pv	qua	inc	for	app
Chave-valor	-	-	- -	- -	- -	+
Linguagem de Marcação	+	+ +	-	-	+	++
Gráfico	- -	-	+	-	+	+
Orientado a Objetos	+ +	+	+	+	+	+
Lógico	+ +	-	-	-	+ +	- -
Ontológico	+ +	+ +	+	+	+ +	+

Fonte: Adaptado de STRANG; LINNHOFF-POPIEN, 2004, p. 6.

seja considerado ideal, com isso, têm surgido as abordagens híbridas. Estas abordagens são consideradas as mais promissoras, pois combinam diferentes modelos de representação, com diferentes níveis de interpretação, para diferentes aspectos.

2.3 Processamento de Contexto

Após a etapa de modelagem de contexto há uma necessidade de criar novos conhecimentos e ter uma melhor compreensão dos contextos adquiridos, essa necessidade é atendida na etapa de Processamento de Contexto. Essa etapa pode ser apresentada como um processo de realizar deduções de contexto de alto nível a um conjunto de contextos de baixo nível. Também podendo ser definida como um mecanismo de raciocínio para inferir novos conhecimentos e melhorar a compreensão dos contextos adquiridos (PERERA et al., 2014).

O processamento de contexto pode ser dividido em três fases (SAHU; ADANE, 2015):

- pré-processamento de contexto: essa fase é responsável por realizar a normalização dos dados coletados na etapa de aquisição de contexto. Devido a ineficiências no hardware do sensor e na comunicação de rede, os dados coletados podem não estar precisos ou conter falhas. Portanto, os dados podem precisar ser normalizados, sendo realizado o preenchimento de valores faltantes ou transformações nos dados para um formato padrão, desta forma facilitando o raciocínio destes dados;

- fusão de dados de sensores: nessa fase é realizado a combinação de dados de contexto de múltiplos sensores para produzir informações mais precisas, completas e confiáveis, as quais não poderiam ser alcançadas através de um único sensor. Na IoT, a fusão é extremamente importante, porque poderá haver uma variedade de sensores disponíveis, existindo um grande número de fontes alter-

nativas para fornecer a mesma informação;

- raciocínio de contexto: essa fase é responsável pela geração de informações de contexto de alto nível usando contexto de nível inferior. Após o raciocínio de contexto é possível identificar situações de interesse e prover serviços para as aplicações cientes de contexto.

Um número significativo de mecanismos provenientes dos campos da inteligência artificial e com base em sistemas de conhecimento podem ser adotados para realizar a fase de raciocínio de contexto. Dada a importância da fase de raciocínio para a obtenção da ciência de contexto e a diversidade de estratégias que podem ser empregadas nessa fase, no Capítulo 3 são apresentadas algumas das principais estratégias utilizadas, sendo destacadas as suas principais características.

2.4 Disseminação de Contexto

A disseminação de contexto é responsável por fornecer métodos para entregar as informações contextuais às aplicações cientes de contexto. Mais formalmente, define-se a disseminação de dados de contexto como a etapa que possibilita a injeção de contexto no sistema ciente de contexto e sua entrega a todas as entidades que expressaram qualquer forma de interesse nesses dados (BELLAVISTA et al., 2012). A disseminação dos dados de contexto pode ser classificada de duas formas, dependendo do método utilizado, sendo elas (ALEGRE; AUGUSTO; CLARK, 2016):

- consulta: uma aplicação ciente de contexto faz um pedido em termos de uma consulta, de modo que o sistema de gerenciamento de contexto possa a partir dessa consulta localizar as informações contextuais processadas e disseminar os resultados para a aplicação que fez a requisição;

- subscrição (também chamado *publish/subscribe*): uma aplicação ciente de contexto, devidamente autorizada, se inscreve no sistema de gerenciamento de contexto descrevendo os requisitos que devem ser atendidos para que ocorra uma disseminação. Assim, as aplicações podem se inscrever para receber informações referentes a um sensor específico ou determinados eventos de interesse decorrentes do processamento contextual.

Para suportar de forma eficaz os serviços cientes de contexto a disseminação de dados de contexto tem de cumprir alguns requisitos principais (BELLAVISTA et al., 2012):

- desacoplamento entre produção/consumo dos dados de contexto: a disseminação de dados de contexto tem de encaminhar de forma transparente os dados de contexto produzidos para todas as aplicações interessadas;

- adaptação a ambientes heterogêneos: a disseminação de dados de contexto tem de suportar cenários heterogêneos, incluindo nós com diferentes capacidades computacionais;

- definição de escopos de visibilidade dos dados: a disseminação de dados de contexto tem de introduzir, preservar e impor marcos de visibilidade diferenciados para cada dado, já que os dados de contexto intrinsecamente têm escopos de visibilidade que a etapa de disseminação deve respeitar.

2.5 Considerações do Capítulo

A revisão da literatura na área apontou que a quase totalidade dos trabalhos desenvolvidos organiza seus esforços de pesquisa em quatro grandes eixos: aquisição; modelagem; processamento; e disseminação de contexto.

Para etapa de aquisição, ficou caracterizada a importância de que uma plataforma para ciência de contexto trate com dados provenientes de diferentes fontes, cada um com sua natureza especifica.

Destaca-se na etapa de modelagem a utilização de abordagens baseadas em ontologias, devido a sua capacidade de representação semântica, e sua elevada possibilidade de raciocínio, podendo inferir novos conhecimentos com base nas informações presentes na ontologia. Apesar da disseminação do uso de ontologias, percebe-se que as mesmas também possuem pontos negativos, por exemplo o desempenho computacional, com isso as abordagens híbridas vêm ganhando destaque, pois combinam os pontos positivos de cada modelo (KOTENKO et al., 2013; KOÇ et al., 2014; LI et al., 2015).

Particularmente, para a etapa de processamento de contexto destaca-se na literatura a fase de raciocínio, para qual a utilização de diversas estratégias vêm sendo discutida. Desta forma, no próximo Capítulo são apresentadas as principais estratégias utilizadas.

3 UMA REVISÃO DE ESTRATÉGIAS PARA RACIOCÍNIO DE CONTEXTO

A revisão da literatura aponta que as diferentes estratégias para raciocínio de contexto possuem vantagens e desvantagens considerando os diferentes domínios de aplicação. Desta forma, diferentes aplicações irão demandar estratégias diversas para implementar o raciocínio de contexto. Assim, a seguir são apresentadas algumas das principais estratégias utilizadas para realizar o raciocínio de contexto, tendo como referência os trabalhos (KNAPPMEYER et al., 2013; PERERA et al., 2014; LI et al., 2015; SEZER; DOGDU; OZBAYOGLU, 2018; TEMDEE; PRASAD, 2018).

3.1 Baseadas em Aprendizagem não Supervisionada

As estratégias baseadas em aprendizagem não supervisionada utilizam um conjunto de treinamento para aprender, mas estes dados não estão categorizados, ou seja, eles não possuem o resultado esperado (YAN et al., 2017). As técnicas desta estratégia analisam os dados de entrada e tentam encontrar padrões nos dados utilizados. A estratégia não supervisionada é prática quando o objetivo é encontrar regularidade nos dados, os quais não foram previamente reconhecidos (KÄMPPÄ, 2016).

As situações a serem identificadas podem ser aprendidas em tempo de execução, sendo uma estratégia adequada para os casos em que as situações dependem fortemente do usuário e um treinamento prévio é impraticável. Nesta categoria destacam-se as técnicas de agrupamento, como a técnica k Vizinhos Mais Próximo.

As técnicas utilizadas nesta categoria costumam ser utilizadas em redes de sensores, para realizar tarefas como de roteamento e também nas operações de posicionamento e localização. Possuem destaque por conseguirem aprender sem precisar de um conjunto de treinamento com as respostas esperadas. Como pontos negativos, destacam-se a dificuldade em realizar a validação, a complexidade que os modelos podem alcançar, a imprevisibilidade dos resultados, e que a utilização de recursos computacionais pode se tornar intensivo (PERERA et al., 2014).

3.2 Baseadas em Aprendizagem Supervisionada

As estratégias baseadas em aprendizagem supervisionada utilizam um conjunto de treinamento, no qual os dados se encontram categorizados (KHAN et al., 2017). Após o treinamento é possível classificar novos eventos com base no conjunto que foi utilizado durante o treinamento.

Na avaliação da aprendizagem supervisionada, os dados de treinamento utilizados para predizer novos valores podem ser divididos em dois ou mais segmentos de dados atribuídos aleatoriamente (SARWAR et al., 2001). Um segmento é sempre reservado como dados de teste, para avaliar a técnica utilizada em relação ao desempenho do conjunto de treinamento. Esses segmentos de dados precisam ser grandes o suficiente para conter representantes de todas as variações dos dados.

A fim de evitar a sobre adaptação e inconsistências dos dados, podem ser usados alguns métodos de validação, entre eles destaca-se a validação cruzada. Neste método os dados da amostra são divididos em K-segmentos atribuídos aleatoriamente. Um segmento é reservado como dados de teste e outros são usados para treinar a técnica empregada. O desempenho dos segmentos K-1 remanescentes é medido em relação ao conjunto de teste, sendo obtida a média das pontuações (KÄMPPÄ, 2016).

Dentre as técnicas desta categoria destaca-se a Árvore de Decisão, na qual é construída uma árvore a partir de um conjunto de dados, podendo utiliza-lá para classificar os novos eventos capturados. Outra técnica que se destaca na categoria de aprendizagem supervisionada são as Máquinas de Vetores de Suporte, as quais são utilizadas para reconhecimento de padrões em computação ciente de contexto.

Dentre os pontos positivos das técnicas dessa categoria destaca-se que costumam alcançar um alto grau de precisão e que possuem uma base matemática e estatística. Em relação as dificuldades encontradas na utilização destas técnicas, pode-se citar a exigência de quantidade significativa de dados para treinamento, a possibilidade de uma maior utilização de recursos computacionais, como processamento e armazenamento, e a necessidade de selecionar os dados que serão utilizados, com o intuito de não tornar o processamento custoso (PERERA et al., 2014).

3.3 Baseadas em Casos

O raciocínio baseado em casos é uma estratégia que usa o conhecimento específico de um problema concreto previamente experimentado ao resolver um problema recentemente identificado (PLA et al., 2014). O novo problema é resolvido por encontrar um caso passado semelhante e reutilizá-lo no tratamento do novo problema.

Uma característica importante do raciocínio baseado em casos é a sua capacidade de aprendizagem incremental a partir de uma nova experiência, que é retida cada vez que um problema é resolvido, tornando-o imediatamente disponível para problemas

futuros (AAMODT; PLAZA, 1994). Outra vantagem importante da estratégia baseada em casos é que ela pode ser usada como uma metodologia e não somente como uma implementação concreta da tecnologia para resolver um problema (WATSON, 1999).

A estratégia baseada em casos geralmente segue quatro métodos cíclicos importantes para a resolução de problemas: (i) recuperar casos semelhantes às descrições do problema; (ii) reutilizar a solução sugerida pelos casos similares; (iii) revisar ou adaptar essa solução para atender o novo problema, se necessário; (iv) reter a nova solução assim que ela for confirmada ou validada.

3.4 Baseadas em Lógica Descritiva

As estratégias baseadas em lógica descritiva são aplicadas em conjunto com a representação de contexto ontológico. A modelagem semântica de conceitos (classes), papéis (propriedades e relações) e indivíduos permite que o conhecimento a ser especificado seja interpretável por máquina (GRUBER, 1993). O raciocínio baseado em ontologias é computacionalmente intensivo e o tempo de resposta dependerá em grande parte do tamanho do conjunto de dados e do conjunto de regras presentes na ontologia. Além da complexidade de raciocínio, a concepção de uma ontologia é dita ser uma tarefa complexa, que exige conhecimentos de domínio (GUIZZARDI, 2007).

O raciocínio baseado em ontologias pode ser utilizado de duas formas, na primeira são utilizados axiomas internos a ontologia, nos quais são definidas regras que inferem conhecimento sobre as classes e hierarquias que são representadas na ontologia. Na segunda forma, podem ser utilizadas regras externas escritas em alguma linguagem, como *Semantic Web Rule Language* (SWRL), permitindo inferir novos conhecimentos sobre as instâncias da ontologia.

Como pontos positivos do raciocínio baseado em lógica descritiva destacam-se a possibilidade de raciocínio complexo, a representação complexa, a obtenção de resultados significativos, e, a possibilidade de validação e verificação da qualidade. Dentre os pontos negativos na sua utilização pode-se citar o baixo desempenho, já que podem exigir maior tempo para realizar o processamento, a necessidade de maior poder computacional e que o raciocínio ontológico não é capaz de encontrar valores faltantes ou informações ambíguas (KNAPPMEYER et al., 2013).

3.5 Baseadas em Lógica Fuzzy

As estratégias baseadas em lógica fuzzy permitem um raciocínio aproximado em vez de um valor fixo, podendo raciocinar valores entre 0 e 1 (ALEGRE; AUGUSTO; CLARK, 2016). A lógica fuzzy consiste em quatro processos principais, incluindo fuzzificação, avaliação de regras fuzzy, agregação e defuzzificação (TEMDEE; PRASAD,

2018). Essas estratégias podem ser utilizadas para representar e raciocinar sobre a incerteza ou a imprecisão do contexto.

Pode ser realizada a fusão de dois ou mais conjuntos de elementos para criar um novo conjunto com sua própria função de associação. Exemplos de operações de fusão são a interseção, união, complemento e modificação. A lógica fuzzy é adequada para descrever contextos subjetivos, realizando fusão desses contextos subjetivos e resolvendo possíveis conflitos entre diferentes contextos.

A lógica fuzzy permite conceber conceitos imprecisos, tais como, alto, curto, escuro, confiável, permitindo assim que os cenários do mundo real sejam representados de forma mais natural (PERERA et al., 2014). Na maioria dos casos, o raciocínio baseado em lógica fuzzy deve ser usado em combinação com outra estratégia para realizar o raciocínio de contexto.

3.6 Baseadas em Lógica Probabilística

As estratégias baseadas em lógica probabilística permitem que as decisões sejam tomadas com base em probabilidades associadas aos eventos (RANGANATHAN; ALMUHTADI; CAMPBELL, 2004). Podem ser utilizadas para combinar os dados dos sensores a partir de diversas fontes. Além disso, podem ser usadas para identificar as resoluções de conflitos entre contextos. Na maioria das vezes, estas técnicas são usadas para entender ocorrência de eventos (LI et al., 2015).

Dentre as técnicas utilizadas nesta estratégia pode-se citar a *Naïve Bayes*, que é considerada a estratégia generativa mais simples para a classificação de variáveis de classe com uma única de pendência de valores. E os modelos ocultos de Markov, que representam dados estruturados sequencialmente, permitindo que o estado seja representado usando evidências observáveis sem ler diretamente o estado (PERERA et al., 2014).

O raciocínio probabilístico é especialmente aplicável em ambientes cientes de contexto devido a potenciais falhas temporárias na comunicação com os sensores e a possibilidade de medições imprecisas de sensores físicos. As principais vantagens desta estratégia são a possibilidade de, combinar diferentes eventos, lidar com a incerteza, e fornecer resultados moderadamente significativos. Como pontos negativos destacam-se a dificuldade de só trabalhar com valores numéricos e a necessidade de saber as probabilidades dos eventos.

3.7 Baseadas em Regras

As estratégias baseadas em regras constituem um dos métodos mais simples para realizar o raciocínio de contexto. Permitem a geração de informações de contexto de

alto nível utilizando contextos de baixo nível. Consiste de uma técnica simples para se definir e estender, não exigindo utilização intensiva de recursos computacionais.

O raciocínio baseado em regras é construído a partir de fatos e regras. Os fatos são declarações incondicionais que são assumidas como corretas no momento em que são usadas, sendo consideradas como descrições de uma situação (GILMAN, 2015). Uma coleção de fatos forma uma base de fato, que é usada como entrada de dados para a execução da regra. As regras são normalmente representadas da seguinte forma: *Se* $<$*Condição*$>$ *Então* $<$*Ação*$>$*.* Aqui, a parte $<$*Condição*$>$ (antecedente) representa as condições necessárias para ser verdadeira, e assim, realizar a execução da regra, na qual várias condições podem ser combinadas em um composto por meio de operadores lógicos. A parte $<$*Ação*$>$ (consequente) representa os fatos derivados da regra, ou ações que devem ser executadas (OTEBOLAKU, 2015).

As estratégias baseadas em regras podem ser empregadas com o suporte de um mecanismo de *Complex Event Processing* (CEP), que pode ser definido como sendo um conceito de processamento de eventos que lida com a tarefa de processar vários eventos com o objetivo de identificar os eventos significativos dentro da nuvem de eventos. O CEP aumenta consideravelmente as capacidades da estratégia baseada em regras para executar o raciocínio de contexto (BADII; CROUCH; LALLAH, 2010).

O raciocínio baseado em regras possui alguns pontos negativos, tais como a possibilidade de se tornar confuso e intratável quando se lida com uma base de regras de tamanho considerável, a necessidade de que as regras devem ser definidas manualmente, o que é propenso a erros devido ao trabalho manual, e ainda, a ausência de um mecanismo para realizar a validação e verificação de qualidade. O raciocínio baseado em regras só pode ser aplicado em sistemas de suporte de contexto com propagação baseada em eventos, não suportando imprecisão e somente sendo aplicada para respostas do tipo booleana (KNAPPMEYER et al., 2013).

3.8 Análise das Estratégias para Raciocínio de Contexto

A Tabela 2 apresenta uma análise comparativa entre as diferentes estratégias apresentadas neste capítulo, as quais podem ser utilizadas para realizar o raciocínio de contexto. São identificados os principias pontos positivos e negativos das estratégias, bem como exemplos de aplicabilidade, os quais se beneficiam das características das estratégias analisadas.

A Tabela 2 evidencia a afirmação já feita de que cada estratégia para raciocínio possui pontos fortes e fracos, e que portanto, não existe uma estratégia que possa atender diferentes tipos de aplicações de forma satisfatória. Esta situação vem estimulando o uso de abordagens híbridas para realizar o raciocínio de contexto, as quais utilizam duas ou mais estratégias para raciocínio. Estas abordagens possuem como

Tabela 2 – Comparação entre as estratégias para raciocínio de contexto.

Estratégia	Ponto Positivos	Pontos Negativos	Aplicabilidade
Aprendizagem Não Supervisionada	Aprendizagem sem necessidade da resposta esperada nos dados de treinamento	Difícil de validar Resultado não é previsível	Detecção de Comportamento Incomum Aplicação em redes de sensores
Aprendizagem Supervisionada	Descobre as relações entre entradas e saídas Alcança boas precisões Trata de problemas complexos	Requer uma grande quantidade de dados Dados de treinamento são necessários	Quando o conjunto de recursos é identificável, possíveis saídas são conhecidas e grandes conjuntos de dados estão disponíveis
Casos	Lida com problemas desconhecidos Utilização de conhecimento anterior para resolução de novos problemas	Especificação cuidadosa de quais são os casos e qual é a sua semelhança são necessários	Definição de ações com base em determinada situação Tarefas de diagnóstico
Regras	Definição simples Simples interpretação Facilmente extensível	Pode ser propenso a erros devido ao trabalho manual Manutenção e verificação difíceis	Definição de ações com base em condições Definição de políticas e preferências
Lógica Descritiva	Definições de conceitos completos Dá possibilidades de raciocinar sobre objetos e suas relações	Não é capaz de encontrar valores faltantes ou informações ambíguas Necessidade de poder computacional	Definição de relações Validação e verificação
Lógica Fuzzy	Permite uma representação natural Uso de recursos pouco intensivos Capacidade de lidar com a incerteza	Propenso a erros devido ao trabalho manual Sem verificação de qualidade ou de validação	Para situações em que o contexto precisa ser convertido em informações mais naturais
Lógica Probabilística	Permite combinar evidências Pode lidar com incerteza Resultados moderadamente significativos	É necessário conhecer as probabilidades Trabalha apenas com valores numéricos	Para situações onde as probabilidades são conhecidas Reconhecimento de situação

motivação o aproveitamento dos pontos fortes de cada estratégia e a minimização dos pontos fracos (BIBRI, 2015; LI et al., 2015).

3.9 Considerações do Capítulo

Neste Capítulo, foi buscado um mapeamento de área no que diz respeito as estratégias exploradas para raciocínio de contexto.

Foi constatado no mapeamento realizado que cada estratégia possui pontos fortes e fracos, o que impulsionou o uso das abordagens ditas híbridas. Assim, com o intuito de buscar subsídios para o esforço de pesquisa central da tese, foi realizada uma Revisão Sistemática de Literatura no tema raciocínio híbrido de contexto, a qual é discutida no Capítulo seguinte.

4 ESTADO DA ARTE EM RACIOCÍNIO HÍBRIDO DE CONTEXTO

A Revisão Sistemática de Literatura (RSL) é uma metodologia de pesquisa que prevê a execução de uma série de passos para construir o estado da arte em uma área de pesquisa, ou seja, identificar um conjunto de artigos que traz uma prospecção criteriosa do tema de pesquisa abordado (KITCHENHAM; CHARTERS, 2007). Uma característica desta metodologia de pesquisa é que todos os seus procedimentos são registrados, permitindo que o estudo realizado seja reproduzível por outros pesquisadores.

4.1 Explorando a RSL na Construção do Estado da Arte

A primeira etapa para o desenvolvimento de uma RSL é a definição das questões de pesquisa que fundamentam a mesma (PETERSEN; VAKKALANKA; KUZNIARZ, 2015). Desta forma, as questões de pesquisa que fundamentam essa RSL e norteiam o estudo são:

QP1: Quais estratégias para raciocínio de contexto são utilizadas em conjunto?

O objetivo dessa questão de pesquisa é investigar se existe um padrão no uso conjunto de estratégias para raciocínio, uma vez que diferentes trabalhos exploram estratégias distintas. Assim, espera-se analisar, por exemplo, se uma estratégia para raciocínio é considerada complementar a outra.

QP2: Em quais domínios são aplicadas as estratégias híbridas para raciocínio?

O objetivo dessa questão de pesquisa é identificar domínios com potencial para explorar estratégias híbridas para o raciocínio de contexto. Além disso, é investigado se existe alguma tendência nos domínios explorados nos trabalhos selecionados.

QP3: Que método é empregado na avaliação?

O objetivo dessa questão de pesquisa é investigar se existe um método usado pela maioria dos trabalhos selecionados. Assim, espera-se identificar que método é consi-

derado adequado para avaliar as abordagens híbridas para raciocínio de contexto.

A *string* de busca utilizada nesta RSL foi:

((“hybrid reasoning”) OR (“hybrid approach”) OR (“hybrid framework”) OR (“hybrid strategy”)) AND ((“context aware”) OR (“context awareness”)).

A *string* de busca foi concebida considerando apenas as estratégias híbridas, então a primeira parte da *string* tem todos os termos acompanhados da palavra *hybrid*, devido a este ser um termo amplamente utilizado na literatura (BETTINI et al., 2010). A segunda parte da sequência de pesquisa (após AND) é usada para pesquisar apenas as estratégias cientes de contexto.

Para identificar qual período deve ser analisado nesta RSL, aplicou-se a *string* concebida para pesquisar em todo o texto de artigos contidos em seis bases de indexação: (i) ACM; (ii) IEEE; (iii) Science Direct; (iv) Scopus; (v) Springer; e (vi) Web of Science. A escolha das bases de indexação se deram devido a sua importância e por cobrirem a maioria dos periódicos e artigos de conferências publicados no campo da ciência da computação com foco em aplicações cientes de contexto. Para auxiliar no desenvolvimento da RSL foi utilizada a ferramenta StArt[1], desenvolvida pela Universidade Federal de São Carlos.

A Figura 2 mostra o número de artigos retornados em cada base de indexação. Como pode ser visto na Figura 2, há um aumento no número de trabalhos retornados a partir de 2004. Assim, decidiu-se realizar a RSL no período entre 2004 e 2018.

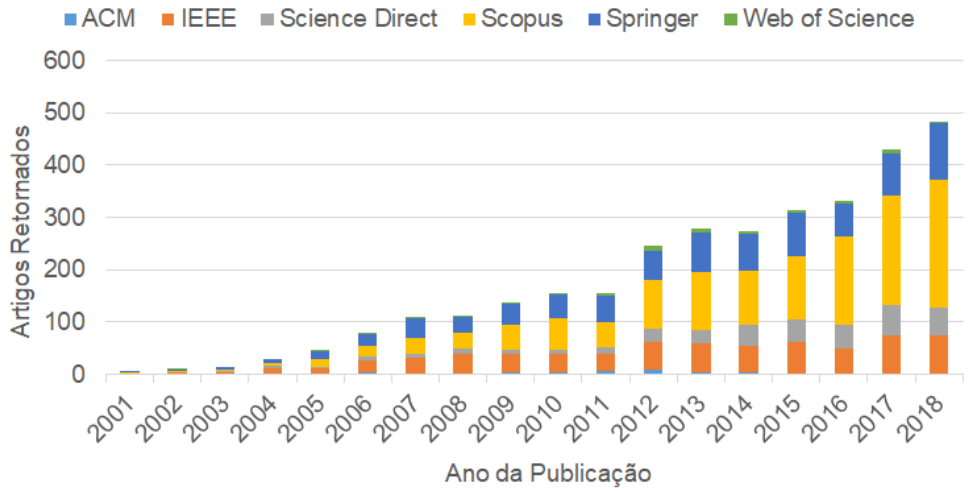

Figura 2 – Volume da produção cientifica retornado por ano.

O processo de busca pelos artigos seguiu o fluxo de execução apresentado na Figura 3, na qual pode ser visualizado que o processo possuiu quatro etapas. A primeira etapa é a aplicação da *string* de busca nas bases de indexação selecionadas. As etapas restantes executam os filtros aplicados nesta RSL que atendem aos seguintes critérios de exclusão:

[1] <http://lapes.dc.ufscar.br/tools/start_tool>

1. trabalhos que não foram publicados entre 2004 e 2018;
2. trabalhos que não contêm a *string* de busca no título, resumo ou palavras-chave;
3. trabalhos duplicados;
4. trabalhos que não são artigos publicados em conferências ou periódicos;
5. trabalhos que não estão escritos em inglês ou português;
6. trabalhos que não apresentam estratégias para raciocínio de contexto;
7. trabalhos que propõem o uso de somente uma estratégia para raciocínio;
8. trabalhos que apresentam pequenas modificações de outro artigo do mesmo grupo de pesquisa.

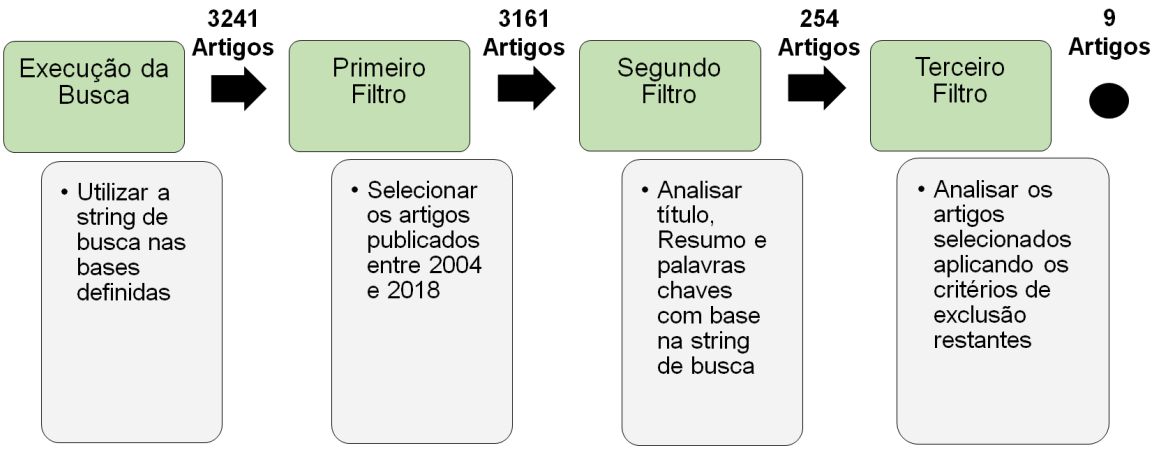

Figura 3 – Fluxo do processo de revisão sistemática.

Na segunda etapa é realizado o primeiro filtro utilizando o primeiro critério de exclusão, o qual elimina os trabalhos que não foram publicados entre 2004 e 2018.

A terceira etapa é responsável por realizar um filtro com base no segundo critério de exclusão, o qual exclui artigos que não contêm a *string* de pesquisa especificamente no título, resumo ou palavras-chave. Diversos trabalhos presentes na literatura utilizam este critério para limitar o processo de busca e evitar resultados falso-positivos da pesquisa de texto completo (DIKICI; TURETKEN; DEMIRORS, 2018; HEEAGER; NIELSEN, 2018; HOSSEINZADEH et al., 2018).

A última etapa é responsável por analisar todo o texto, tendo como base os critérios de exclusão restantes, sendo selecionados somente os artigos que propõem abordagens híbridas para raciocínio de contexto.

O terceiro critério de exclusão remove os artigos duplicados, enquanto o quarto critério de exclusão exclui estudos que são livros, revisão de conferência ou anotações de aula (ZAVALA; FRANCH; MARCO, 2018).

O quinto critério de exclusão remove artigos que não foram escritos em inglês ou português. O sexto critério de exclusão exclui artigos que não apresentam estratégias para raciocínio, enquanto o sétimo critério de exclusão remove trabalhos que propõem o uso de apenas uma estratégia para o raciocínio de contexto.

O último critério de exclusão elimina trabalhos preliminares da mesma pesquisa, sendo selecionado apenas o artigo mais recente e com maior completude (DIKICI; TURETKEN; DEMIRORS, 2018). Para aplicação deste critério, todo o texto do artigo foi analisado, verificando se os trabalhos tratavam da mesma pesquisa, registrando apenas pequenas diferenças entre eles.

A Figura 4 apresenta uma visão geral dos artigos retornados por cada base de indexação durante o processo de revisão sistemática empregado.

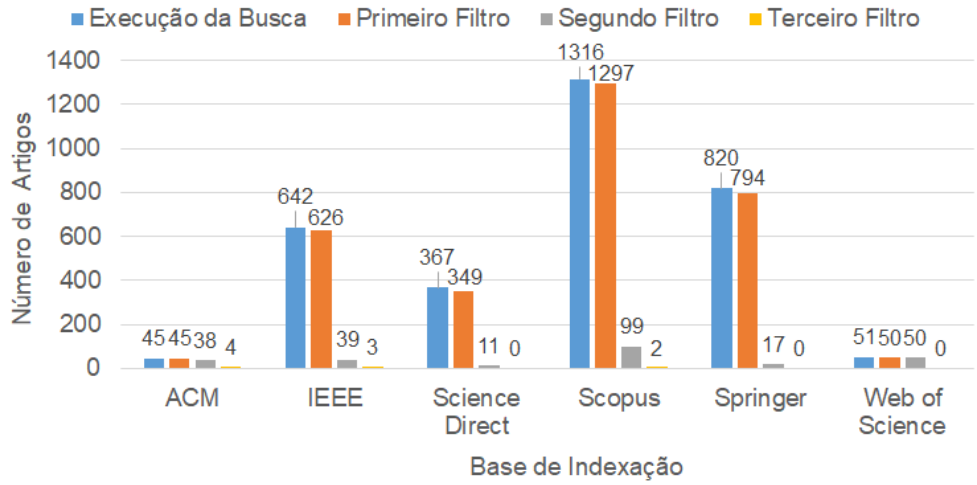

Figura 4 – Artigos retornados em cada base de indexação.

No último filtro desta RSL, foram analisados 254 documentos, no qual foram detectados 140 artigos duplicados. Considerando os 114 artigos restantes, dezesseis documentos foram excluídos pelo critério de documentos que não são artigos publicados em conferências ou periódicos. Depois, foram excluídos 2 trabalhos porque não foram escritos em inglês ou português.

Em relação aos 96 artigos restantes, passou-se a realizar uma revisão detalhada do texto do artigo. Primeiramente optou-se por aplicar o critério de excluir trabalhos que não propõem estratégias para raciocínio de contexto, o que resultou na exclusão de 38 artigos.

Considerando os 58 artigos restantes procurou-se excluir trabalhos que propõem o uso de apenas uma estratégia para o raciocínio contextual, o que resultou na exclusão de 43 trabalhos. Dentre esses artigos, as seguintes estratégias para o raciocínio foram detectadas:

- dezenove utilizam lógica descritiva;

- oito usam aprendizagem não supervisionada;
- sete utilizam regras;
- seis empregam aprendizagem supervisionada;
- dois exploram lógica fuzzy;
- um usa lógica probabilística.

Por último, nos 15 artigos restantes, aplicou-se o critério de exclusão que verifica os trabalhos que apresentam pequenas diferenças em relação a trabalhos anteriores de um mesmo grupo de pesquisa, analisando também a integralidade do trabalho, o que resultou na exclusão de 6 artigos.

A Tabela 3 apresenta uma visão geral do número de artigos que foram suprimidos com base em cada critério de exclusão. Conforme pode ser visualizado, o critério que excluiu o maior número de trabalhos foi de analisar o título, resumo e palavras-chave tendo como base a *string* de busca, sendo excluídos 2907 artigos.

Tabela 3 – Número de artigos suprimidos em cada critério de exclusão.

	Critérios de Exclusão							
Base Digital	1	2	3	4	5	6	7	8
ACM	0	7	0	0	0	16	14	3
IEEE	16	587	14	0	1	12	9	1
Science Direct	18	338	5	0	0	2	3	1
Scopus	19	1198	69	15	1	4	9	1
Springer	26	777	12	1	0	0	3	0
Web of Science	1	0	40	0	0	4	5	0
Total	80	2907	140	16	2	38	43	6
Artigos Restantes	3161	254	114	98	96	58	15	9

Ao final do processo da RSL foram selecionados nove artigos que propõem o uso de abordagens híbridas para raciocínio de contexto. Estes artigos são apresentados com maiores detalhes na próxima seção.

4.2 Trabalhos Relacionados Identificados na RSL Realizada

Após a realização da revisão sistemática da literatura apresentada anteriormente, foram selecionados nove artigos, os quais são apresentados a seguir, sendo explorados aspectos referentes as suas estratégias para raciocínio, bem como suas principais funcionalidades.

4.2.1 Raciocínio Baseado Lógica Descritiva e em Regras

Nesta seção, são apresentados os trabalhos (AGOSTINI; BETTINI; RIBONI, 2009; LOPES et al., 2012), os quais propõem de raciocínio híbrido baseado em lógica descritiva e em regras.

4.2.1.1 Estratégia Híbrida para o Middleware CARE

Em Agostini; Bettini; Riboni (2009), foi proposto o *middleware Context Aggregation and REasoning* (CARE), que visa apoiar a adaptação ciente ao contexto dos serviços de internet em um ambiente de computação móvel. Um dos principais requisitos do *middleware* é suportar o raciocínio de contexto de maneira eficiente.

A Figura 5 apresenta os principais componentes da arquitetura CARE. Cada entidade possui um gerenciador de perfil dedicado para manipular seus próprios dados de contexto *User Profile Manager* (UPM), *Operator Profile Manager* (OPM) e *Service Provider Profile Manager* (SPPM). Os perfis incluem dados de contexto brutos e dados de contexto baseados em lógica descritiva que são expressos por meio de referências a classes e relações ontológicas.

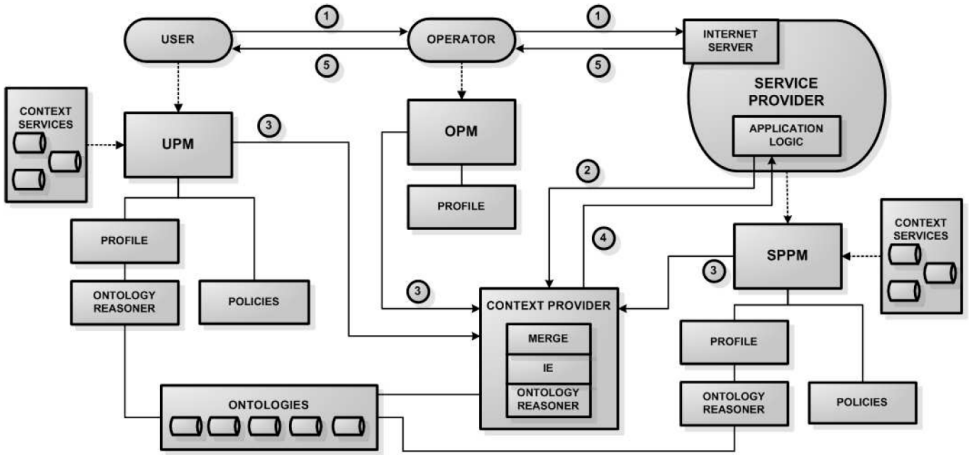

Figura 5 – Visão geral do middleware CARE.
Fonte: AGOSTINI; BETTINI; RIBONI, 2009, p. 8.

Para realizar o raciocínio contextual, os autores propuseram uma abordagem híbrida explorando a estratégia baseada em lógica descritiva usando ontologia, e a estratégia baseada em regras de uma linguagem de programação lógica. O raciocínio

ontológico é executado para derivar classes particulares de dados de contexto que não podem ser expressos usando regras.

A interação entre as estratégias baseadas em ontologias e regras é fornecida pela possibilidade de definir políticas cujas pré-condições envolvam dados de contexto que podem ser derivados pelo raciocínio ontológico. Esses dados são usados em condições prévias de políticas para determinar novas informações, como preferências do usuário e diretivas de adaptação. Tanto o usuário quanto o provedor de serviços podem declarar políticas na forma de regras sobre dados de contexto, a fim de derivar dados de contexto de nível mais alto a partir de dados mais simples e determinar os parâmetros de adaptação do serviço.

Para avaliar a proposta, os autores apresentam experimentos sobre o raciocínio ontológico, nos quais uma ontologia modela um ambiente sociocultural de usuários móveis. Assim, os autores avaliam apenas a estratégia baseada em ontologia, utilizando-a para derivar a atividade específica realizada por um usuário.

4.2.1.2 Estratégia Híbrida para a Abordagem DynamiCC

O artigo de Lopes et al. (2012) propõe uma abordagem ciente do contexto chamada de DynamiCC. A proposta tem como premissa a concepção de um modelo de contexto e uma arquitetura de software que possibilitem a interpretação e a composição dinâmica das informações adquiridas por sensores posicionados em diferentes localidades do ambiente ubíquo. Com isso, a DynamiCC busca viabilizar a construção de contextos complexos, cujas informações contextuais possam ser obtidas de forma distribuída, bem como ter suas regras de processamento alteradas sem a necessidade de interromper a execução das aplicações.

A DynamiCC tem como premissas: (i) o emprego de um modelo híbrido de contexto, constituído por uma ontologia e uma base de dados relacional; (ii) o processamento das informações adquiridas por sensores em diferentes células do ambiente ubíquo; (iii) a notificação as partes interessadas, conforme as regras de processamento do contexto utilizadas pelas diferentes aplicações; (iv) a possibilidade dos sensores e/ou atuadores serem ativados, desativados, configurados e consultados a qualquer momento, com base em seus respectivos tratadores, de forma ubíqua, em qualquer célula do ambiente; e (v) a possibilidade de alterações das regras de processamento do contexto durante a execução das aplicações.

A arquitetura de software da DynamiCC é apresentada na Figura 6, na qual podem ser visualizados os módulos distribuídos entre os dois tipos de servidores: Celular e de Borda. Para realização do processamento das informações contextuais é proposta uma abordagem híbrida para raciocínio, a qual combina um raciocínio baseado em lógica descritiva, com base em axiomas e regras para inferir novos conhecimentos baseadas nas instâncias da ontologia. Além deste, é proposta a utilização de um

raciocínio baseado em regras do tipo ECA (Evento-Condição-Ação), o qual realiza o raciocínio das informações contextuais providas por um modelo relacional.

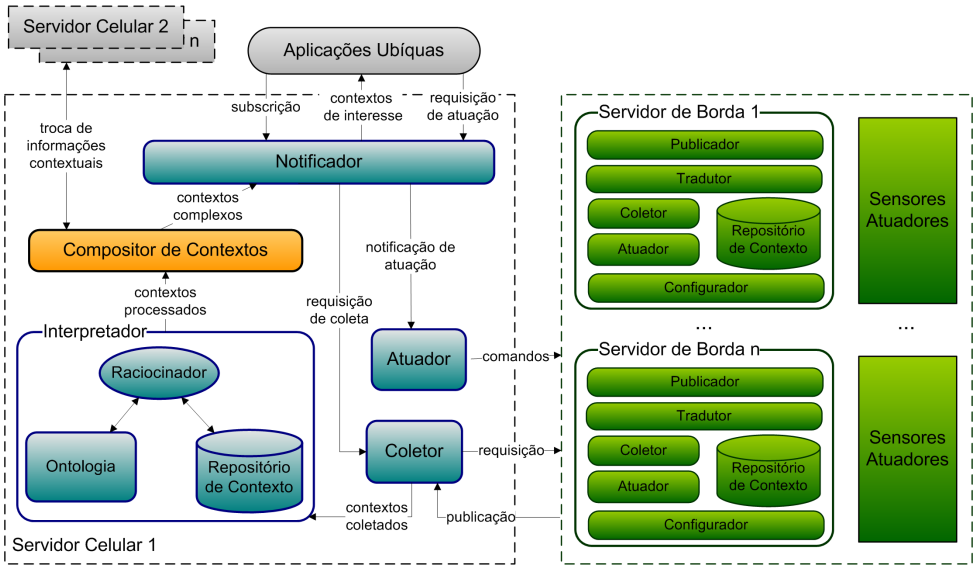

Figura 6 – Visão geral da abordagem DynamiCC.
Fonte: LOPES et al., 2012, p. 6.

Embora utilizando uma abordagem híbrida para o raciocínio contextual, a DynamiCC utiliza as duas estratégias propostas individualmente. Quando os dados coletados são enviados para o modelo relacional a DynamiCC utiliza a estratégia baseada em regras, caso os dados sejam transmitidos para o modelo ontológico é explorado o raciocínio baseado em lógica descritiva. As aplicações cientes de contexto devem informar quais informações monitoradas serão processadas por cada estratégia no momento da configuração da DynamiCC.

Para avaliar a DynamiCC, os autores apresentam quatro cenários que ilustram em apenas um deles a aplicação da estratégia de raciocínio utilizando uma ontologia, permitindo obter informações implícitas, de acordo com as relações e propriedades das classes da ontologia. Enquanto os outros três cenários cobrem os seguintes aspectos: (i) dados adquiridos do ambiente e ação nele; (ii) comunicação entre os componentes arquitetônicos; e (iii) composição de contextos dinâmicos.

4.2.2 Raciocínio Baseado Aprendizagem Supervisionada e em Lógica Descritiva

Nesta seção, são apresentados os trabalhos (RAZZAQ; AMIN; LEE, 2017; RIBONI; BETTINI, 2011; XU et al., 2011) que propõem o raciocínio híbrido baseado em aprendizagem supervisionada e em lógica descritiva.

4.2.2.1 *Estratégia Híbrida para o Sistema COSAR*

O artigo de Riboni; Bettini (2011) define uma arquitetura para um sistema de reconhecimento de atividade sensível ao contexto móvel chamado COSAR. A arquitetura realiza o monitoramento dos sensores usados no corpo e espalhados no ambiente. A Figura 7 apresenta o sistema COSAR.

A camada inferior (*SENSORS*) monitora os sensores presentes no corpo de um usuário, fornecendo dados como leituras de acelerômetros, parâmetros fisiológicos e também de sensores espalhados pelo ambiente. Os dados fornecidos por sensores ambientais e corporais são comunicados por meio de uma conexão sem fio ao *USER MOBILE DEVICE* e mesclados com os dados do sensor recuperados pelo próprio dispositivo, para criar um vetor de recursos que será usado para prever a atividade do usuário.

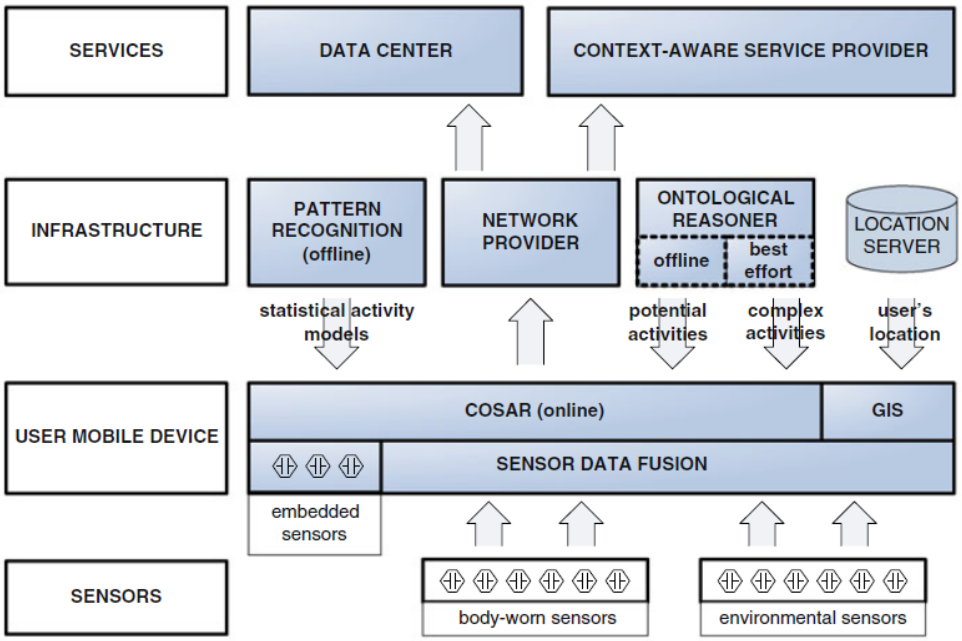

Figura 7 – Visão geral do sistema COSAR.
Fonte: RIBONI; BETTINI, 2011, p. 274.

Para prever a atividade do usuário, foi proposto um raciocínio híbrido, explorando uma estratégia baseada em lógica descritiva e aprendizagem supervisionada. Para realizar o raciocínio da estratégia de lógica descritiva foi utilizada uma ontologia, a qual modela atividades, artefatos, pessoas, vias de comunicação e localizações simbólicas, e que expressa relações e constrangimentos entre essas entidades. A mesma ontologia é usada para descrever e reconhecer atividades sociais que, segundo os autores, dificilmente seriam identificáveis por uma estratégia de aprendizagem puramente supervisionada.

A estratégia baseada em aprendizagem supervisionada realiza a inferência com base nos dados brutos recuperados dos sensores para prever as atividades mais pro-

váveis. Enquanto a ontologia é aplicada para refinar os resultados da inferência da estratégia de aprendizagem supervisionada, selecionando o conjunto de atividades possíveis executadas por um usuário com base em seu contexto atual. Assim, a informação de contexto é principalmente explorada pelo raciocinador ontológico, e apenas em casos limitados pela estratégia de aprendizagem supervisionada.

Para avaliar a proposta, os autores apresentam um experimento baseado na identificação de atividades físicas realizadas em ambientes fechados e ao ar livre por seis voluntários. Na avaliação, os autores comparam a precisão e o custo computacional da estratégia híbrida proposta com o uso de aprendizagem supervisionada individualizada. Os resultados obtidos mostram a superioridade da estratégia híbrida.

4.2.2.2 Estratégia Híbrida para o Middleware IMERA

Em Xu et al. (2011), foi proposto um *middleware* de reconhecimento de contexto chamado IMERA para prover inteligência a um ambiente. A arquitetura do *middleware* é apresentada na Figura 8, a qual é dividida em quatro camadas: (i) *Sensor Layer* tem acesso a diferentes entidades computacionais e diversos dispositivos de sensores e atuadores; (ii) *Interface Layer* permite a troca de informações entre dispositivos e a camada de gerenciamento do *middleware*; (iii) *Middleware Management Layer* que é baseada na fusão de dados de sensores, um mecanismo de raciocínio, uma base de conhecimento de contexto, um banco de dados de contexto e um mecanismo de consulta de contexto; e (iv) *High Level Interface Layer* que realiza a comunicação entre o *middleware* e o aplicação.

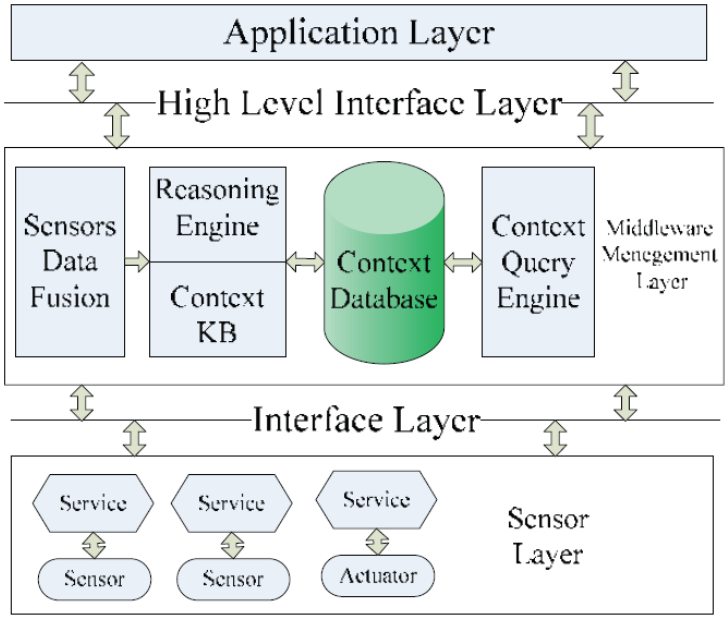

Figura 8 – Visão geral do middleware IMERA.
Fonte: XU et al., 2011, p. 2.

Para realizar o processamento das informações contextuais, os autores propuse-

ram um raciocínio híbrido baseado nas estratégias de lógica descritiva usando uma ontologia e aprendizagem supervisionada com o uso da técnica de árvore de decisão. Desta forma, os autores enfatizam que a estratégia proposta fornece uma resolução para problemas mais flexível, robusta e eficiente em situações complexas de inteligência ambiental (XU et al., 2011).

Os autores não mostram nenhuma informação sobre o fluxo de execução de raciocínio ou como os aplicativos podem selecionar o uso de uma ou outra estratégia. Além disso, no trabalho não são apresentados exemplos da estratégia baseada em ontologia, e se nessa estratégia eles usaram axiomas ou regras ontológicas.

Para avaliar a proposta, os autores apresentam um cenário, no qual eles dizem como ambos os raciocinadores podem ser usados. A estratégia de ontologia processaria a atividade do usuário a partir de contextos coletados, enquanto a estratégia supervisionada construiria a árvore de decisão com base no perfil do usuário.

4.2.2.3 Estratégia Híbrida para o Framework Mining Minds

Em Razzaq; Amin; Lee (2017) é proposta uma estratégia híbrida para aumentar a precisão do raciocínio de contexto do *framework Mining Minds*, um *framework* que oferece suporte personalizado para saúde e bem-estar. Para realizar o raciocínio das informações contextuais é proposta uma abordagem híbrida, a qual é baseada aprendizagem supervisionada e em lógica descritiva usando uma ontologia. A estratégia baseada em aprendizagem supervisionada suporta a inferência classificando contextos de alto nível com base no conjunto de dados instanciado na ontologia.

As fases da abordagem híbrida é mostrada na Figura 9, a qual contém métodos para transformação de *Resource Description Framework*/*eXtensible Markup Language* (RDF/XML) para csv/arff, tarefas essenciais para extração de vetor de recursos de RDF, a seleção de algoritmo e a etapa de treinamento do modelo usando os recursos extraídos. Para realização do armazenamento dos dados de contexto providos pela ontologia os autores propõem a utilização do modelo de triplas com o uso da ferramenta Jena-TDB[2].

O fluxo de execução de raciocínio explorado pelos autores é sempre o mesmo, a estratégia baseada em ontologia verifica a consistência de contextos, deduz informações de contexto de alto nível e armazena essas informações no modelo de triplas. Depois, a estratégia de aprendizado supervisionado usa os dados armazenados no modelo de triplas, como um conjunto de dados de treinamento. Assim, com essa combinação, os autores aprimoram a precisão da classificação da estratégia de aprendizagem supervisionada.

Para avaliação experimental, foram coletados conjuntos de dados envolvendo 20 usuários em diferentes dias e horários. Os usuários receberam *smartwatch*,

[2]https://jena.apache.org/documentation/tdb/

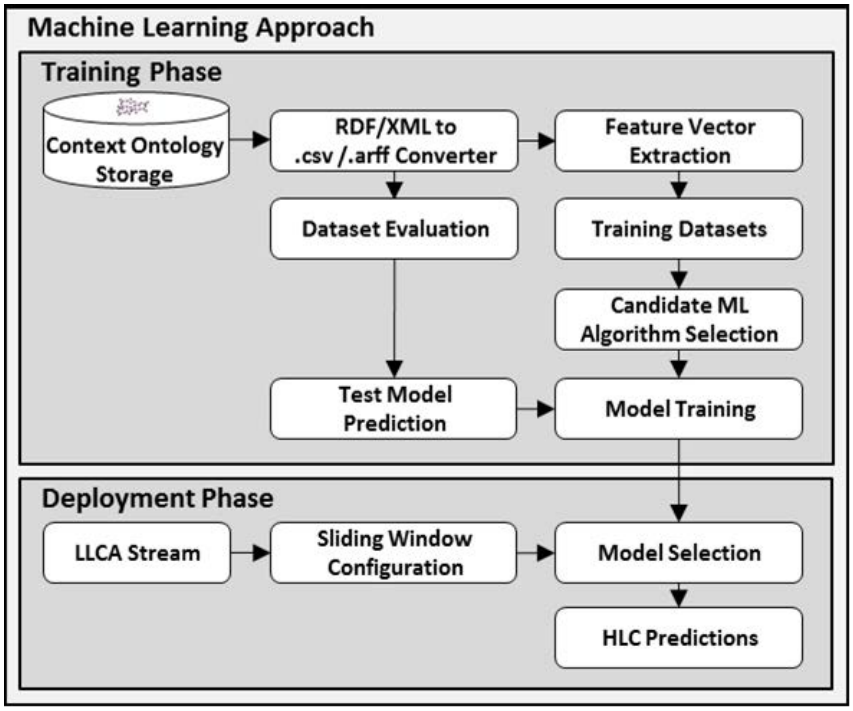

Figura 9 – Fases da estratégia híbrida do framework Mining Minds
Fonte: RAZZAQ; AMIN; LEE, 2017, p. 405.

smartphone e posturas esqueléticas foram detectadas usando um *kinect*. Nos testes realizados foi combinada uma ontologia com seis diferentes técnicas de aprendizagem supervisionada: *Naive Bayes*, *KStar*, *IBK*, *J48*, *RandomForest* e *RandomTree*. A técnica que combinada com a ontologia apresentou o melhor resultado foi a *RandomForest*, enquanto que a *KStar* apresentou o pior resultado. A estratégia de aprendizagem supervisionada *RandomForest* usada em conjunto com a estratégia ontológica aumentou o desempenho de precisão de 91,5% para 99,99%.

4.2.3 Raciocínio Baseado em Casos e em Lógica Descritiva

Em Strobbe et al. (2012) é proposta uma plataforma de serviços intitulada de *Context-Aware Service Platform* (CASP). Essa plataforma tem como principais características ser ciente de contexto e tratar da agregação e da abstração de informações de contexto usando ontologias para representar as mesmas.

A plataforma CASP utiliza na etapa de processamento de contexto duas estratégias de raciocínio, uma baseada em casos e a outra em lógica descritiva. A estratégia baseada em lógica descritiva resolve problemas a partir do zero, enquanto os sistemas baseados em casos usam situações pré-registradas para lidar com novas instâncias semelhantes.

A Figura 10 mostra a arquitetura do núcleo da plataforma CASP. Quando um provedor de contexto atualiza informações na base de conhecimento, os raciocinadores baseados em lógica descritiva e em casos são notificados. O raciocínio baseado em

lógica descritiva avalia as regras fornecidas e adiciona as informações derivadas à base de conhecimento. O raciocínio baseado em casos recupera as situações semelhantes e usa as soluções correspondentes para atualizar as informações na base de conhecimento representada pelo modelo ontológico.

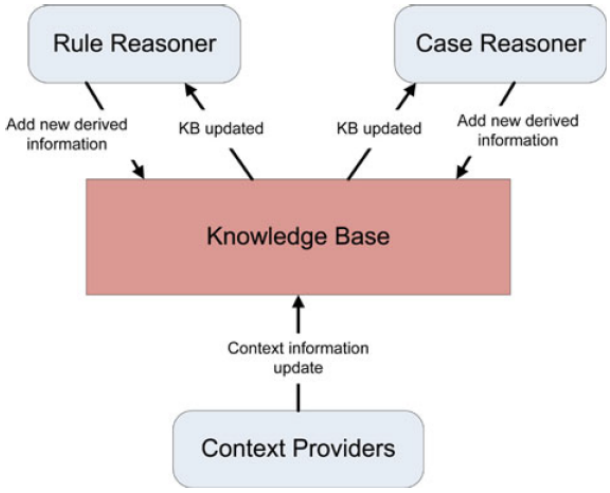

Figura 10 – Abordagem do processamento de contexto em CASP.
Fonte: STROBBE et al., 2012, p. 587.

As duas estratégias de raciocínio são utilizadas individualmente, ou seja, os autores não propuseram qualquer combinação dessas estratégias. O fluxo de execução do raciocínio é sempre o mesmo, depois de ocorrer uma atualização na base de conhecimento as estratégias são notificadas, executam o processo de raciocínio e adicionam as novas informações obtidas na base de conhecimento.

Para avaliar a proposta, os autores apresentam dois casos de uso em que ambas estratégias são usadas para obter informações extras. No caso de uso de compartilhamento de escritório, a combinação do raciocínio baseado em lógica descritiva e em casos permitiu aprender trajetórias típicas de um usuário e melhorar a localização em tais trajetos em 42%. Em ambos os casos de uso, a abordagem híbrida proporcionou uma melhoria significativa na obtenção de informações.

4.2.4 Raciocínio Baseado em Casos e em Lógica Fuzzy

O artigo de Yuan; Herbert (2014) apresenta um *framework* para raciocínio híbrido desenvolvido para o sistema *Context-Aware Real-time Assistant* (CARA). O *framework* realiza o monitoramento de um ambiente doméstico inteligente, fornecendo uma fusão de dados cientes de contexto, bem como mecanismos para detecção de anomalias que suportam a atividade de análise e geração de alertas.

O raciocínio híbrido é baseado em casos e em lógica fuzzy, permitindo tornar o CARA mais robusto e adaptável a um ambiente sujeito a mudança. Emprega-se o uso da estratégia baseada em casos para detectar anomalias condicionais para automa-

ção residencial e a baseada em lógica fuzzy para lidar com exceções, bem como para obter recuperação de casos e adaptação de casos sensíveis à consulta.

Os autores afirmam que a abordagem híbrida para raciocínio que propõem tem o potencial de minimizar a dificuldade de obter os dados contextuais, uma vez que a interpretação dos dados detectados como contexto é, em geral, imperfeita e incompleta. O método de raciocínio baseado em regras difusas sobrepõe, até certo ponto, a outros modelos matemáticos desenvolvidos para lidar com a imprecisão e a incerteza (YUAN; HERBERT, 2014). A abordagem de raciocínio baseado em casos aborda problemas de dados incompletos e limitações de conhecimento de domínio.

As interações de alto nível da abordagem híbrida para raciocínio são apresentadas na Figura 11. Dados brutos provenientes de sensores são processados e integrados com conhecimento de contexto pelos serviços de fusão de dados, produzindo contextos para a construção de consultas de casos e conjuntos fuzzy.

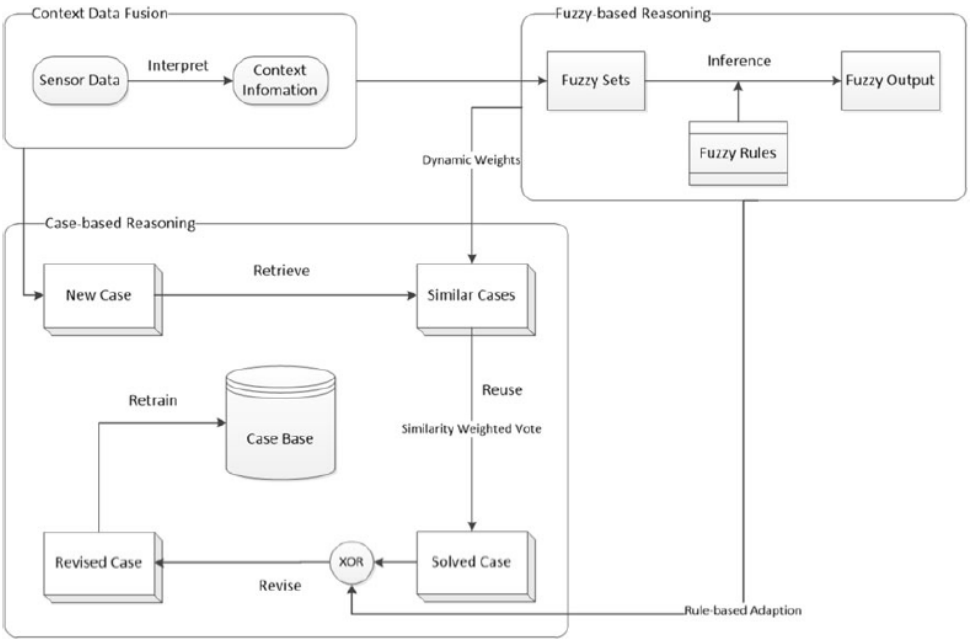

Figura 11 – Estrutura do raciocínio híbrido de contexto do framework CARA.
Fonte: YUAN; HERBERT, 2014, p. 874.

Após a fusão dos dados, o componente de raciocínio baseado em casos inicia a detecção de anomalias e automação residencial. Enquanto isso, o componente baseado em regras fuzzy carrega as regras difusas da base de dados de regras de inferência para gerar os contextos de alto nível (por exemplo, condição médica e evento de acidente) e ainda, é responsável por realizar a identificação da situação atual do usuário (normal, anormal ou de emergência).

Por último, se a situação detectada for anormal ou uma emergência, então uma notificação ou alarme é enviado para o servidor de monitoramento remoto e uma chamada de serviço de emergência pode ser disparada. Os dados brutos coletados e

o caso revisado são armazenados para melhorar a base de casos e a subsequente análise adicional.

O fluxo de execução de raciocínio explorado pelos autores é sempre o mesmo, a estratégia fuzzy é usada para adaptar o raciocínio baseado em casos. Os autores aplicam fuzzy à consulta de entrada e usam o valor da saída fuzzy para ajustar os pesos usados para a medida de similaridade de recursos. Além disso, as saídas de ambas as estratégias são combinadas usando um operador lógico para produzir valores de resposta.

Para avaliar a proposta, os autores realizaram um experimento para testar a correção do sistema proposto em um ambiente de saúde generalizado. A estratégia de raciocínio foi avaliada com base em sua precisão para detectar a situação de interesse. No experimento realizado, a estratégia baseada híbrida obteve um melhor desempenho para detecção de anomalias.

4.2.5 Raciocínio Baseado em Aprendizagem Supervisionada e Lógica Probabilística

No artigo de Baz; Yoruk; Cetin (2016) é proposto um sistema que classifica individualmente os produtos de varejo em uma prateleira de loja. O sistema de classificação proposto combina os pontos fortes de classificadores com e sem contexto. O objetivo do sistema é projetar um modelo probabilístico que codifique as relações entre os produtos na prateleira e combine isso com os métodos de classificação de imagem baseada em visão.

Para realizar o raciocínio dos dados contextuais, é proposta uma combinação da estratégia de aprendizagem supervisionada com a estratégia de lógica probabilística. São propostas duas abordagens híbridas diferentes, a primeira combina *Support Vector Machines* (SVMs) e um modelo gráfico generativo com base em *Hidden Markov Models* (HMMs). Na segunda, os SVMs são combinados com uma abordagem discriminativa baseada em *Conditional Random Fields* (CRFs) tendo como intuito formar um novo contexto.

Na Figura 12 é ilustrado uma visão geral do sistema, que consiste em duas partes principais. A primeira parte visa classificar o produto de varejo usando informações visuais provenientes da imagem do produto. Na segunda parte, é inferida as categorias de produtos combinando os resultados do classificador livre de contexto da primeira parte com o modelo de contexto estatístico. O modelo de contexto é baseado em um modelo gráfico estruturado em cadeia, em que cada nó representa um produto detectado e as bordas codificam suas relações espaciais de adjacência na cena.

O fluxo de execução de raciocínio explorado pelos autores é sempre o mesmo, as saídas da estratégia de aprendizado supervisionado são usadas como entradas na estratégia lógica probabilística. Assim, com essa combinação, os autores aprimoram

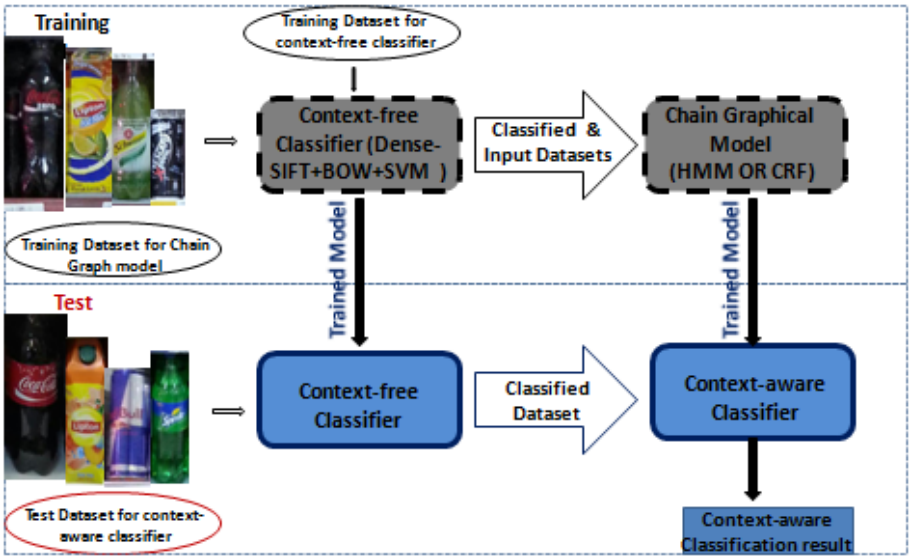

Figura 12 – Abordagem empregada para raciocínio dos produtos de varejo.
Fonte: BAZ; YORUK; CETIN, 2016, p. 2.

os resultados de classificação de produtos de varejo refinados.

Para avaliar a proposta, os autores apresentam uma avaliação experimental usando o conjunto de dados Vispera[3], que consiste em 3920 imagens anotadas de prateleiras de varejo contendo produtos de refrigerantes. A avaliação foi baseada na detecção de produtos pelas estratégias de raciocínio. Os resultados mostram que com a combinação da técnica SVMs com o método HMM foi alcançada uma melhoria de 9,5%. Enquanto que a combinação de SVMs com CRF alcançou uma melhoria de 11.4%.

4.2.6 Raciocínio Baseado em Lógica Descritiva e em Lógica Probabilística

No artigo Li; Martínez; Rubio (2017) é proposta uma abordagem híbrida para o raciocínio de dados contextuais para o *middleware Smart and Networking Underwater Robots in Cooperation Meshes* (SWARMs). A abordagem proposta é constituída de duas estratégias para raciocínio: (i) lógica descritiva, sendo utilizados tanto axiomas internos à ontologia, quanto regras escritas na linguagem SWRL; e (ii) lógica probabilística utilizando a técnica *Multi-Entity Bayesian Network*, a qual consiste de um sistema lógico que integra a lógica da primeira ordem com a teoria da probabilidade bayesiana[4].

A abordagem híbrida é apresentada na Figura 13, na qual pode ser visualizado as duas estratégias utilizadas. Para a seleção das estratégias de raciocínio dois fatores-chave são considerados. O primeiro deles é o desempenho da estratégia escolhida, sendo dado a opção de escolher a que possui o melhor desempenho. O segundo fator

[3] www.vispera.co
[4] <http://www.pr-owl.org/mebn/index.php>

é a compatibilidade com a modelo utilizado para a representação do contexto.

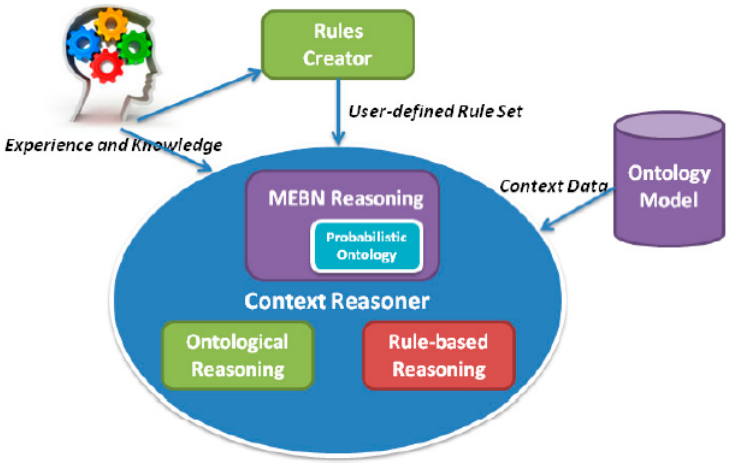

Figura 13 – Abordagem híbrida do middleware SWARMs.
Fonte: LI; MARTÍNEZ; RUBIO, 2017, p. 8.

Os autores destacam que, com a incorporação das duas estratégias é proporcionado um melhor desempenho para o raciocínio de contexto, aproveitando-se dos pontos positivos de cada estratégia, com o intuito de compensar as fraquezas das mesmas. Os autores destacam que a abordagem híbrida proposta é eficaz para fornecer diferentes capacidades de raciocínio e satisfazer as diferentes necessidades das aplicações (LI; MARTÍNEZ; RUBIO, 2017).

As duas estratégias de raciocínio são utilizadas individualmente, ou seja, os autores não propuseram qualquer combinação entre as mesmas. O fluxo de execução de raciocínio para as duas estratégias é o mesmo, as estratégias consultam as informações armazenadas no modelo ontológico, executam o processo de raciocínio e adicionam as novas informações obtidas no modelo ontológico.

Para avaliar a proposta, os autores descrevem um estudo de caso sobre a detecção de derramamento de óleo, sendo verificado a utilidade e aplicabilidade do mecanismo de raciocínio híbrido do contexto. Os resultados simulados mostraram que, em teoria, o método de raciocínio de contexto híbrido pode fazer inferências lógicas em contextos específicos e também o raciocínio probabilístico sob incertezas no campo de robôs subaquáticos.

4.2.7 Análise dos Trabalhos Relacionados a Raciocínio Híbrido de Contexto

A Tabela 4 apresenta uma análise comparativa entre os trabalhos selecionados na RSL realizada. Esta análise é realizada com base nos seguintes critérios: (i) os tipos de sensores empregados na etapa de aquisição; (ii) os modelos utilizados para realizar a representação do contexto; (iii) as estratégias usadas para o raciocínio; (iv) a forma de armazenamento dos dados contextuais; e (v) algumas ferramentas utilizadas para o desenvolvimento do trabalho.

Tabela 4 – Comparação entre os trabalhos relacionados.

Trabalhos	Aquisição	Modelo	Raciocínio	Armazenamento	Ferramentas
AGOSTINI; BETTINI; RIBONI, 2009	Sensores Físicos Lógicos	Ontológico	Lógica Descritiva e Regras	Informação Não Disponível	Protégé, Racer
RIBONI; BETTINI, 2011	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Informação Não Disponível	Java, Weka, Protégé, RacerPro
XU et al., 2011	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Relacional	Informação Não Disponível
LOPES et al., 2012	Sensores Físicos Lógicos	Ontológico e Relacional	Lógica Descritiva e Regras	Relacional	Java, Jena, OSGI, Protégé, Pellet, PostgreSQL
STROBBE et al., 2012	Sensores Físicos Lógicos	Ontológico	Baseado em Casos e Lógica Descritiva	Informação Não Disponível	Java, Jena, OWL API, jCOLIBRI
YUAN; HERBERT, 2014	Sensores Físicos Lógicos	Fuzzy	Baseado em Casos e Lógica Fuzzy	XML	jCOLIBRI
BAZ; YORUK; CETIN, 2016	Informação Não Disponível	Informação Não Disponível	Aprendizagem Supervisionada e Lógica Probabilística	Informação Não Disponível	VLFEAT toolbox
RAZZAQ; AMIN; LEE, 2017	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Jena TDB (triple store)	Java, Maven, Jena, Protégé, Pellet, OWL-API, Weka
LI; MARTÃNEZ; RUBIO, 2017	Sensores Físicos Lógicos	Ontológico	Lógica Descritiva e Lógica Probabilística	Informação Não Disponível	Pellet

Com base na Tabela 4 constata-se que na etapa de aquisição apenas um trabalho não deixa claro como é feita a aquisição dos dados de contexto. Enquanto que os demais trabalhos apresentam suporte para lidar com diferentes tipos de sensores, permitindo assim o atendimento a aplicações de diferentes naturezas.

Analisando a etapa de modelagem, pode-se visualizar que quatro trabalhos utilizam o modelo baseado em ontologias, principalmente devido a sua característica de representação semântica. A utilização de modelos ontológicos segue a tendência da análise realizada em Strang; Linnhoff-Popien (2004) , a qual destacou esta estratégia de modelagem como a mais propícia a ser utilizada para representação de contexto. Destaca-se ainda que em Lopes et al. (2012) foi proposta a utilização de uma mode-

lagem híbrida, combinando os modelos ontológico e relacional.

Em relação ao raciocínio contextual, apesar de os trabalhos usarem estratégias híbridas, eles utilizam as diferentes estratégias de modo estático, não permitindo que as aplicações escolham as estratégias que melhor atendem às suas necessidades. Além disso, nota-se a ausência de uma abordagem que possibilite a combinação das diferentes estratégias para o raciocínio contextual.

Em relação ao armazenamento das informações contextuais, nota-se que a maioria dos trabalhos analisados não discute informações referentes a esta funcionalidade, apesar de a mesma ser importante para as aplicações cientes de contexto. O trabalho de Yuan; Herbert (2014) utiliza o modelo baseado em linguagem de marcação, empregando a linguagem XML, principalmente pela facilidade da mesma ser utilizada para transmitir as informações entre diferentes componentes. Enquanto os trabalhos de Lopes et al. (2012) e Xu et al.(2011) empregam um modelo relacional de armazenamento, o qual não é considerado um modelo satisfatório para o armazenamento de dados providos pelo modelo ontológico (CAN et al., 2017). Além destes, o trabalho de Razzaq; Amin; Lee (2017) propõe a utilização do modelo de triplas com o uso do Jena TDB, sendo um modelo mais propício para realizar a persistência das informações providas por ontologias.

Acredita-se serem necessários mecanismos com diferentes características para armazenamento de contexto, já que as aplicações cientes de contexto atuais tendem a lidar com diferentes tipos de dados, níveis semânticos entre outros aspectos. Por sua vez, a eficiência destes mecanismos é um aspecto significativo, considerando a escalabilidade do volume de dados contextuais envolvido e sua constante atualização ou consulta.

4.3 Discussão das Questões de Pesquisa da RSL

Com base na análise dos artigos selecionados, é possível responder as questões de pesquisa que apoiaram esta RSL. As questões de pesquisa e as respostas adquiridas são:

QP1: Quais estratégias para raciocínio de contexto são utilizadas em conjunto?

Foi possível observar que não há padrão para combinações de estratégias, por exemplo, estratégias baseadas em lógica descritiva são utilizadas com regras, casos, lógica probabilística e aprendizagem supervisionada. A combinação mais utilizada foi lógica descritiva com aprendizagem supervisionada. A estratégia mais utilizada foi a lógica descritiva em oito trabalhos, seguida de aprendizagem supervisionada em cinco, regras em três e raciocínio baseado em casos e lógica probabilística em dois.

QP2: Em quais domínios são aplicadas as estratégias híbridas para raciocínio?

O artigo de Strobbe et al. (2012) ilustrou a aplicabilidade da estratégia em dois casos de uso, o primeiro é de um de escritório com compartilhamento de área de trabalho e o segundo é de uma mensagem instantânea aprimorada. Os trabalhos de Agostini; Bettini; Riboni (2009); Riboni; Bettini (2011) foram utilizados para identificação da atividade física, enquanto os outros trabalhos de Baz; Yoruk; Cetin (2016); Li; Martínez; Rubio (2017); Razzaq; Amin; Lee (2017); Yuan; Herbert (2014); Xu et al. (2011) são aplicados respectivamente em: ambiente inteligente; sistema de saúde; sistemas de classificação; um paradigma de saúde e bem-estar; e robôs subaquáticos.

QP3: Que método é empregado na avaliação?

Observou-se que os trabalhos selecionados consideram adequada a utilização do método baseado em cenários de uso para avaliar suas estratégias de raciocínio híbrido. Entre os trabalhos selecionados que propuseram o uso de uma estratégia de aprendizado supervisionado, pode-se notar que um deles, o trabalho de Baz; Yoruk; Cetin (2016), usa um conjunto de dados público, enquanto outros dois, os trabalhos de Razzaq; Amin; Lee (2017); Riboni; Bettini (2011) , produzem seu próprio conjunto de dados, com seis e vinte usuários respectivamente.

4.4 Considerações do Capítulo

A análise dos nove artigos decorrentes da Revisão Sistemática da Literatura possibilitou identificar no que diz respeito ao raciocínio de contexto o estado da arte na área. Deste modo, foi possível inferir a não disponibilidade de abordagens que permitam uma combinação dinâmica das diferentes estratégias para o raciocínio de contexto.

O provimento de uma proposta desta natureza poderá permitir às aplicações maior flexibilidade na utilização dos contextos tratados, o que pode contribuir na identificação das diferentes situações de interesse.

Com essa motivação de pesquisa, no próximo Capítulo é apresentada a arquitetura DyCoRS que tem como diferencial ser flexível no oferecimento dos recursos oferecidos. A premissa é disponibilizar uma abordagem dinâmica para raciocínio de contexto, permitindo realizar a combinação de diferentes estratégias, tudo dependendo dos contextos de interesse envolvidos.

5 ARQUITETURA DYCORS: EXPLORANDO COMPOSIÇÃO DINÂMICA DE ESTRATÉGIAS PARA RACIOCÍNIO DE CONTEXTO

Este capítulo apresenta a Arquitetura DyCoRS para tratamento das informações contextuais. A DyCoRS visa prover suporte à ciência de contexto para as aplicações ubíquas, oferendo uma maior flexibilidade na utilização dos recursos para tratamento contextual. O esforço de concepção da DyCoRS considera os princípios operacionais do *middleware Execution Environment for Highly Distributed Applications* (EXEHDA) (LOPES et al., 2014), contribuindo com o Subsistema de Reconhecimento de Contexto e Adaptação, ampliando suas funcionalidades de forma a melhor atender as demandas inerentes à ciência de contexto.

Nas Seções deste Capítulo são discutidas as características da DyCoRS, sendo apresentada sua organização arquitetural e sistematizadas as funcionalidades providas por cada um de seus componentes. Os Componentes da DyCoRS para a sua concepção consideraram a discussão realizada no Capítulo 2 sobre as etapas necessárias para a construção de ciência de contexto.

5.1 Organização Arquitetural da DyCoRS

A DyCoRS foi concebida tendo por base a oferta de diferentes recursos para ciência de contexto, os quais podem ser explorados de forma dinâmica pelas aplicações. O diferencial da DyCoRS é permitir que as aplicações escolham os recursos oferecidos pela arquitetura conforme suas demandas, proporcionando uma maior flexibilidade para obtenção da ciência de contexto. Esta abordagem permite à arquitetura lidar com aplicações que necessitem configurar, por exemplo, quais estratégias para raciocínio de contexto serão utilizadas e se serão utilizadas de forma individual ou combinada.

Considerando a integração da DyCoRS com o *middleware* EXEHDA, no qual a distribuição é uma premissa operacional, para a concepção da DyCoRS foi considerada uma organização baseada em componentes, atendendo uma orientação da literatura

para a concepção de software em sistemas distribuídos (COULOURIS et al., 2013). A organização arquitetural da DyCoRS é apresentada na Figura 14, na qual foram utilizados diagramas de componentes elaborados em uma linguagem denominada *Technical Architecture Modeling* (TAM), a qual é baseada na *Unified Modeling Language* (UML) (SAP, 2007).

A arquitetura da DyCoRS é composta de sete Componentes: (i) Comunicação com o Meio; (ii) Pré-processamento; (iii) Raciocínio; (iv) Armazenamento; (v) Correlação; (vi) Reconfiguração e (vii) Disseminação. Na arquitetura DyCoRS, os dados de contexto são coletados pelo Componente de Comunicação com o Meio e repassados para o Componente de Pré-processamento para serem normalizados.

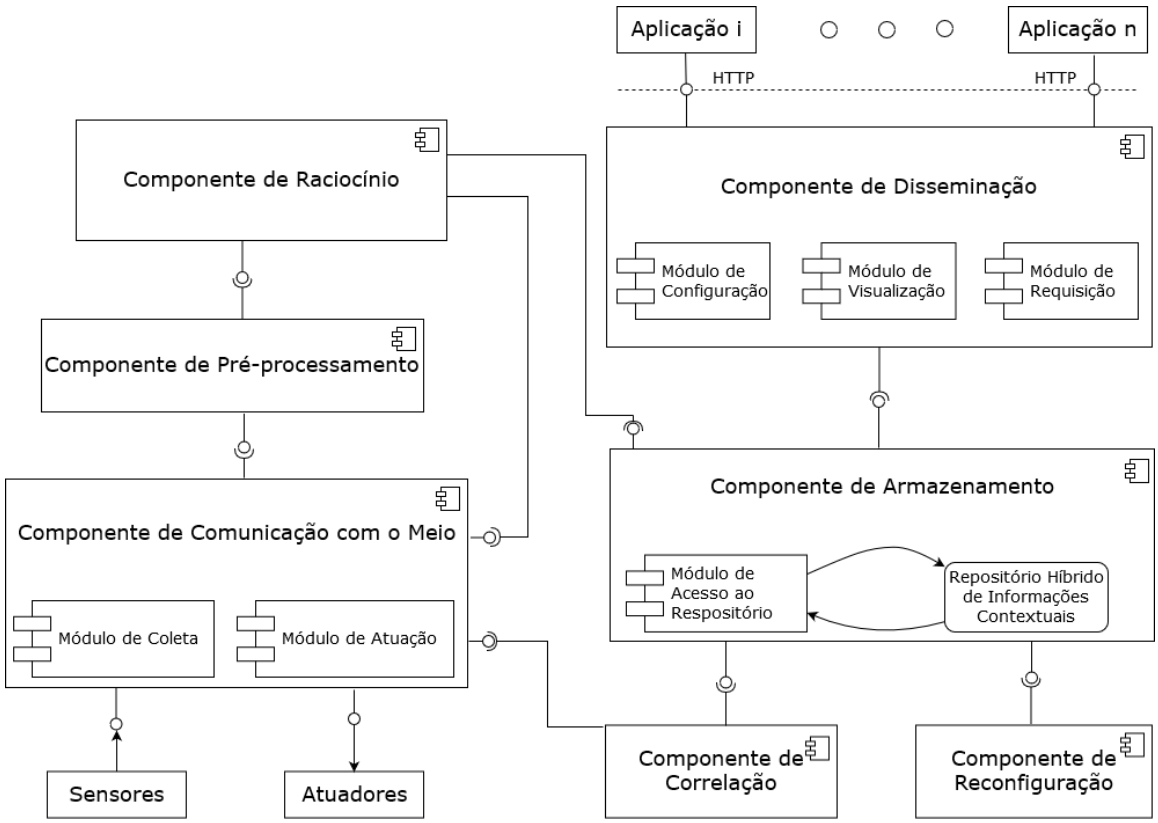

Figura 14 – Organização arquitetural da DyCoRS.

O Componente de Raciocínio recebe os dados de contexto já normalizados pelo Componente de Pré-processamento e realiza o processamento dos mesmos, após os envia para serem persistidos no Componente de Armazenamento. Além disso, após o processamento, podem ser realizadas requisições de atuação para o Componente de Comunicação com o Meio. O Componente de Correlação permite correlacionar as informações contextuais presentes nos diferentes modelos de armazenamento e caso seja necessário, realiza a requisição de atuação para o Componente de Comunicação com o Meio. O Componente de Disseminação disponibiliza para as aplicações cientes de contexto uma interface de manipulação das informações presentes no Compo-

nente de Armazenamento, as quais podem ser tanto as informações utilizadas para execução da DyCoRS, bem como aquelas que foram coletadas e identificadas pela arquitetura. O Componente de Reconfiguração disponibiliza uma abordagem para manipular em tempo de execução as informações exploradas pela DyCoRS no seu funcionamento.

A seguir é apresentada uma discussão dos Componentes arquiteturais da DyCoRS.

5.1.1 Componente de Comunicação com o Meio

O Componente de Comunicação com o Meio foi projetado para realizar a comunicação com os sensores e atuadores de interesse da aplicação. Este Componente provê suporte para o procedimento de coleta agendada dos dados contextuais, como também para obtenção instantânea dos mesmos por solicitação das aplicações.

Este Componente possui dois módulos: Coleta e Atuação. O Módulo de Coleta realiza a aquisição de eventos de diferentes fontes, lidando tanto com sensores lógicos (arquivos de log, consumo de memória, CPU, entre outros) como também com sensores físicos (temperatura, umidade, batimentos cardíacos, entre outros). Além disso, esse Componente pode receber eventos de diferentes dispositivos, funcionando como um servidor, empregando diferentes protocolos, por exemplo, syslog e *Hypertext Transfer Protocol* (HTTP).

O registro de dados de contexto obtidos considera os seguintes parâmetros: (i) o intervalo de tempo entre medições; e (ii) a flutuação mínima do valor coletado.

O Módulo de Atuação executa as ações configuradas com base nas situações identificadas. Estes procedimentos compreendem comandos de ativação e desativação dos atuadores, bem como de parametrização de seu comportamento operacional. As ações realizadas pelo Módulo de Atuação podem ser de dois tipos:

- envio de alertas via e-mail, *Short Message Service* (SMS) ou por um serviço de mensagens instantâneas, sendo informada a situação identificada e a sugestão da ação a ser realizada;
- execução de comandos que, entre outras opções, podem atuar no dispositivo ou em dispositivos remotos, resultando na adaptação do ambiente em tempo de execução.

As requisições para realização de atuação são provenientes dos Componentes de Raciocínio e de Correlação após a detecção de situações de interesse identificada por estes componentes.

5.1.2 Componente de Pré-processamento

O Componente de Pré-processamento foi projetado devido à necessidade de normalização dos dados de contexto coletados pelo o Componente de Comunicação com o Meio. Esta necessidade se dá devido a premissa da DyCoRS de lidar com dados de diferentes naturezas, enquanto consequência do suporte à diferentes tipos de aplicações.

Além disso, esta normalização contribui para uma melhor visualização dos dados, quando disponibilizadas pelo Componente de Disseminação ás diferentes aplicações.

Para tanto, este Componente executa a separação dos dados coletados em diferentes campos e pode adicionar novas informações contextuais. Para o uso no Componente de Pré-processamento, gramáticas foram desenvolvidas com base em formatos de contexto predefinidos. Como consequência, os dados coletados são separados em campos. Além disso, novos dados contextuais podem ser adicionados, por exemplo, dados relacionados à geolocalização de um endereço *Internet Protocol* (IP).

O Componente de Pré-processamento pode ser configurado para realizar uma primeira filtragem nos dados coletados, por exemplo, identificar e ignorar dados inválidos emitidos por algum sensor. Assim, só são repassados para o Componente de Raciocínio dados considerados válidos e que podem ser processados em busca de situações de interesse.

5.1.3 Componente de Raciocínio

O Componente de Raciocínio foi projetado para operar de forma sinérgica com outros recursos da arquitetura, oferecendo às diferentes aplicações, que usem os serviços oferecidos pela DyCoRS, uma maior flexibilidade no atendimento de suas demandas de processamento contextual. A principal funcionalidade é realizar as tarefas pertinentes ao processamento das informações contextuais, permitindo identificar as situações de interesse das aplicações.

Este Componente foi concebido para proporcionar o uso de uma abordagem dinâmica para raciocínio de contexto. Esta abordagem dinâmica faculta que as aplicações possam escolher qual ou quais estratégias querem utilizar no processamento contextual e o Componente de Raciocínio se responsabiliza por realizar a composição das mesmas.

Cada técnica para raciocínio de contexto é catalogada no repositório com um *Uniform Resource Identifier* (URI) único, possibilitando assim o acesso a técnica específica que a aplicação ciente de contexto deseja utilizar. O Componente de Raciocínio já tem instanciadas três das principais estratégias empregadas na literatura (PERERA et al., 2014):

- baseada em regras seguindo o padrão evento-condição-ação;
- baseada em lógica descritiva, explorando ontologias. Esta estratégia pode ser usada de duas formas, na primeira, os axiomas internos à ontologia são utilizados para inferir conhecimento sobre as classes representadas na ontologia. Na segunda forma, podem ser aplicadas regras externas escritas na linguagem SWRL, permitindo ao sistema inferir novas informações de contexto sobre as instâncias da ontologia;
- baseada em aprendizagem supervisionada que disponibiliza diversas técnicas para serem usadas, sendo que após o processo de treinamento uma decisão rápida pode ser emitida.

O Componente de Raciocínio por demanda das aplicações poderá ter outras estratégias para raciocínio contextual instanciadas. A composição dinâmica destas estratégias para raciocínio tem por base uma metalinguagem que foi concebida para manipular regras composicionais, explorando de forma sinérgica os recursos da arquitetura DyCoRS, o que viabiliza a utilização de diferentes estratégias para raciocínio de forma combinada. Entende-se que a metalinguagem proposta é uma contribuição central do Componente de Raciocínio

A metalinguagem possui *tags* de marcação que começam com o símbolo "#" seguido de um identificador e um número. Os identificadores previstos para as regras composicionais são:

M: representa uma regra composicional;

A: caracteriza uma estratégia de aprendizagem supervisionada;

O: retrata uma estratégia baseada em lógica descritiva;

C: simboliza um comando a ser executado.

Além dos identificadores, podem ser usados operadores lógicos (*and, or, not*) para integrar a lógica usada na regra composicional. Para dar suporte ao emprego da abordagem dinâmica de raciocínio, foram concebidas tabelas para o banco de dados relacional presente no Componente de Armazenamento. A Figura 15 apresenta o diagrama Entidade Relacionamento (ER), mostrando os conceitos da abordagem composicional e como foi projetada sua modelagem, permitindo assim a sua concepção.

Considerando a abordagem composicional, foi criada uma entidade intitulada de "Raciocinio" para modelar a regra composicional a ser utilizada, explorando para isto um atributo "id", conforme pode ser visualizado na Figura 15. O componente de Raciocínio identifica qual regra será utilizada buscando pelo respectivo "id" e analisando o valor do atributo "regra" referente a regra composicional que será aplicada.

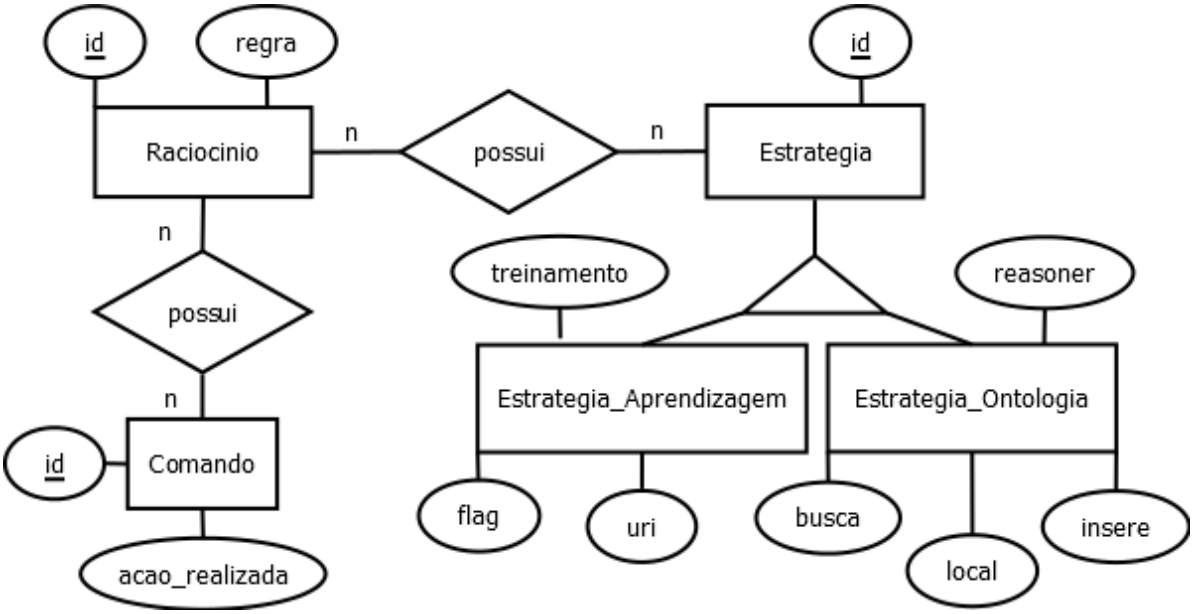

Figura 15 – Diagrama ER das regras composicionais de raciocínio

Por sua vez, para atender a premissa que na DyCoRS o raciocínio de contexto pode ser composto de diversas estratégias, foi modelada a entidade “Estrategia”, a qual disponibiliza um atributo “id”, que nesta entidade é responsável por diferenciar as estratégias. Como pode ser ser visto na Figura 15, as estratégias baseadas em aprendizagem e em ontologias são modeladas em entidades independentes, sendo especializações de “Estrategia”.

Esta abordagem de concepção de separar as categorias de estratégias para raciocínio, teve como motivação uma melhor utilização do espaço de armazenamento, devido a diversidade de atributos inerentes a cada estrategia. Além disso, essa abordagem facilita a inclusão de novas categorias de estratégias para raciocínio. A seguir, uma caracterização das entidades utilizadas para modelar as categorias de raciocínio baseadas em aprendizagem e ontologias é apresentada.

- A entidade “Estrategia_Aprendizagem” possui o atributo “uri” que é responsável por identificar a técnica de aprendizagem. Além deste, essa entidade possui o atributo “treinamento”, o qual especifica qual conjunto de treinamento deve ser utilizado para treinar a técnica de aprendizagem escolhida. Ainda, foi modelado um atributo nomeado de “flag” que possui a finalidade de especificar a necessidade de realizar o treinamento da técnica escolhida, assim, quando for necessário realizar treinamento seu valor será “1”, caso contrário “0”;

- A entidade “Estrategia_Ontologia” possui o atributo “reasoner” que especifica qual instrução deve ser executada para realizar o raciocínio da ontologia. Os atributos “busca” e “insere”, são referentes as consultas SPARQL que devem ser executadas para, respectivamente, buscar os dados presentes na ontologia

e inserir os dados no modelo de triplas do Componente de Armazenamento. Ainda, essa entidade possui o atributo “local”, o qual representa o local no disco onde se encontra o arquivo da ontologia utilizada.

Ainda na Figura 15, pode ser visualizada a entidade “Comando”, a qual possui o atributo “id” que é responsável pela identificação dos comandos, sendo o valor deste atributo utilizado na regra composicional para identificar qual o comando que deve ser executado. Além deste, a entidade possui o atributo “acao_realizada”, o qual especifica as ações que devem ser realizadas dependendo da situação que foi identificada. Sendo que estas ações são repassadas para o Componente de Comunicação com o Meio que é responsável por executar a ação necessária. Os comandos são representados nas regras composicionais pela *tag* “#C” seguido de um identificador, o qual representa o atributo “id” da entidade “Comando”.

Considerando esta modelagem, o Componente de Raciocínio realiza a execução da regra composicional de acordo com um determinado fluxo de execução, como pode ser visto na Figura 16. Inicialmente, o Componente de Raciocínio obtém a regra composicional inicial no Componente de Armazenamento, após, ele identifica se existem outras regras composicionais presentes na mesma, representadas pelo identificador “M”. Na sequência, são identificadas as categorias das estratégias que são utilizadas na regra composicional. Caso sejam utilizadas técnicas de aprendizagem, representadas pelo identificador “A”, busca-se o atributo “uri” com a URI para utilização da técnica. Além disso, o Componente de Raciocínio busca qual conjunto deve ser utilizado para treinamento da técnica, sendo analisado o valor do atributo “flag” com o intuito de verificar a necessidade de execução do treinamento.

Caso seja identificado o identificador “O”, que se refere a utilização do raciocínio baseado em lógica descritiva, o Componente de Raciocínio busca o atributo “local”, onde está contido o local para busca da ontologia para processamento dos dados. Após, é identificado o mecanismo de *reasoner* a ser utilizado, sendo representado pelo atributo “reasoner”. Além disso, é buscado o atributo “busca” que representa a consulta SPARQL que deve ser executada para consultar às instâncias inferidas durante o processo de *reasoner*. Por último, o Componente de Raciocínio identifica o atributo “insere” que representa a consulta SPARQL que irá inserir as novas instâncias no modelo de triplas do Componente de Armazenamento.

Após a identificação das técnicas utilizadas e seu respectivo processamento, o Componente de Raciocínio identifica o comando representado na regra composicional. Por fim, é realizada a execução da regra composicional e, caso seja necessário, envia-se uma notificação para o Componente de Comunicação com o Meio, informando a necessidade de atuação. Portanto, esta atuação identificada é definida pelo comando presente na regra composicional.

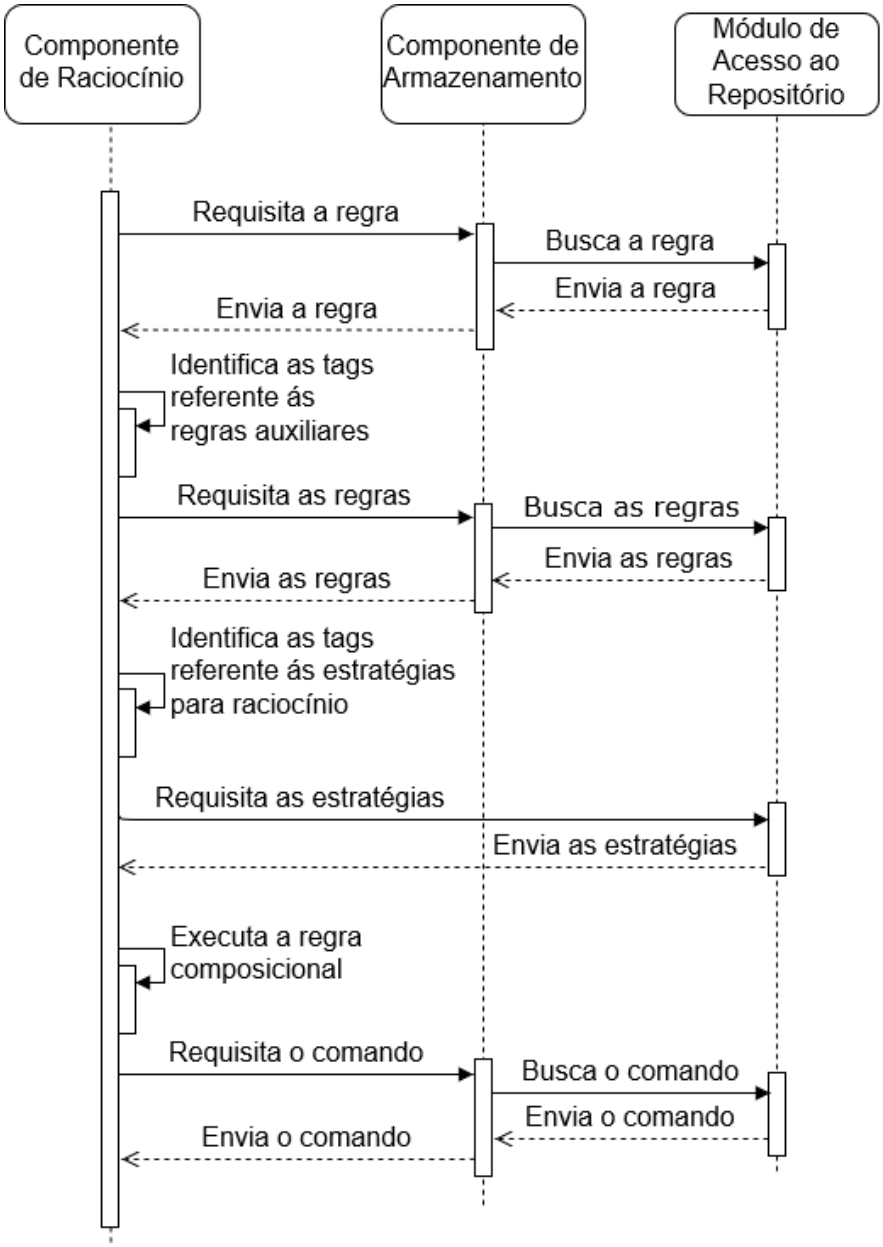

Figura 16 – Fluxo de execução do Componente de Raciocínio

5.1.4 Componente de Armazenamento

O Componente de Armazenamento foi projetado para permitir que a DyCoRS possa ter suporte às diferentes demandas de armazenamento que se fazem necessárias para as aplicações. Disponibilizar para as aplicações cientes de contexto diferentes modelos de persistência é uma característica importante para otimizar os procedimentos de armazenamento e recuperação das informações contextuais.

As informações contextuais tratadas pela DyCoRS são de diferentes naturezas, o que exige um tratamento diferenciado para cada tipo de dado. Para tanto foi concebido para este Componente o Repositório Híbrido de Informações Contextuais (RHIC), o qual oferece três modelos de armazenamento:

- relacional: possui um comportamento satisfatório em muitas situações, por exemplo, sendo usado para armazenar a relação entre a aplicação e o sensor monitorado, proporcionando fácil acesso à informação e facilitando a sua modificação quando necessário;

- não-relacional: possui como premissa o suporte para o tratamento de grandes volumes de dados. Este modelo faz melhor uso do espaço em disco ao armazenar dados semiestruturados em comparação com o modelo relacional (ROSA et al., 2015). Para implementação deste modelo foi escolhida a categoria orientada a documentos, a qual armazena estruturas de dados independentes na forma de árvores hierárquicas e auto descritivas, constituídas de mapas, coleções e valores escalares (SHARP et al., 2013);

- triplas: permite a manipulação de dados providos por ontologias de forma mais eficiente em comparação com sua manipulação em memória ou usando um modelo relacional, abordagens que são comumente vistas em outros trabalhos (CAN et al., 2017). Com a inserção desse modelo, espera-se prover facilidades similares às experimentadas nos sistemas gerenciadores dos modelos relacionais, no qual o acesso e a manipulação das informações contidas nas ontologias é realizado de forma automatizada.

Destaca-se que as ontologias são processadas antes de serem armazenadas no modelo de triplas, desta forma, no momento em que as aplicações desejarem consultar as informações presentes no modelo de triplas, o processo de inferência já estará realizado. Essa abordagem, resulta em uma melhora no tempo de resposta do sistema, já que o processo de inferência pode se tornar lento dependendo do número de instâncias e das regras presentes na ontologia a ser processada.

Ainda com relação a Figura 14, o Componente de Armazenamento conta ainda com o Componente de Armazenamento o Módulo de Acesso ao Repositório, o qual é responsável pela disponibilização de métodos que permitem o acesso às informações contextuais armazenadas no RHIC. Este módulo disponibiliza uma interface de interoperabilidade que permite busca, inserção e exclusão das informações contextuais presentes nos diferentes modelos do RHIC. Os métodos disponibilizados para tratamento das informações são apresentados na Figura 17, na qual pode ser visualizado que o Módulo possui três subclasses, sendo uma para cada modelo de armazenamento presente no RHIC.

Os métodos recebem por parâmetros a consulta separada em campos, com o intuito de facilitar a composição da mesma, sendo passadas somente as informações necessárias, tais como, variáveis de retorno e possíveis filtros que serão aplicados. Particularmente os métodos “executaSPARQL” e “executaSQL” recebem como parâmetro uma *string* que contém toda a consulta que deve ser executada.

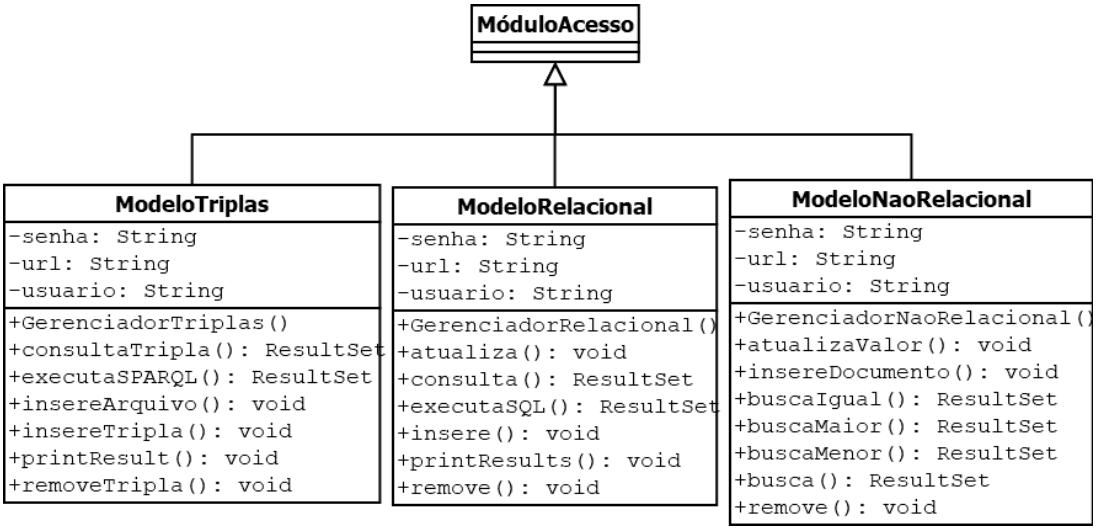

Figura 17 – Diagrama de classes do Módulo de Acesso ao Repositório

As aplicações cientes de contexto são responsáveis por determinar em qual modelo preferem armazenar as informações contextuais, sendo indicada a utilização do modelo relacional para dados que possuem relação entre eles, e assim, aproveitando-se das características do modelo. O modelo de triplas para armazenar as ontologias, de forma a propiciar facilidades para o tratamento das informações presentes nas mesmas. Enquanto o armazenamento no modelo não-relacional é recomendado quando é necessário o tratamento de grande volume de dados e/ou quando os dados armazenados possuem formatos variados.

5.1.5 Componente de Correlação

Este Componente foi concebido com a finalidade central de tratar históricos de informações contextuais. Com a utilização de diferentes modelos de armazenamento pela DyCoRS, esse Componente deve poder manipular todos os tipos de informações contextuais contidas no RHIC. Pensando nisso, o Componente de Correlação foi projetado para permitir a identificação de situações de interesse que precisam dos dados históricos que estão presentes nos diferentes modelos, por exemplo, de dados semânticos existentes no modelo de triplas e de dados semiestruturados contidos no modelo não-relacional.

O Componente de Correlação manipula as informações contextuais usando, assim como o Componente de Raciocínio, uma metalinguagem baseada em regras composicionais. Para permitir o uso de dados de diferentes modelos na mesma regra, é empregado o uso de *tags* de marcação que começam com o símbolo “#” seguido de um identificador e de um número. Quando os dados devem ser buscados no modelo relacional deve ser usado o identificador “R”, no modelo não-relacional “N” e, no modelo de triplas “T”.

Para dar suporte ao método de correlação foi necessário o projeto de tabelas no

banco de dados relacional presente no Componente de Armazenamento. A Figura 18 apresenta o diagrama ER referente aos conceitos da abordagem de correlação e como foi projetada sua modelagem, permitindo a sua concepção.

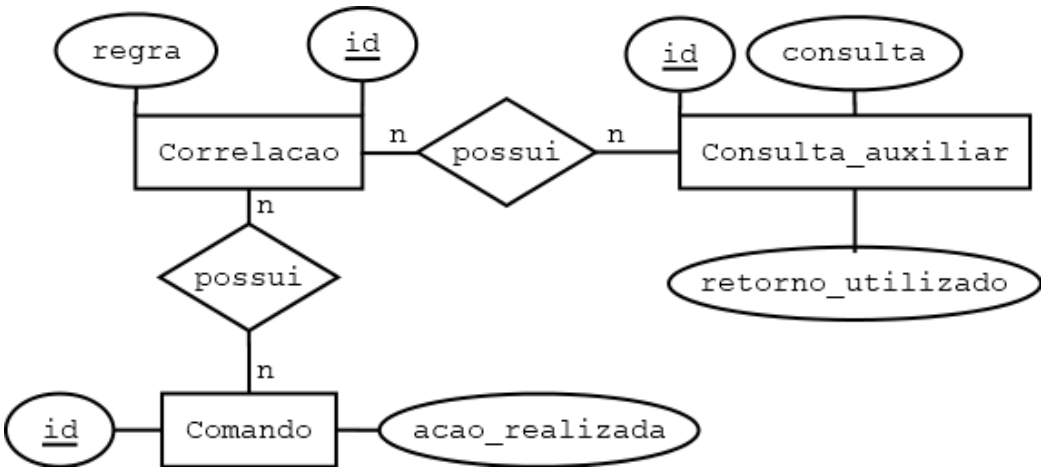

Figura 18 – Diagrama ER das regras de correlação

Conforme pode ser visualizada na Figura 18, foi criada uma entidade para modelar a abordagem de Correlação, intitulada de “Correlacao”, na qual está presente o atributo “id”, responsável por identificar as regras de correlação, sendo este atributo a chave primária da tabela. A DyCoRS consegue identificar qual regra será utilizada buscando pelo respectivo “id” e atributo “regra” referente a regra composicional que será aplicada. Destaca-se que cada regra de correlação pode ser composta de uma ou mais consultas auxiliares que podem ser realizadas nos diferentes modelos do Componente de Armazenamento. Além disso, as regras podem possuir um ou mais comandos de ações a serem utilizados após a identificação da situação de interesse.

As consultas auxiliares são modeladas em uma entidade específica intitulada de “Consulta_auxiliar”. Essa entidade possui o atributo “id”, o qual é o número utilizado para identificar as consultas auxiliares, e o atributo “consulta” que é referente a consulta de fato a realizada. Essa entidade também possui o atributo “retorno_utilizado”, o qual é o valor que será substituído na regra composicional no lugar da *tag* de marcação referente a consulta realizada. Quanto ao retorno das consultas auxiliares, um dos atributos que foram solicitados pode ser usado. Alternativamente, o método que verifica se a consulta encontrou algum resultado pode ser usado, retornando “true” se isso ocorrer e “false” caso contrário.

Pode ser visualizada, ainda na Figura 18, a entidade “Comando”, a qual é a mesma entidade apresentada na Figura 15. Essa entidade é utilizada tanto pelo Componente de Raciocínio quanto pelo de Correlação.

O Componente de Correlação realiza a execução da regra de correlação de dados de acordo com o fluxo de execução apresentado na Figura 19. Inicialmente, o Componente de Correlação obtém as regras de correlação no Componente de Ar-

mazenamento, após, as consultas auxiliares são identificadas e executadas, sendo realizadas as substituições das *tags* de marcação contidas na regra. Posteriormente, com a composição da regra com os valores necessários, sua avaliação é realizada, tendo como retorno o identificador da ação a ser executada. Por último, é enviada uma solicitação para o Componente de Comunicação com o Meio, informando a ação a ser realizada.

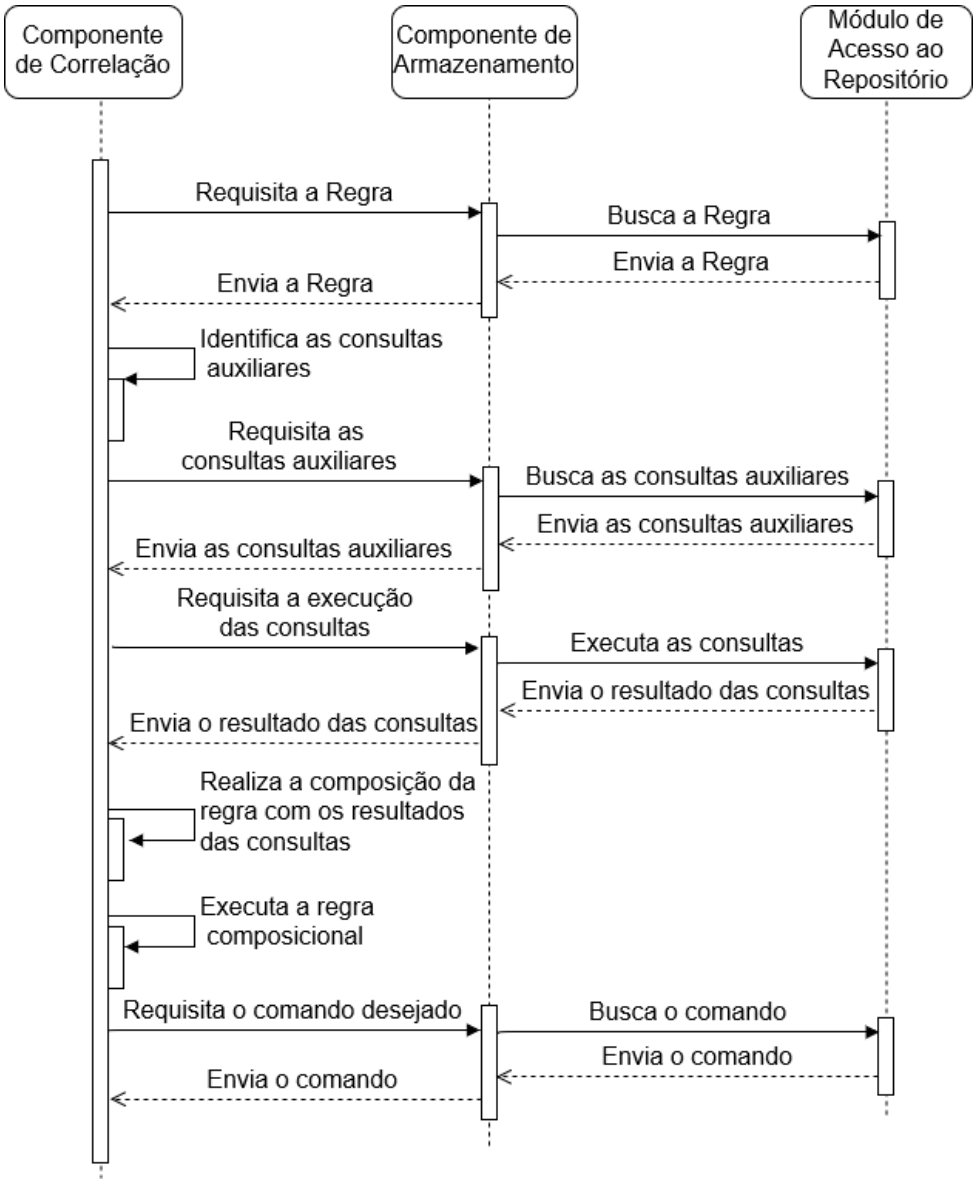

Figura 19 – Fluxo de execução do Componente de Correlação

Vale ressaltar que as regras de correlação utilizadas podem explorar dados armazenados em diferentes modelos, tornando esta abordagem composicional expressiva e, consequentemente, permitindo a DyCoRS identificar situações de interesse com uma maior flexibilidade.

5.1.6 Componente de Reconfiguração

O Componente de Reconfiguração foi projetado com a motivação de permitir a modificação do perfil operacional dos diferentes Componentes da DyCoRS em tempo de execução. Essa característica aumenta a flexibilidade da DyCoRS, permitindo que a mesma se adapte à mudanças de configuração realizadas, sem precisar parar os serviços oferecidos. Considerando isso, um exemplo de reconfiguração é a alteração da periodicidade da coleta dos dados que são monitorados pelo Componente de Comunicação com o Meio.

O Componente de Reconfiguração utiliza um processo de reconfiguração previamente descrito por meio de políticas, empregando uma abordagem baseada em regras pré-configuradas pelo desenvolvedor. Desta forma, após a reconfiguração, cada Componente passa a utilizar os novos parâmetros configurados, permitindo à DyCoRS se ajustar dinamicamente para melhor atender aos contextos de interesse da aplicação ciente de contexto.

A Figura 20 apresenta o diagrama ER referente as relações presentes na modelagem das informações necessárias para utilização das regras de reconfiguração.

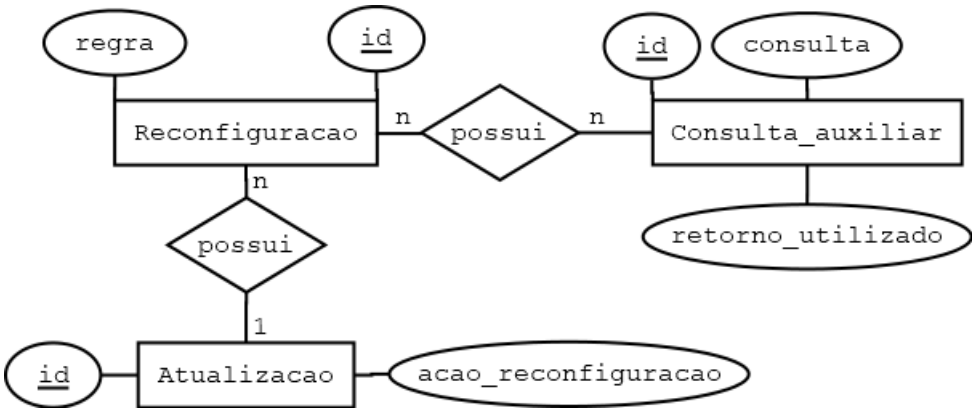

Figura 20 – Diagrama ER das regras de reconfiguração

Conforme pode ser observado na Figura 20, as regras de reconfiguração possuem consultas auxiliares e uma atualização que vai ser executada caso seja necessário. A entidade “Consulta_auxiliar” é a mesma apresentada na Figura 18 e é utilizada pelo Componente de Correlação. A entidade “Atualizacao” possui um atributo “id” utilizado para identificar as ações de atualização e o atributo “acao_atualizacao” que representa a reconfiguração que de fato deve ser executada.

O Componente de Reconfiguração também emprega a metalinguagem concebida, baseada em regras composicionais usando *tags* de marcação. As regras de reconfiguração começam com o símbolo “#” seguido de um identificador e um número, permitindo assim reconhecer a consulta auxiliar desejada. Quando os dados são buscados no modelo relacional, usa-se o identificador “R”, no modelo não-relacional “N” e no modelo de triplas, “T”.

O Componente de Reconfiguração realiza a execução da regra de reconfiguração de acordo com o fluxo de execução apresentado na Figura 21. Inicialmente, as regras desejadas são obtidas no Componente de Armazenamento, após, as *tags* de marcação presentes nas regras são identificadas, sendo que cada *tag* é substituída por uma consulta auxiliar que irá buscar alguma informação no Componente de Armazenamento. Posteriormente, as consultas auxiliares que substituem as *tags* são executadas e seus retornos inseridos na regra.

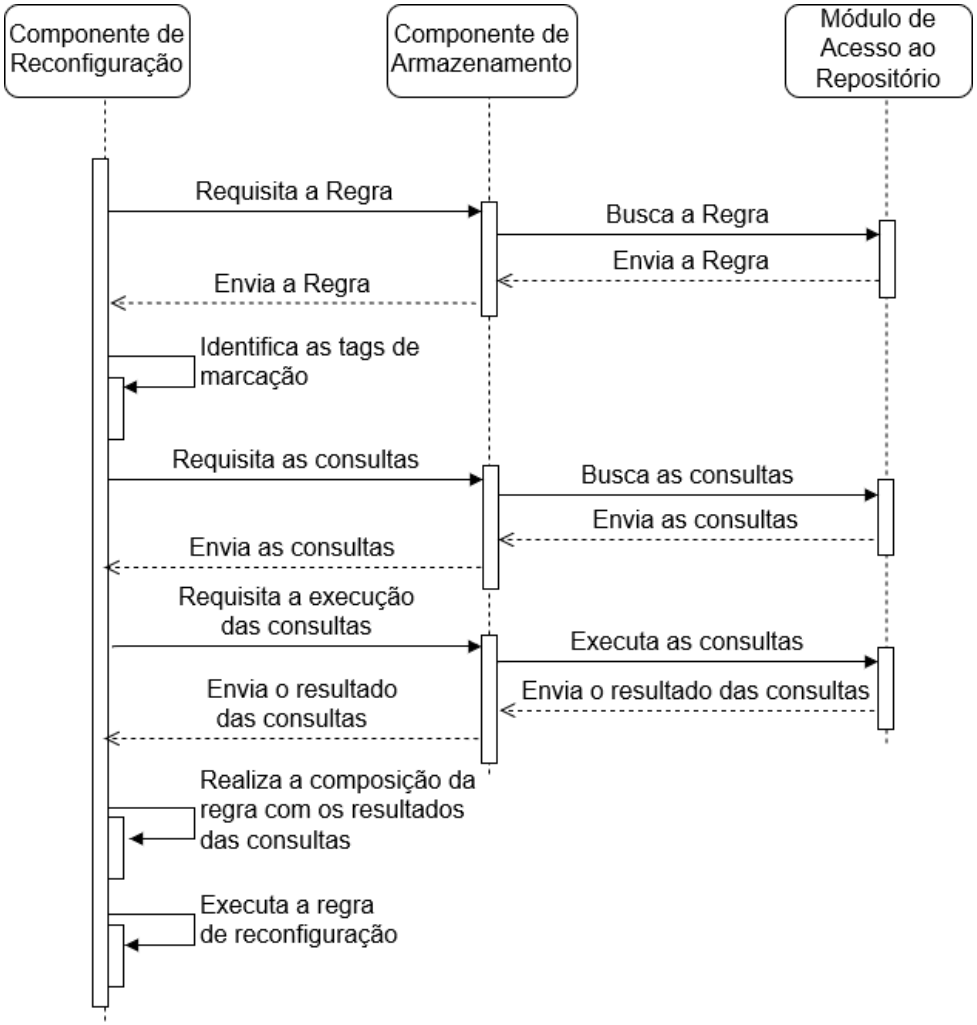

Figura 21 – Fluxo de execução do Componente de Reconfiguração

Com a inserção das informações na regra de reconfiguração sua avaliação é realizada, tendo como consequência a execução da ação de reconfiguração previamente estabelecida. Essa ação é a execução de uma operação de atualização no Componente de Armazenamento tendo como finalidade realizar alguma modificação no perfil operacional de execução da DyCoRS. As ações de reconfiguração são realizadas sempre no modelo Relacional do RHIC, já que é neste modelo que estão inseridos os parâmetros operacionais explorados na execução da DyCoRS.

5.1.7 Componente de Disseminação

O Componente de Disseminação foi projetado com o intuito de disponibilizar um mecanismo que facilite a comunicação das aplicações cientes de contexto com a DyCoRS. Assim, são disponibilizadas interfaces que permitem a configuração dos recursos oferecidos pela DyCoRS, bem como a visualização e manipulação das informações contextuais coletadas e das situações identificadas.

Este Componente possui o Módulo de Requisição, que recebe as requisições de acesso aos dados, as quais informam quais dados deverão ser monitorados. Além desse, possui também o Módulo de Configuração que disponibiliza um método para facilitar a configuração dos serviços cientes de contexto que serão providos. Com o uso da interface provida por esse Módulo são especificadas as regras composicionais que serão utilizadas no raciocínio dos dados contextuais. Além disso, é configurado onde serão armazenados os contextos adquiridos e quais regras de correlação serão exploradas no Componente de Correlação.

O Componente de Disseminação possui ainda o Módulo de Visualização, sendo este responsável por possibilitar a visualização das informações contextuais que foram capturadas e estão armazenadas no Componente de Armazenamento. As informações podem ser visualizadas em formato tabular, permitindo análise dos dados brutos, como também por meio de gráficos, que permitem identificar padrões de comportamento com base nos dados capturados.

5.2 Análise Comparativa com Trabalhos Relacionados

A Tabela 5 apresenta uma comparação das principais características da DyCoRS com os trabalhos relacionados apresentados na Seção 4.2.

Conforme pode ser visualizado na Tabela 5, a concepção da DyCoRS, na etapa de modelagem de contexto, explora a utilização do modelo relacional juntamente com o modelo baseado em ontologias, considerando a tendência observada na literatura no emprego de modelos híbridos para representação de contexto.

Analisando a estratégia de raciocínio empregada pelos trabalhos relacionados destaca-se que, apesar destes utilizarem diferentes estratégias, as mesmas são exploradas, em sua maioria, de forma individual, ou em uma combinação estática. Enquanto que a DyCoRS usa uma abordagem dinâmica, na qual as aplicações podem escolher quais estratégias querem empregar no raciocínio do contexto, podendo utilizar as estratégias tanto de forma individual quanto combinada, considerando os diferentes momentos da aplicação e os respectivos contextos de interesse.

Destaca-se ainda que diferentes formas de armazenamento são providas pela DyCoRS, possibilitando a persistência nos modelos: relacional; triplas; e não-relacional. Enfatiza-se que diferentemente dos trabalhos apresentados, a DyCoRS possui uma

Tabela 5 – Comparação entre os trabalhos relacionados e a DyCoRS.

Trabalhos	Aquisição	Modelo	Raciocínio	Armazenamento	Avaliação	Ferramentas
AGOSTINI; BETTINI; RIBONI, 2009	Sensores Físicos Lógicos	Ontológico	Lógica Descritiva e Regras	Informação Não Disponível	Cenários de Uso	Protégé, Racer
RIBONI; BETTINI, 2011	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Informação Não Disponível	Cenários de Uso	Java, Weka, Protégé, RacerPro
XU et al., 2011	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Relacional	Cenários de Uso	Informação Não Disponível
LOPES et al., 2012	Sensores Físicos Lógicos	Ontológico e Relacional	Lógica Descritiva e Regras	Relacional	Cenários de Uso	Java, Jena, OSGI, Protégé, Pellet, PostgreSQL
STROBBE et al., 2012	Sensores Físicos Lógicos	Ontológico	Baseado em Casos e Lógica Descritiva	Informação Não Disponível	Cenários de Uso	Java, Jena, OWLAPI, jCOLIBRI
YUAN; HERBERT, 2014	Sensores Físicos Lógicos	Fuzzy	Baseado em Casos e Lógica Fuzzy	XML	Cenários de Uso	jCOLIBRI
BAZ; YORUK; CETIN, 2016	Informação Não Disponível	Informação Não Disponível	Aprendizagem Supervisionada e Lógica Probabilística	Informação Não Disponível	Cenários de Uso	VLFEAT toolbox
RAZZAQ; AMIN; LEE, 2017	Sensores Físicos Lógicos	Ontológico	Aprendizagem Supervisionada e Lógica Descritiva	Jena TDB (triple store)	Cenários de Uso	Java, Maven, Jena, Protégé, Pellet, OWL-API, Weka
LI; MARTÂNEZ; RUBIO, 2017	Sensores Físicos Lógicos	Ontológico	Lógica Descritiva e Lógica Probabilística	Informação Não Disponível	Cenários de Uso	Pellet
DyCoRS	Sensores Físicos Lógicos	Ontológico e Relacional	Aprendizagem Supervisionada, Lógica Descritiva e Regras	Relacional, Não Relacional e Triplas	Cenários de Uso	Java, Python, Protégé, Jena, Pellet, scikit-learn, PostgreSQL, Virtuoso e MongoDB

estratégia que permite correlacionar dados contextuais armazenados em diferentes modelos.

Como pode ser visualizado, a DyCoRS emprega o mesmo método de avaliação dos trabalhos relacionados, sendo a sua avaliação baseada em cenários. Particularmente, quando da utilização de técnicas de aprendizagem supervisionada, são exploradas bases de dados de amplo uso pela comunidade científica, tanto na etapa de treinamento, como também para análise de acurácia das técnicas empregadas.

Por fim, também estão descritas na Tabela 5 as ferramentas utilizadas na concepção dos trabalhos relacionados e da arquitetura DyCoRS. Na seleção de ferramentas se optou pelo uso daquelas que vêm se destacando na literatura. Para manipulação de modelos ontológicos optou-se por utilizar o *framework* Jena[1] e o Pellet[2], estas ferramentas já vêm sendo empregadas com sucesso pelo grupo para estas finalidades. Por sua vez, para a disponibilização de estratégias de aprendizagem explorou-se o scikit-learn[3].

Enquanto sistema para gerenciamento do modelo de triplas optou-se pela escolha do Virtuoso[4], o qual possui desempenhos satisfatórios em estudos do grupo disponíveis na literatura (ROSA et al., 2019), destacando-se pela escalabilidade demonstrada. Já o MongoDB[5] foi escolhido para ser utilizado como sistema para gerenciamento do modelo não-relacional, o qual se destacou, em trabalho anterior do grupo (ROSA et al., 2015), pelo seu bom desempenho. Enquanto o PostgreSQL[6] foi escolhido para ser utilizado como sistema para gerenciamento do modelo relacional, sobretudo considerando o seu histórico de emprego no grupo de pesquisa, e a perspectiva de manter compatibilidade com outros trabalhos já desenvolvidos.

5.3 Considerações do Capítulo

Considerando as lacunas identificadas nos trabalhos relacionados discutidos no Capítulo 4, destaca-se como diferencial da DyCoRS ser dinâmica no provimento de recursos para a obtenção da ciência de contexto. Deste modo, com a arquitetura proposta é possível prover uma maior flexibilidade para as aplicações escolherem os parâmetros operacionais que serão explorados pelos Componentes da DyCoRS.

Particularmente em relação ao processamento dos dados contextuais, com o uso da abordagem híbrida para raciocínio de contexto, é disponibilizada para as aplicações a possibilidade de combinarem de forma dinâmica, pelo emprego de regras composicionais, diferentes estratégias para raciocínio.

Além disso, com as regras de correlação é possível tratar informações armazenadas em diferentes modelos de armazenamento, e, por sua vez, com as regras de reconfiguração é viabilizada a possibilidade de modificar os parâmetros operacionais da DyCoRS em tempo de execução.

[1] https://jena.apache.org/
[2] https://www.w3.org/2001/sw/wiki/Pellet
[3] http://scikit-learn.org/stable/
[4] http://virtuoso.openlinksw.com/
[5] https://www.mongodb.org/
[6] http://www.postgresql.org/

6 AVALIAÇÃO E DISCUSSÃO DOS RESULTADOS

Dentre os diversos métodos de avaliação discutidos na literatura da área de ciência de contexto, este trabalho optou pelo emprego de uma avaliação baseada em cenários. Destacam-se como principais razões para esta escolha os seguintes critérios: (i) ser considerada uma alternativa madura para ser empregada em ambientes cientes de contexto na UbiComp (PATIDAR; SUMAN, 2015); (ii) ser explorada para avaliar a utilização de uma arquitetura de software em relação às demandas de um conjunto de aplicações (SHANMUGAPRIYA; SURESH, 2012); (iii) ser utilizada como método de avaliação pelos trabalhos relacionados à arquitetura DyCoRS.

Para uma melhor apropriação de como a avaliação da DyCoRS ocorreu, as diferentes ferramentas de software selecionadas ao longo da realização do trabalho de pesquisa são descritas na próxima seção.

6.1 Principais Ferramentas Utilizadas

Esta seção é dedicada à apresentar as principais características das ferramentas consideradas ao longo do desenvolvimento do trabalho, e seu respectivo impacto na prototipação da DyCoRS. Dentre os critérios empregados na escolha das ferramentas, destaca-se o seu amplo uso pela comunidade acadêmica, bem como estarem categorizadas como software livre.

PostgreSQL

O PostgreSQL é um sistema gerenciador de banco de dados objeto-relacional de código aberto coordenado pelo PostgreSQL Global Development Group. Dentre as suas principais características pode se destacar: (i) compatibilidade com a maioria dos sistemas operacionais; (ii) atendimento das propriedades de atomicidade, consistência, isolamento e durabilidade; (iii) suporte para a utilização de chaves estrangeiras, *joins, views, triggers* e procedimentos armazenados; (iv) compatibilidade com várias linguagens de programação (SCHERER; JACOBSEN; SANTOS, 2008).

MongoDB

O MongoDB é um sistema gerenciador de banco de dados não-relacional orientado a documentos, criado em 2007, desenvolvido na linguagem C++. Os documentos podem ser considerados análogos aos registros em um banco de dados relacional, e as operações de inserir, atualizar e excluir podem ser executadas em uma coleção. Uma coleção pode armazenar vários documentos, funcionando de forma similar à uma tabela do modelo relacional. Uma coleção pode armazenar documentos de diferentes estruturas devido ao fato do MongoDB ser um banco de dados sem esquema (FILHO, 2015).

Virtuoso

Virtuoso é um sistema gerenciador de banco de dados híbrido que combina a funcionalidade de um banco de dados relacional, objeto relacional, XML e triplas RDF. O software é projetado para tirar vantagem do suporte de segmentação do sistema operacional e de múltiplos processadores. O armazenamento de dados é realizado utilizando uma quadrupla, na qual além de armazenar a tripla básica (sujeito, predicado e objeto), também é armazenado o grafo relacionado. Com isso o Virtuoso consegue trabalhar com múltiplos grafos de forma simultânea (RAZA, 2019).

Jena

O *framework* Jena é disponibilizado na linguagem Java e fornece uma abstração e interface para manipulação de ontologias. Foi desenvolvido por pesquisadores do laboratório de pesquisa da *Hewlett-Packard* (HP), o HP Labs, tendo sido seu projeto incubado pela Apache em 2010. O *framework* fornece classes e interfaces para criação e manipulação de ontologias baseadas em RDF e *Ontology Web Language* (OWL), e possui um motor de inferência para OWL e RDF Schema (GOMES et al., 2019).

Pellet

O Pellet é implementado em Java e oferece uma variedade de recursos, incluindo atendimento por consulta conjunta, suporte a regras, raciocínio, entre outros. Para tornar seus recursos de raciocínio facilmente acessíveis aos usuários, o Pellet fornece várias interfaces, incluindo uma interface de linha de comando e *Application Programming Interfaces* (APIs) para os toolkits RDF/OWL Jena e Manchester OWL-API (SIRIN et al., 2007).

Scikit-learn

O Scikit-learn é um módulo Python que integra uma ampla gama de algoritmos de aprendizagem, disponibilizando tanto técnicas de aprendizagem supervisionadas

quanto não supervisionadas. Este pacote se concentra em facilitar o uso de técnicas de aprendizagem para não especialistas usando uma linguagem de alto nível e de propósito geral. A ênfase é colocada na facilidade de uso, desempenho, documentação e consistência da API (PEDREGOSA et al., 2010). Ele tem dependências mínimas e é distribuído sob a licença *Berkeley Software Distribution* (BSD) simplificada, incentivando seu uso em ambientes acadêmicos e comerciais.

6.2 Cenários de Uso

Com o intuito de avaliar as funcionalidade providas pela DyCoRS foram projetados e implementados protótipos de três cenários de uso. Esses cenários consideram a premissa de aumentar a flexibilidade para as aplicações, explorando particularidades referentes ao emprego combinado de diferentes estratégias para raciocínio de contexto, provendo assim uma detecção de situações mais rica.

6.2.1 Cenário 1 - Computação Aplicada à Saúde

Esse cenário é baseado no trabalho de Dias (2017), no qual a DyCoRS é empregada para realizar o monitoramento de sinais vitais de pacientes e gerar alertas precoces com a utilização de escores. A aplicação de monitoramento de sinais vitais utiliza o Componente de Disseminação para realizar a configuração da DyCoRS, escolhendo como será realizado o raciocínio dos dados contextuais e quais estratégias serão empregadas. Além disso, pode explorar o Componente de Disseminação para visualização dos dados coletados, bem como das situações identificadas.

O Componente de Comunicação com o Meio, nesse cenário, é responsável por receber os sinais vitais de monitores multiparamétricos comerciais ou de sensores de sinais vitais individualizados. Desta forma, podem ser monitorados diferentes sinais vitais, tais como, a saturação do oxigênio e os batimentos cardíacos de um paciente.

Para realizar o processamento dos sinais vitais coletados são utilizadas regras composicionais. Essas regras são formadas de uma estratégia de aprendizagem supervisionada com o uso de duas técnicas de regressão. Desta forma, a DyCoRS possibilita prever possíveis situações de colapso de um paciente e assim antecipar o acionamento do profissional de saúde para a tomada de ação.

A Tabela 6 apresenta os valores analisados neste cenário para gerar o alerta precoce com base nos sinais vitais obtidos. Com a utilização da DyCoRS as regras utilizadas para a geração dos escores podem ser customizáveis, bem como os parâmetros analisados pela mesma, permitindo assim que as regras utilizadas sejam adequadas a diferentes padrões de comportamento dos pacientes. A Figura 22 mostra uma visão geral dos Componentes da DyCoRS e o necessário tratamento das informações contextuais para atender esse cenário.

Tabela 6 – Escore de alerta precoce com base nos valores dos sinais vitais.

Sinal Vital	Escore: 3	Escore: 2	Escore: 1	Escore: 0	Escore: 1	Escore: 2	Escore: 3
Frequência Cardíaca		≤ 40	41 – 50	51 – 100	101 – 110	111 – 129	≥ 130
Saturação O_2	< 85	85 – 89	90 – 92	≥ 93			

Fonte: Adaptado de DIAS, 2017, p. 54.

Conforme pode ser visualizado na Figura 22, o Componente de Comunicação com o Meio é responsável por receber os dados dos sinais vitais. Após a coleta, os dados são normalizados pelo o Componente de Pré-processamento, repassando somente os dados válidos para o Componente de Raciocínio. Após o processamento pelo Componente de Raciocínio os dados coletados, bem como as situações identificadas são armazenadas no Componente de Armazenamento. Por fim, caso seja necessária a execução de alguma atuação, o Componente de Comunicação recebe a requisição do Componente de raciocínio e realiza a atuação.

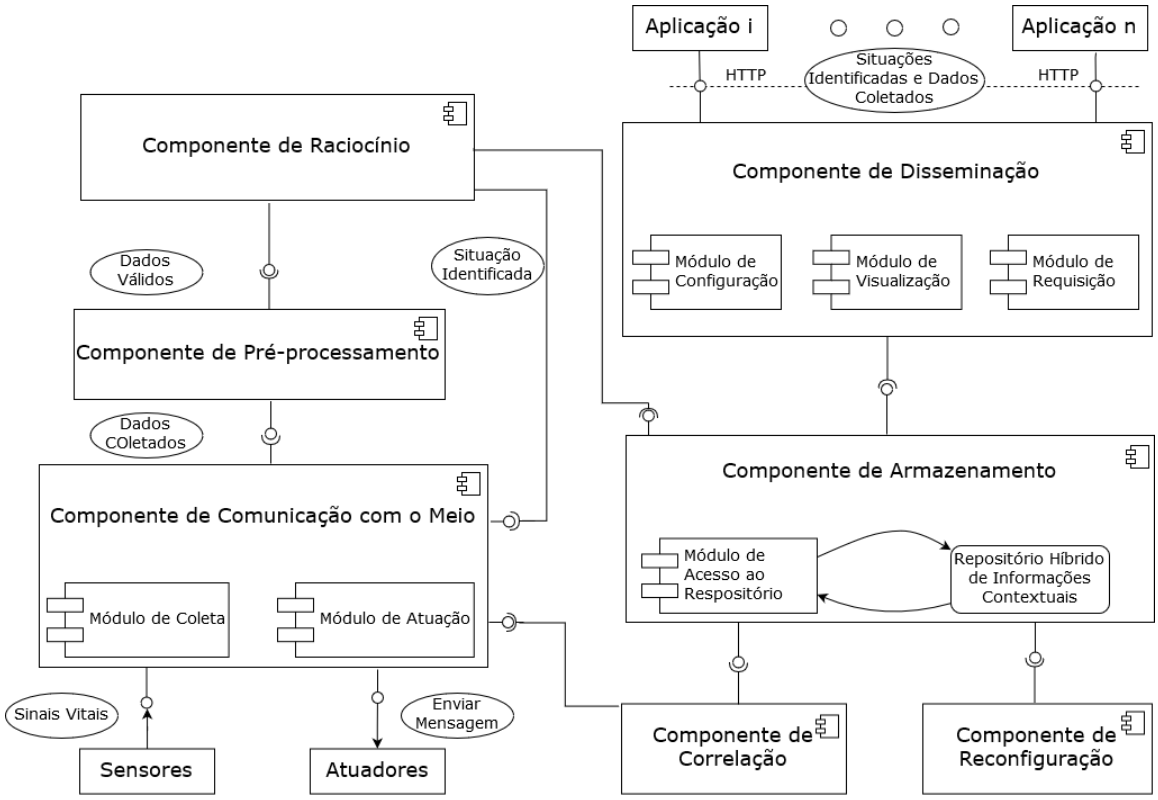

Figura 22 – Visão geral do tratamento das informações realizado no cenário 1.

A Figura 23 apresenta a regra composicional empregada nesse cenário, a qual é formada de outras duas regras, que são representadas por #M1 e #M2. Além disso, esta regra possui um condicional e um comando que deve ser executado caso a condição seja avaliada como verdadeira.

Como primeiro passo para execução da regra composicional, é necessário que o

```
#M1
#M2
if((ews_cardiaca==2 and ews_saturacao==2) or (ews_cardiaca==3 or ews_saturacao==3)):
        #C1(nome, “critico”)
```

Figura 23 – Regra composicional utilizada no cenário 1.

Componente de Raciocínio realize a identificação das *tags* de marcação contidas na regra a ser executada. Desta forma, são identificadas as *tags* referentes as regras auxiliares M1 e M2, as quais são apresentadas na Figura 24, fazendo-se necessário a busca pelas regras representadas pelos *ids* com valores 1 e 2 na tabela Raciocínio (vide Figura 15), e a substituição das *tags* contidas na regra composicional. Após, é necessário verificar se as regras que foram buscadas possuem outras regras representadas por uma *tag* #M ou se possuem alguma *tag* #A referente a técnica de aprendizagem ou #O para a estratégia baseada em ontologia.

Com a verificação realizada, são identificadas as *tags* #A1 e #A2, as quais podem ser visualizadas na Figura 24 assim percebe-se a necessidade de buscar quais são as técnicas presentes na tabela Estrategia_Aprendizagem que possuem os *ids* com os valores 1 e 2. Após a busca, é então realizada a substituição das *tags* pelas URIs referentes as técnicas empregadas. Como são utilizadas estratégias de aprendizagem é realizada a concatenação da URI com a *string predict*, desta forma, permitindo realizar a predição do dado passado por parâmetro.

A Figura 24 apresenta as regras auxiliares #M1 e #M2. Na subfigura (a) é mostrada a regra M1, sendo esta empregada para tratar o sinal vital frequência cardíaca. Já a subfigura (b) mostra a regra M2, a qual trata os dados providos pelo sinal vital saturação de O_2.

```
if #A1<130:
        ews_cardiaca=3
elif #A1<=40 or (#A1>=111 and #A1<=129):
        ews_cardiaca=2
elif (#A1>=41 and #A1<=50) or (#A1>=101 and #A1<=110):
        ews_cardiaca=1
elif #A1>=51 and A1<= 100:
        ews_cardiaca=0
```

(a) Regra referente a M1.

```
if #A2 <85:
        ews_saturacao=3
elif #A2 >=85 and #A2 <=89:
        ews_saturacao=2
elif #A2 >=90 and #A2 <=92:
        ews_saturacao=1
elif #A2 >=93:
        ews_saturacao=0
```

(b) Regra referente a M2.

Figura 24 – Regras composicionais referentes a M1 e M2

A Figura 25 mostra a regra auxiliar M1 já composta da técnica de aprendizagem referente a *tag* #A1, na qual pode ser visualizada a escolha da técnica de aprendizagem

linear Lars[1]. Essa regra é referente a geração de alertas com base nos dados coletados do sinal vital frequência cardíaca, sendo gerado os valores do escore analisado na regra composicional.

```
if lars.predict(dado)<130:
        ews_cardiaca=3
elif lars.predict(dado)<=40 or (lars.predict(dado) >=111 and lars.predict(dado) <=129):
        ews_cardiaca=2
elif (lars.predict(dado)>=41 and lars.predict(dado)<=50) or (lars.predict(dado)>=101 and lars.predict(dado)<=110):
        ews_cardiaca=1
elif lars.predict(dado)>=51 and lars.predict(dado)<= 100:
        ews_cardiaca=0
```

Figura 25 – Regra M1 com a técnica referente a tag A1.

Na Figura 26 é mostrada a regra auxiliar M2 já composta da técnica de aprendizagem referente a *tag* #A2, na qual pode ser visualizada a escolha da técnica de aprendizagem linear Lasso[2]. Essa regra é referente a geração de alertas com base nos dados coletados do sinal vital saturação O_2, sendo gerados os valores do escore analisado na regra composicional.

```
if lasso.predict(dado)<85:
        ews_saturacao=3
elif lasso.predict(dado)>=85 and lasso.predict(dado)<=89:
        ews_saturacao=2
elif lasso.predict(dado)>=90 and lasso.predict(dado)<=92:
        ews_saturacao=1
elif lasso.predict(dado)>=93:
        ews_saturacao=0
```

Figura 26 – Regra M2 com a técnica referente a tag A2.

É importante notar que é passado o mesmo parâmetro para as duas técnicas de raciocínio, isso se deve ao fato que neste cenário é realizada a predição sempre do próximo valor esperado para aquele sinal. Os valores dos sinais vitais são passados para cada técnica durante a etapa de treinamento, o que nesse cenário é realizado a cada nova coleta dos sinais vitais.

A Figura 27 apresenta a regra composicional aplicada no cenário, na qual todas as *tags* de marcação já foram substituídas, resultando na combinação da estratégia baseada em regras e da estratégia de aprendizagem. Além destas, o comando a ser executado caso a regra seja avaliada como verdadeira também já foi inserido no lugar da *tag* #C1. Este comando é referente ao envio de uma mensagem pelo aplicativo de

[1] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lars.html
[2] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html

mensagem instantânea para o profissional de saúde responsável, sendo repassado na mensagem o nome do paciente e a situação de predição crítica.

```
if lars.predict(dado)<130:
        ews_cardiaca=3
elif lars.predict(dado)<=40) or (lars.predict(dado)>=111 and lars.predict(dado)<=129):
        ews_cardiaca=2
elif (lars.predict(dado)>=41 and lars.predict(dado)<=50) or (lars.predict(dado)>=101 and lars.predict(dado)<=110):
        ews_cardiaca=1
elif lars.predict(dado)>=51 and lars.predict(dado)<=100:
        ews_cardiaca=0
if lasso.predict(dado)<85:
        ews_saturacao=3
elif lasso.predict(dado)>=85 and lasso.predict(dado)<=89:
        ews_saturacao=2
elif lasso.predict(dado)>=90 and lasso.predict(dado)<=92:
        ews_saturacao=1
elif lasso.predict(dado)>=93:
        ews_saturacao=0
if((ews_cardiaca==2 and ews_saturacao==2) or (ews_cardiaca==3 or ews_saturacao==3)):
        sender.send_msg(str(medico).decode("utf-8"), str('Paciente Estado Critico').decode("utf-8"))
```

Figura 27 – Regra aplicada para geração de alertas no cenário 1.

Com o intuito de analisar a predição realizada pelas técnicas de aprendizagem supervisionada exploradas nesse cenário, utilizou-se medições disponibilizadas pela base de dados *Multiparameter Intelligent Monitoring in Intensive Care*[3] (MIMIC). A base de dados MIMIC inclui dados de mais de 90 pacientes registrados na Unidade de Terapia Intensiva (UTI), coletados entre os anos de 1992 e 1999, contendo em média 40 horas de registros de sinais fisiológicos (GOLDBERGER et al., 2000).

A verificação do comportamento da DyCoRS para tratar a demanda desse cenário foi baseada na métrica de precisão, a qual é definida como a proporção de instâncias classificadas corretamente. O uso dessa métrica é justificado pelo fato de ser amplamente empregada para avaliar a qualidade dos resultados em diversas pesquisas que utilizam técnicas de aprendizagem supervisionada (JAPKOWICZ; SHAH, 2011).

Nesse cenário a previsão realizada é considerada correta, quando a técnica prevê de forma acertada qual o escore que a próxima leitura de sinal vital possuirá. Com o intuito de analisar a precisão da estratégia de aprendizagem realizando a predição da categoria do próximo valor do sinal vital, foram utilizadas instâncias de dez pacientes da base de dados MIMIC. A Tabela 7 apresenta as instâncias dos sinais vitais frequência cardíaca e saturação de O_2 de cada paciente. As instâncias estão divididas nas categorias de escore e o número total de dados analisados.

A Tabela 8 apresenta a precisão alcançada pelas técnicas de aprendizagem empregadas pela regra composicional, tanto para frequência cardíaca quanto para a sa-

[3]https://www.physionet.org/physiobank/database/mimicdb/

Tabela 7 – Instâncias presentes em cada categoria dos sinais vitais.

	Frequência Cardíaca					Saturação O_2				
Paciente	Escore=0	Escore=1	Escore=2	Escore=3	Total	Escore=0	Escore=1	Escore=2	Escore=3	Total
055	56906	58703	9019	20	124648	109576	218	119	36	109949
211	74490	570	141	0	75544	35363	5844	300	279	41786
221	79557	3249	488	0	83294	83294	0	0	0	83294
226	20497	48226	32260	10847	111830	75368	16781	9116	1014	102279
230	66602	22	33	7	66664	30714	4401	1537	253	36905
248	102061	11664	5088	21	118834	55998	403	3	47	56451
252	98742	0	0	0	98742	73395	14599	4297	992	93283
253	145354	275	0	0	145629	101860	32573	1902	643	136978
401	83932	2	0	0	83934	43651	21465	11270	61	76447
403	149324	0	0	0	149324	50801	79050	17532	62	147445

turação de O_2. Nota-se que em algumas células da tabela está contido o caractere “-”, o qual representa que para aquele paciente nenhuma instância daquela categoria foi analisada.

Tabela 8 – Precisão das estratégias no cenário 1.

	Frequência Cardíaca					Saturação O_2				
Paciente	Escore=0	Escore=1	Escore=2	Escore=3	Total	Escore=0	Escore=1	Escore=2	Escore=3	Total
055	0.996	0.993	0.979	0.9	0.993	0.999	0.848	0.907	0.972	0.999
211	0.999	0.875	0.836	-	0.997	0.998	0.989	0.936	0.971	0.996
221	0.998	0.945	0.956	-	0.995	1.0	-	-	-	1.0
226	0.966	0.977	0.985	0.995	0.979	0.998	0.988	0.989	0.993	0.996
230	0.999	0.5	0.696	0.571	0.999	0.997	0.980	0.989	0.992	0.995
248	0.982	0.778	0.832	0.476	0.956	0.999	0.950	0.666	0.957	0.999
252	1.0	-	-	-	1.0	0.998	0.987	0.985	0.991	0.995
253	0.999	0.92	-	-	0.999	0.998	0.993	0.978	0.984	0.996
401	0.999	0.5	-	-	0.999	0.997	0.989	0.990	0.868	0.993
403	1.0	-	-	-	1.0	0.996	0.995	0.990	0.951	0.995

Conforme pode ser observado na Tabela 8, a precisão das técnicas de aprendizagem supervisionada foi satisfatória, alcançando mais de 95% de acerto no total de instâncias classificadas de cada paciente. Nota-se que em alguns casos foram acertadas todas as predições, sendo que esses casos aconteceram para os pacientes que possuem todas as instâncias da mesma categoria de escore. As piores taxas de precisão obtidas foram de 50%. Foi analisado que estas são categorias que possuíam poucas instâncias, e ainda, são de casos que os sinais vitais mudam rapidamente

entre as categorias de escore dificultando assim o acerto por parte da técnica de aprendizagem.

6.2.2 Cenário 2 - Segurança da Informação

O presente cenário tem como foco a identificação de situações de interesse na área de segurança da informação, mais especificamente, objetiva identificar tentativas de ataque à uma rede de computadores.

Para a concepção desse cenário, procurou-se identificar na literatura a tendência de solução para enfrentar este tipo de ataque. Dentre as abordagens identificadas destaca-se a análise do tráfego da rede, empregando para isso uma técnica de aprendizagem supervisionada (HE, 2014; AL-YASEEN; OTHMAN; NAZRI, 2017). Com o uso da DyCoRS é possível monitorar o tráfego da rede e processar as conexões realizadas. Para esse processamento podem ser empregadas diferentes estratégias para raciocínio, as quais podem ser combinadas com o uso da metalinguagem concebida.

A Figura 28 mostra uma visão geral dos Componentes da DyCoRS envolvidos e o necessário tratamento das informações contextuais para atender esse cenário. A aplicação de detecção de tentativas de ataque à rede utiliza o Componente de Disseminação para realizar a configuração da DyCoRS, escolhendo como será realizado o raciocínio dos dados contextuais e quais estratégias serão empregadas. Além disso, pode explorar o Componente de Disseminação para visualização dos dados coletados, bem como das situações identificadas.

Conforme pode ser visualizado na Figura 28, o Componente de Comunicação com o Meio fica responsável por coletar o tráfego da rede, funcionando como um *sniffer* na rede. Após as conexões serem coletadas elas são repassadas para o Componente de Pré-processamento, que realiza a normalização dos respectivos dados. Durante essa etapa, são realizadas substituições de dados no formato *string* para o formato numérico, por exemplo, substituindo o protocolo de transporte usado pelo seu respectivo número, contribuindo assim para operação da técnica de raciocínio explorada.

Após a normalização, como pode ser visto na Figura 28, os dados são repassados para o Componente de Raciocínio, o qual é responsável por categorizar a conexão coletada, podendo ser entendida como uma conexão normal ou estar entre uma das quatro categorias de ataque mais presentes na literatura. Para implementar essa categorização é utilizada a estratégia baseada em regras combinada com a estratégia de aprendizagem supervisionada, empregando duas técnicas de classificação.

A Figura 29 apresenta a regra composicional empregada para atender este cenário. Pode ser visualizado que a regra composicional é composta de duas estratégias de aprendizagem supervisionada, as quais são representadas por #A3 e #A4. Além disso, possui um comando que pode ser executado caso seja necessário.

Primeiramente, o Componente de Raciocínio realiza a identificação de *tags* de

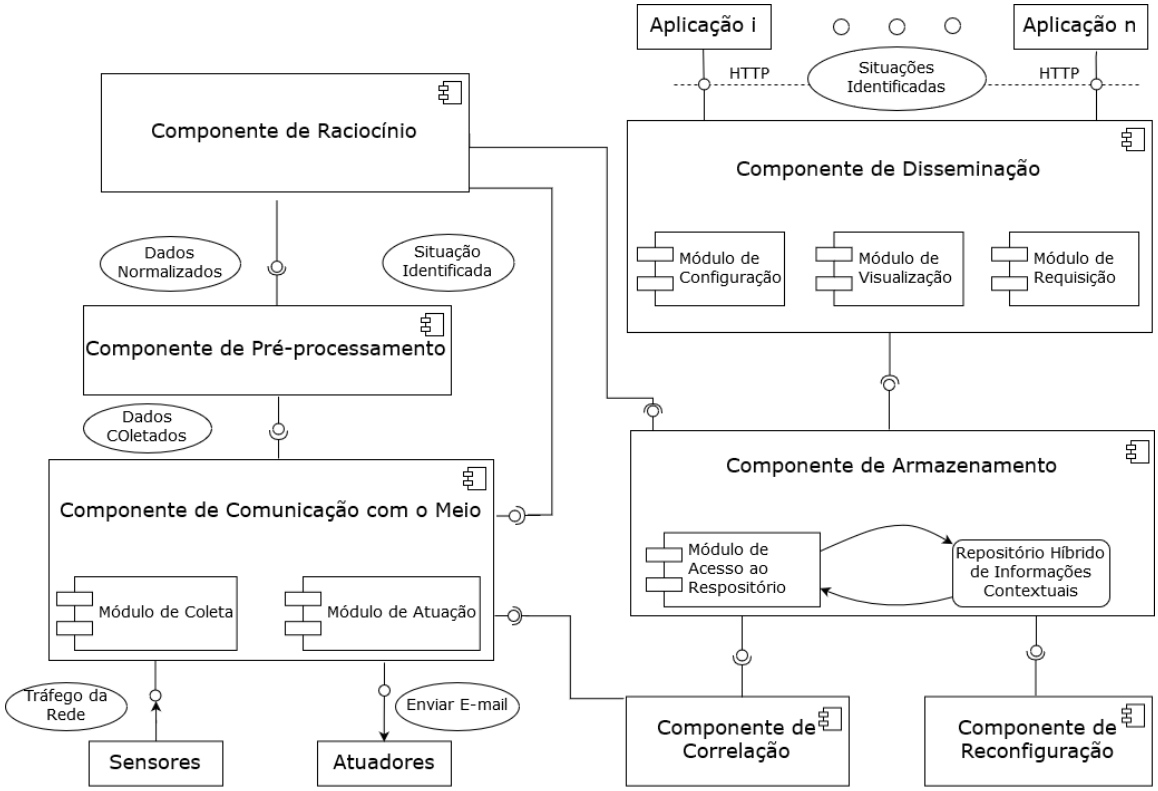

Figura 28 – Visão geral do tratamento das informações realizado no cenário 2.

```
if(#A3 != 0):
        ataque = #A4
        If(ataque == 1)
                #C2("Ataque DoS")
        If(ataque == 2)
                #C2("Ataque R2L")
        If(ataque == 3)
                #C2("Ataque U2R")
        If(ataque == 4)
                #C2("Ataque Probe")
```

Figura 29 – Regra composicional utilizada no cenário 2.

marcação contidas na regra composicional a ser executada. Desta forma, são identificadas as *tags* referentes as técnicas de aprendizagem supervisionada representadas pelas tags #A3 e #A4. Assim, se faz necessário buscar quais são as técnicas presentes na tabela Estrategia_Aprendizagem que possuem os *ids* com os valores 3 e 4. Após a busca, é então realizada a substituição das *tags* pelas URIs referentes as técnicas empregadas.

A *tag* #A3 é referente a técnica de aprendizagem *Random Forest*[4], enquanto a *tag*

[4]https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

#A4 referencia a técnica *Decision Tree*[5]. Como são utilizadas estratégias de aprendizagem é realizada a concatenação da URI com a *string predict*, desta forma, permitindo realizar a predição do dado passado por parâmetro.

Na regra composicional criada para este cenário, a técnica referente a *tag* #A3 realiza um filtro nas conexões capturadas, categorizando-as em conexão normal com o valor 0 ou ataque retornando o valor 1. Com essa filtragem pode-se aumentar a taxa de acerto da categorização das conexões e diminuir as taxas de falso positivo e falso negativo.

A técnica representada pela *tag* #A4 é treinada para categorizar as conexões entre 4 categorias de ataques, sendo utilizada sempre que, durante a etapa de filtragem, detecta-se que a conexão é um ataque. Conforme pode ser visto na Figura 29 a técnica retorna 1 quando é um ataque *Denial of Service* (DoS), 2 quando é *Remote to Local* (R2L), 3 para *User to Root* (U2R), e por último, 4 para a categoria de ataque Probe.

A colaboração sinérgica dos resultados obtidos pelas estratégias representadas pelas *tags* #A3 e #A4 caracteriza uma situação típica de contribuição da DyCoRS. As avaliações realizadas, de forma análoga ao que se pode identificar na literatura, apontaram que em cenários com elevados volumes de dados contextuais, se faz necessário reduzir o escopo de processamento das técnicas de aprendizado.

A Figura 30 apresenta a regra composicional que é aplicada neste cenário, na qual todas as *tags* de marcação já foram substituídas, resultando na combinação da estratégia baseada em regras e da estratégia de aprendizagem supervisionada. Além destas, o comando a ser executado caso a conexão seja classificada como ataque também já foi inserido no lugar da *tag* #C2. Este comando é referente ao envio de um e-mail para o administrador da rede informando que foi identificado um ataque ao respectivo dispositivo, informando a categoria de ataque identificada.

É importante notar que é passado o mesmo parâmetro para as duas técnicas de aprendizagem, isso se deve ao fato que neste cenário é realizada a categorização da mesma conexão de rede coletada pelas duas técnicas. As técnicas são treinadas somente no início da execução tornando mais rápido o processo de categorização de uma nova conexão.

Para validação deste cenário, foi utilizada como fonte de contexto relacionado ao tráfego da rede a base de dados “KDD Cup 99 Data”, a qual é considerada umas das principais bases utilizadas na avaliação de mecanismos para detecção de tentativas de ataques a servidores de rede (ELEKAR; WAGHMARE; PRIYADARSHI, 2015). Para realizar o treinamento da árvore de decisão optou-se por utilizar o conjunto de treinamento *kddcup.data_10_percent*. O arquivo é formado de 494021 registros de conexão, contendo conexões já classificadas em cinco categorias de conexões, sendo uma de-

[5]https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

```
if(random.predict(conexao)  != 0):
        ataque = decision.predict(conexao)
        If(ataque == 1)
                s.sendmail(user,  admin, “Ataque DoS”)
        If(ataque == 2)
                s.sendmail(user,  admin, “Ataque R2L”)
        If(ataque == 3)
                s.sendmail(user,  admin, “Ataque U2R”)
        If(ataque == 4)
                s.sendmail(user,  admin, “Ataque Probe”)
```

Figura 30 – Regra aplicada no cenário 2.

las a categoria de conexão “Normal” e quatro categorias de ataques (DoS, R2L, U2R, Probe) (LEE; STOLFO; MOK, 1999).

Neste cenário só são utilizados cinco dos 41 atributos de cada conexão presente na base de dados. Assim, é possível realizar a coleta dos atributos necessários com o uso do Componente de Comunicação com o Meio, pois se fossem usados os outros atributos seria necessário um processamento extra para inferir os atributos. Com o intuito de testar o desempenho das técnicas empregadas, utilizou-se o conjunto de teste *corrected*, este conjunto é formado por 311029 conexões, sendo destas 60593 sendo conexões normais, e 250436 tentativas de ataques.

A Tabela 9 apresenta um resumo da validação realizada, na qual na primeira linha é mostrado o número de conexões presentes em cada categoria contida na base de dados utilizada. Na segunda linha é apresentado o número de conexões classificadas corretamente para cada categoria, enquanto que na terceira linha é exibida a taxa de acertos.

Tabela 9 – Precisão das estratégias no cenário 2.

	Normal	DoS	R2L	U2R	Probe
Conexões na base de dados	60593	229853	16189	228	4166
Conexões classificadas corretamente	59778	223774	1809	51	2832
Taxas de acerto	98.65%	97,35%	11,17%	22,36%	67,97%

Destaca-se que na Tabela 9 os resultados referentes a conexão normal foram calculados com a técnica de aprendizagem *Random Forest* aplicada na etapa de filtragem. Já os resultados das categorias de ataque são referentes a técnica de aprendizagem *Decision Tree* usada somente para identificar a categoria do ataque de uma

conexão.

Conforme pode ser visualizado na Tabela 9, para as categorias R2L e U2R a performance do classificador foi consideravelmente inferior as outras, isso se deve ao fato de a base de dados possuir um número menor de conexões dessas categorias no conjunto de treinamento. Assim, resultando na dificuldade por parte das técnicas para aprender a classificar de forma correta as conexões dessas categorias.

6.2.3 Cenário 3 - Ambiente de Vivência Assistida

O presente cenário possui como foco a identificação de situações de interesse em ambientes de vivência assistida, a qual propicia a concepção de aplicações voltadas a amparar o usuário em suas diferentes particularidades comportamentais.

A instanciação deste cenário foi baseada no projeto CASAS (COOK, 2012), o qual aborda os problemas e riscos associados a se ter um idoso que resida sozinho. Nesse cenário é abordada a situação em que o idoso sai de casa e pode demorar para retornar. Quando da identificação desta situação um familiar deve ser avisado para uma possível ação. A Figura 31 mostra uma visão geral dos Componentes da DyCoRS envolvidos e o necessário tratamento das informações contextuais para atender esse cenário.

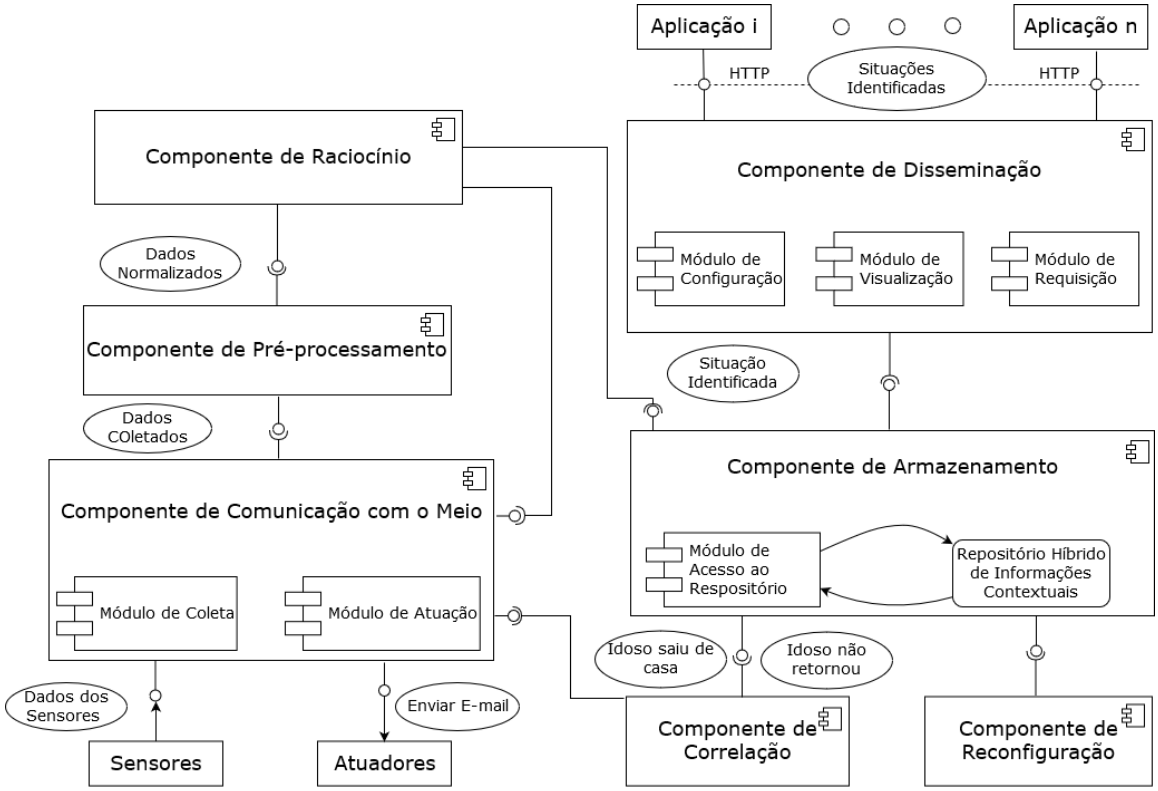

Figura 31 – Visão geral do tratamento das informações realizado no cenário 3.

A aplicação explorada neste cenário utiliza o Componente de Disseminação para realizar a configuração da DyCoRS, escolhendo como será realizado o raciocínio dos

dados contextuais, a regra de correlação explorada. Além disso, pode usar o Componente de Disseminação para visualização dos dados coletados, bem como das situações identificadas.

Como fonte de contexto nesse cenário são utilizados os valores gerados pelos sensores de presença binário disponíveis na residência, os quais são monitorados pelo Componente de Comunicação com o Meio como pode ser visto na Figura 31. Após a aquisição dos dados contextuais, os mesmos passam por um processo de normalização dos dados pelo Componente de Pré-processamento, sendo realizada a padronização dos valores para serem utilizados como entrada para o Componente de Raciocínio.

Para realizar o raciocínio destes dados contextuais é empregada no Componente de Raciocínio uma estratégia baseada em lógica descritiva explorando uma ontologia. A ontologia utilizada possui 258 axiomas, 51 classes, 6 objetos de propriedades e 11 propriedades de dados, maiores informações podem ser encontradas em Machado et al. (2019) . A identificação das situações de interesse é realizada com o uso de regras SWRL por meio da relação entre a localização simbólica e o objeto para reconhecer qual atividade o indivíduo está realizando.

A ontologia utilizada neste cenário não possui suporte para identificar a situação em que o idoso sai de casa e levou um tempo maior que o esperado para voltar. Deste modo, optou-se por explorar o Componente de Correlação tratando as situações identificadas ao invés de estender a ontologia. A motivação central nesta decisão está em procurar manter os custos com o processamento contextual o menor possível, sendo mantidos na ontologia somente os dados atuais. Desta forma, quando se faz necessário explorar dados históricos utiliza-se o Componente de Correlação para tratar as informações contextuais armazenadas. Assim, são utilizadas as identificações de duas situações de forma separada, a de entrar em casa e a de sair de casa, e após a identificação das duas situações as mesmas são tratadas com o uso do Componente de Correlação.

Os dados coletados são inseridos na ontologia, possibilitando seu processamento no raciocínio ontológico. O Componente de Raciocínio deve ser configurado para disparar o *reasoner*, por exemplo, a cada 10 minutos, sendo esse intervalo de tempo dependente da natureza da aplicação. Após esse processo, todas as instâncias contidas na ontologia são enviadas para o Componente de Armazenamento para serem persistidas no modelo de triplas, mantendo o histórico dos eventos coletados e das situações identificadas.

Após o armazenamento, as situações identificadas podem ser correlacionadas com o uso do Componente de Correlação, permitindo identificar a situação desejada nesse cenário, que é a ausência do idoso por um intervalo significativo de tempo. A Figura 32 apresenta a regra de correlação explorada nesse cenário.

A *tag* de marcação#T1 é referente a consulta SPARQL que busca, no modelo de Triplas do Componente de Armazenamento, se foi detectada a situação de o idoso sair de casa. Enquanto que a *tag* #T2 é referente a consulta SPARQL responsável por buscar, também, no modelo de Triplas, se a situação de o idoso entrar em casa foi detectada. Essa situação leva em consideração uma janela de tempo com base no horário que foi identificada a situação de o idoso sair da casa.

```
if ( #T1 && !#T2 ) #C3
```

Figura 32 – Regra de correlação usada para identificar a situação de interesse.

As *tags* de marcação são substituídas pelo retorno do método que verifica se a consulta teve algum retorno. O método retorna “true” caso positivo e “false” caso contrário. Destaca-se que na regra composicional criada o retorno da consulta #T2 é negado, ou seja, para a regra ser avaliada como verdadeira é necessário ser detectada a situação que o idoso sair da residência e que não seja identificada a situação que o idoso entrar em casa.

Caso a regra de correlação seja avaliada como verdadeira, o Módulo de Atuação realiza a execução do comando #C3, o qual envia uma mensagem para o aplicativo de mensagem instantânea de um familiar, avisando da ausência do morador por um tempo significativo, para que o mesmo providencie contato com o idoso.

Com o intuito de verificar a acurácia da estratégia baseada em lógica descritiva, foram analisadas as instâncias presentes na base de dados CASAS verificando a métrica de precisão, a qual é a proporção de instâncias classificadas corretamente. O uso dessa métrica é justificado pelo fato de a mesma ser amplamente empregada para avaliar a qualidade dos resultados em diversas pesquisas (JAPKOWICZ; SHAH, 2011). O número de ocorrências analisadas e a respectiva acurácia foram:

- entrar em casa: acertou 400 atividades de 431 presentes na base de dados, obtendo uma acurácia de 92,80%;
- sair de casa: acertou 405 atividades de 431 presentes na base de dados, obtendo uma acurácia de 93,96%;

Os resultados obtidos mostram que a operação combinada dos Componentes de Raciocínio e Correlação da DyCoRS contribuíram para não elevar o tempo de processamento contextual. Além disso, caracterizaram que a estratégia baseada em lógica descritiva alcançou resultados promissores para ser empregado no reconhecimento de atividades em ambientes de vivência assistida.

6.3 Considerações do Capítulo

O emprego de cenários de uso para avaliação da DyCoRS considerou tanto o aspecto desse método de avaliação ter seu uso consolidado por pesquisas análogas a DyCoRS, como também, a expectativa da arquitetura concebida de ser empregada para atender diferentes tipos de aplicação cientes de contexto.

Os resultados atingidos, apontam que proposta arquitetural da DyCoRS, combinada com o uso de regras composicionais, pode ser explorada com benefícios em aplicações de diferentes áreas.

Neste capítulo foram apresentados três cenários de uso, nos quais se fez necessário explorar diferentes características da DyCoRS. Foi possível observar que as aplicações podem se beneficiar da utilização de uma abordagem dinâmica para selecionar estratégias para raciocínio de contexto, do emprego de diferentes modelos de armazenamento, bem como da exploração de regras de correlação para dados históricos de contexto.

7 CONSIDERAÇÕES FINAIS

Tendo como motivação central a questão de pesquisa “Como deve ser uma proposta para ciência de contexto que viabilize uma combinação dinâmica de diferentes estratégias para raciocínio?”, foi concebida nesta Tese a arquitetura DyCoRS. Esta arquitetura tem como contribuição central a concepção de uma abordagem híbrida para raciocínio de contexto, possibilitando assim, a combinação dinâmica de diferentes estratégias para raciocínio, capacitando a DyCoRS ser empregada em diferentes domínios de aplicação.

A sistematização sobre sistemas cientes de contexto realizada nesta Tese permitiu identificar que apesar da disponibilidade na literatura de trabalhos atinentes a esta área, ainda existem desafios de pesquisa em aberto nas diferentes etapas que integram a ciência de contexto.

Dentre as etapas para obtenção da ciência de contexto, identificou-se que, particularmente, a etapa de processamento de contexto vem sendo amplamente discutida na literatura tendo como motivação a crescente complexidade das aplicações inerentes às infraestruturas computacionais atuais como a da IoT. Notou-se que apesar das diversas estratégias para raciocínio discutidas na literatura, o emprego de somente uma estratégia para atender aplicações de diferentes naturezas pode não ser suficiente, com isso as abordagens híbridas vêm ganhando destaque.

Como parte do esforço de definição das contribuições a serem alcançadas com o desenvolvimento desta Tese, identificou-se como necessário o levantamento do estado da arte com o foco em abordagens híbridas para raciocínio de contexto. Com esta motivação, foi disparado um esforço de avaliação da literatura disponível, a qual não encontrou trabalhos explorando a discussão das várias alterativas possíveis relacionadas à abordagens híbridas para raciocínio de contexto. Neste sentido, fez-se necessária a realização de uma RSL na área.

Para a realização da RSL foi feita uma avaliação do ano em que as publicações na área começaram a ganhar volume. Neste sentido, foram analisados os trabalhos publicados entre os anos de 2004 e 2018 em seis bases de indexação, sendo identificados 3241 documentos. Após a aplicação dos processos de filtragem foram identificados

nove trabalhos que propõem estratégias híbridas para o raciocínio contextual. Estes trabalhos foram descritos, sendo também realizada uma análise comparativa entre eles. Ainda, como contribuição alcançada, destaca-se a identificação de critérios que poderão ser utilizados por outros projetos, bem como das características exploradas pelos trabalhos selecionados nas etapas envolvidas para obtenção da ciência de contexto.

Enfatiza-se que os projetos analisados, em sua grande maioria, não tratam aspectos relacionados ao armazenamento das informações contextuais. Entretanto, destaca-se ser necessário lidar com o armazenamento das informações contextuais, devido a importância de manter o histórico do contexto, bem como permitir a disponibilização das informações capturadas sob demanda para aplicações cientes de contexto.

Ainda em decorrência da análise dos trabalhos relacionados, notou-se a ausência de uma abordagem que permita a combinação das diferentes estratégias para o raciocínio de contexto. Esses projetos, via de regra, exploram as estratégias de forma estática, não sendo oferecidas propostas para combinação dinâmica das mesmas. A premissa de pesquisa considerada foi que a concepção de uma proposta dinâmica poderá aumentar a flexibilidade para as aplicações combinarem as estratégias mais oportunas para tratarem os seus dados de contexto, contribuindo para uma maior acurácia para a identificação das situações de interesse.

Considerando estas lacunas na área de ciência de contexto, foi concebida, nesta Tese, a arquitetura intitulada de DyCoRS, a qual tem como diferencial ser dinâmica no provimento de recursos para a obtenção da ciência de contexto. Dentre estes recursos, destaca-se a concepção da abordagem híbrida para raciocínio de contexto com base em uma metalinguagem que permite combinar, de forma dinâmica, diferentes estratégias para raciocínio.

A decisão sobre o método para avaliação da DyCoRS levou em conta aqueles observados nos trabalhos relacionados, selecionados pela RSL realizada, bem como a tendência observada quando da revisão da literatura sobre ciência de contexto. Desta forma, foram contemplados três cenários de uso, os quais contemplam situações que traduzem aplicações reais na perspectiva da IoT, explorando diferentes recursos da arquitetura concebida.

No primeiro cenário de uso os recursos da arquitetura DyCoRS foram explorados com o intuito de possibilitar prever possíveis situações de colapso de um paciente, e assim antecipar o acionamento dos times de resposta rápida para a tomada de ação. Para tanto, foram processados os sinais vitais da frequência cardíaca e a saturação de O_2. Para implementar a geração de alertas foi explorada a metalinguagem concebida, a qual possibilitou combinar: uma estratégia baseada em regras; e uma estratégia de aprendizagem supervisionada com o uso de duas técnicas de regressão. Assim,

com a utilização da DyCoRS, diferentemente de outros trabalhos, foi possível combinar os escores gerados por diferentes sinais vitais e customizar as regras exploradas para a geração dos escores, podendo adequar as mesmas a diferentes padrões de comportamento dos pacientes.

No segundo cenário de uso a DyCoRS foi empregada para a identificação de situações de interesse na área de segurança da informação, mais especificamente, foram detectados possíveis ataques à rede. Para realizar o processamento dos dados contextuais coletados da infraestrutura de rede, foram utilizadas duas estratégias de aprendizagem supervisionada, as quais foram combinadas com a estratégia baseada em regras. Explorando os recursos da arquitetura, a primeira técnica foi aplicada para realizar um filtro nos dados contextuais capturados, classificando-os em conexão normal ou ataque, enquanto que a segunda é dedicada a processar as conexões classificadas como ataque, com o intuito de identificar o tipo de conexão analisado.

O terceiro cenário de uso teve como propósito a identificação de situações pertinentes a um ambiente de vivência assistida. Para realizar o raciocínio sobre os dados contextuais foi empregada uma estratégia baseada em lógica descritiva. Após o processamento, utilizou-se o Componente de Correlação da DyCoRS para correlacionar duas situações identificadas, permitindo o reconhecimento de uma nova possível situação.

7.1 Trabalhos Futuros

Considerando o esforço de estudo e pesquisa durante o desenvolvimento desta Tese, bem como as tendências na área de ciência de contexto identificadas quando da revisão de literatura, destacam-se as seguintes frentes enquanto alternativas para continuidade do trabalho:

- criar interfaces que utilizem técnicas de visualização de dados, com o intuito de facilitar a interpretação das informações contextuais armazenadas, bem como das situações que foram identificadas;
- ampliar as estratégias de raciocínio disponibilizadas, por exemplo, permitindo a utilização da estratégia de lógica fuzzy;
- explorar a abordagem de reconfiguração proposta, empregando a mesma para reconfigurar os recursos oferecidos em tempo de execução;
- aplicar a arquitetura DyCoRS em diferentes cenários de utilização, de modo que as diferentes aplicações se beneficiem da flexibilidade nos recursos oferecidos.

7.2 Publicações

Durante o desenvolvimento deste trabalho foram publicados artigos em periódicos e anais de congressos, em condição de primeiro autor. A seguir uma relação destes artigos contemplando seus resumos e dados da publicação

- State of the art in hybrid strategies for context reasoning: A systematic literature review. **Information and Software Technology, v. 111, p. 121-129, 2019.** Este artigo apresenta o estado da arte na área de ciência de contexto, empregando estratégias híbridas para raciocínio. É explorada uma revisão sistemática da literatura, analisando trabalhos publicados entre os anos de 2004 e 2018. Durante o processo de revisão foram identificados 3241 trabalhos, sendo selecionados ao final dez trabalhos que propõem estratégias híbridas. Os trabalhos selecionados são apresentados, discutidos, e comparados, sendo identificada a necessidade do desenvolvimento de estratégias para raciocínio que sejam dinâmicas, permitindo assim a escolha das estratégias que serão utilizadas por parte das aplicações cientes de contexto.

- An IoT Architecture to Provide Hybrid Context Reasoning. **2nd IFIP International Internet of Things (IOT), 2019.** Este artigo apresenta a concepção de uma arquitetura de software de IoT, fornecendo uma abordagem para o raciocínio de contexto híbrido baseado em regras composicionais, permitindo uma composição dinâmica de diferentes estratégias. Permitindo que as aplicações escolham entre diferentes estratégias de raciocínio, aquelas que são mais apropriadas, dependendo dos contextos que serão tratados.

- Reconhecimento de Atividades em Ambientes de Vivência Assistida: uma Proposta Explorando um Modelo Ontológico. In: **Seminário de Pesquisa em Ontologias no Brasil – ONTOBRAS 2019.** Este artigo apresenta a concepção de um modelo capaz de prover o reconhecimento de atividades executadas em um ambiente inteligente, sendo este modelo concebido com base em uma ontologia. O modelo proposto foi avaliado por meio de um estudo de caso utilizando uma base de dados de uma casa inteligente, tendo sido obtida uma acurácia média de 94.89% no reconhecimento das atividades.

- A Hybrid Architecture to Enrich Context Awareness through Data Correlation. **The 33rd ACM/SIGAPP Symposium On Applied Computing, SATTA - Software Architecture: Theory, Technology, and Applications Track, 2018.** Neste artigo é proposta uma arquitetura ciente de contexto para processar informações com base em modelos híbridos, a qual possui seis componentes: (i) aquisição de contexto; (ii) pré-processamento; (iii) processamento de contexto com uma estratégia de raciocínio híbrido; (iv) armazenamento de dados com suporte de três modelos de banco de dados; (v) comunicação com o repositório que permite o acesso a informações contextuais; e, (vi) abordagem de correlação baseada em regras de composição que permitem a combinação de dados armazenados em modelos distintos.

- Construção do Estado da Arte em Estratégias Híbridas para Raciocínio de Contexto: uma Abordagem explorando Revisão Sistemática da Literatura. In: **Congresso da Sociedade Brasileira de Computação, Simpósio Brasileiro de Computação Ubíqua e Pervasiva, SBCUP 2018.** Neste trabalho é construído o estado da arte na área de ciência de contexto, empregando estratégias híbridas para raciocínio. É explorada uma revisão sistemática da literatura, contribuindo tanto pela identificação de trabalhos relevantes na área, bem como pela especificação de critérios para sua seleção. Durante o processo de revisão foram identificados 2952 trabalhos, sendo selecionados ao final sete trabalhos que propõem estratégias híbridas. Os trabalhos selecionados são apresentados, discutidos, e uma comparação entre eles é realizada.

- Estado da arte em ambientes u-learning cientes de contexto: uma Revisão Sistemática da Literatura. In: **Congresso Brasileiro de Informática na Educação - Simpósio Brasileiro de Informática na Educação, 2018.** Esse artigo apresenta o estado da arte em ambientes u-learning cientes de contexto. Para isso, são analisados trabalhos publicados entre 2012 e 2017. Durante o processo de revisão foram identificados 217 artigos, sendo selecionados ao final cinco trabalhos que propõem ambientes u-learning cientes de contexto. Os trabalhos selecionados são apresentados, discutidos e analisados.

- EXEHDA-HM: A compositional approach to explore contextual information on hybrid models. **Future Generation Computer Systems, v. 73, p. 1-12, 2017.** Neste artigo é apresentada uma solução para lidar com o acesso e processamento de dados armazenados em modelos híbridos. A abordagem proposta se destaca pelo projeto de um repositório que suporta três modelos de banco de dados e uma estratégia de processamento composicional com base em regras. Com o uso da abordagem proposta, as aplicações podem combinar dados armazenados em bases diferentes em uma única regra, melhorando assim a identificação das situações de interesse.

- Ambientes Educacionais Ubíquos: uma Proposta Explorando Processamento Semântico para Ciência de Contexto. In: **Congresso Brasileiro de Informática na Educação - Simpósio Brasileiro de Informática na Educação SBIE 2017.** Este trabalho propõe uma arquitetura, para tornar a aprendizagem ubíqua mais próxima de uma perspectiva real. Para isso foi realizado o mapeamento e a integração do EXEHDA e o AdaptWeb, visando agregar suas funcionalidades e fornecer serviços de aprendizagem ubíqua. O comportamento da arquitetura proposta é avaliado por meio de um cenário de aplicação, o qual permite verificar sua aplicabilidade.

- $[SA]^2$: uma abordagem consciente de situação para segurança em infraestruturas computacionais. **Revista Brasileira de Computação Aplicada, v. 8, p. 89-103, 2016.** Este artigo apresenta uma abordagem ciente de situação para segurança de ambientes computacionais. A solução destaca-se pela arquitetura concebida para o fornecimento da ciência de situação, explorando diferentes funcionalidades desde a coleta, passando por um processamento híbrido de contexto, o armazenamento de dados contextuais e a decorrente atuação.

- Explorando Modelos Contextuais Híbridos: uma Abordagem Composicional. In: **Congresso da Sociedade Brasileira de Computação. SBCUP - Simpósio Brasileiro de Computação Ubíqua e Pervasiva, 2016.** Neste artigo é proposto uma estratégia de processamento baseada em regras composicionais, as quais possibilitam à aplicação combinar informações contextuais armazenadas em diferentes modelos em uma única regra, tendo como intuito facilitar a identificação de situações de interesse.

- Uma Abordagem Composicional Para Correlacionar Informações Contextuais Presentes em Modelos Híbridos. In: **Encontro de Pós-Graduação da Universidade Federal de Pelotas - ENPOS 2016.** Neste artigo é apresentada uma estratégia de processamento composicional de contexto, a qual possui como diferencial a capacidade de correlacionar as informações presentes em diferentes modelos de contexto.

Além destes artigos, outras publicações relacionadas ao trabalho desenvolvido aconteceram em regime de co-autoria, a seguir uma relação dos artigos com os dados da publicação.

- A Distributed Event-Driven Architectural Model based on Situational Awareness applied on Internet of Things. **Information and Software Technology, v. 111, p. 144-158, 2019.**

- An IoT Proposal for the Irrigation Management. **2nd IFIP International Internet of Things (IOT), 2019.**

- I^2VSM Approach: Self-Monitoring of Patients Exploring Situational Awareness in IoT. **2nd IFIP International Internet of Things (IOT), 2019**

- A Hierarchical Architectural Model for Network Security Exploring Situational Awareness. **The 34th ACM/SIGAPP Symposium on Applied Computing - Networking, 2019.**

- Revisão Sistemática sobre Segurança Adaptativa Ciente de Contexto para a Internet das Coisas. In: **Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos - SBRC 2019.**

- EXEHDA-SF: a IoT Based Proposal for the Irrigation Management. **Simpósio Brasileiro de Telecomunicações e Processamento de Sinais - SBrT2019.**

- Abordagem I^3VSM: Acompanhamento Autônomo de Pacientes Explorando Ciência de Situação na IoT. **Simpósio Brasileiro de Telecomunicações e Processamento de Sinais - SBrT2019.**

- Análise de Desempenho de Ferramentas para Persistência de Dados Ontológicos em Triplas: Experimentos e Resultados. In: **Seminário de Pesquisa em Ontologias no Brasil – ONTOBRAS 2019.**

- SPODi: Simulador do Perfil Operacional de Dispositivos Intravenosos para Auxilio à Tomada de Decisões Médicas. In: **Simpósio Brasileiro de Computação Aplicada à Saúde - SBCAS 2019.**

- I2WAC: Uma Proposta para Gerenciamento de Irrigação Explorando Ciência de Situação na IoT. In: **Congresso da Sociedade Brasileira de Computação, Seminário Integrado de Software e Hardware - SEMISH 2019.**

- EXEHDA-SO: Uma Abordagem Ontológica para Ciência de Situação Aplicada a Segurança da Informação. In: **Congresso da Sociedade Brasileira de Computação, Simpósio Brasileiro de Computação Ubíqua e Pervasiva, SBCUP 2018.**

- Uma Contribuição à Ciência de Contexto no Middleware EXEHDA Explorando Fog Computing. **In: Congresso da Sociedade Brasileira de Computação, Seminário Integrado de Software e Hardware, SEMISH 2018.**

- Hybrid approach to provide situational awareness for information security in computational environments. In: **Conferência Latino-americana de Informática, SLMDI - Simpósio Latino-Americano de Gerenciamento de Dados e Informação, 2018.**

- An Ontological Approach to Situational Awareness Applied to Information Security. In: **Conferência Latino-americana de Informática, SLIHS - Simpósio latino-americano em Infraestrutura, Hardware e Software, 2018.**

- Applying the Internet of Things in Precision Viticulture: An Approach Exploring the EXEHDA Middleware. In: **Conferência Latino-americana de Informática, SLIHS - Simpósio latino-americano em Infraestrutura, Hardware e Software, 2018.**

- Avaliação do Desempenho de Ferramentas para Persistência de Dados em Formato de triplas. In: **3ª Semana Integrada UFPel - XXVI Congresso de Iniciação Científica, 2017.**

- Análise de desempenho de sistemas de gerenciamento de dados em triplas com base no benchmark WatDiv. In: **XVIII Simpósio em Sistemas Computacionais de Alto Desempenho - Workshop de Iniciação Científica em Arquitetura de Computadores e Computação de Alto Desempenho, 2017.**

- Performance evaluation of SPARQL queries in triplestore management systems. In: **XXIII Simpósio Brasileiro de Sistemas Multimídia e Web, 2017, Gramado/RS. WebMedia'2017: Workshops e Pôsteres, 2017.**

- Análise do desempenho de consultas SPARQL em sistemas gerenciadores de triplas. In: **XVII Escola Regional de Alto Desempenho do Estado do Rio Grande do Sul - ERAD/RS, 2017.**

- Uma Arquitetura Hierárquica Multinível para Ciência de Situação em Segurança da Informação. In: **Congresso da Sociedade Brasileira de Computação, SBCUP - Simpósio Brasileiro de Computação Ubíqua e Pervasiva, 2016.**

- Uma Abordagem Ontológica para Ciência de Situação no Domínio de Segurança da Informação. In: **Encontro de Pós-Graduação da Universidade Federal de Pelotas, XVIII ENPOS, 2016.**

REFERÊNCIAS

AAMODT, A.; PLAZA, E. Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. **AI Commun.**, Amsterdam, The Netherlands, The Netherlands, v.7, n.1, p.39–59, Mar. 1994.

AGOSTINI, A.; BETTINI, C.; RIBONI, D. Hybrid Reasoning in the CARE Middleware for Context Awareness. **International Journal of Web Engineering and Technology**, Inderscience Publishers, Geneva, SWITZERLAND, v.5, n.1, p.3–23, May 2009.

AL-YASEEN, W. L.; OTHMAN, Z. A.; NAZRI, M. Z. A. Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. **Expert Systems with Applications**, Elsevier, v.67, p.296 – 303, 2017.

ALEGRE, U.; AUGUSTO, J. C.; CLARK, T. Engineering Context-aware Systems and Applications. **J. Syst. Softw.**, New York, NY, USA, v.117, n.C, p.55–83, July 2016.

BADII, A.; CROUCH, M.; LALLAH, C. A Context-Awareness Framework for Intelligent Networked Embedded Systems. **Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services**, Washington, DC, USA, p.105–110, 2010.

BAZ, I.; YORUK, E.; CETIN, M. Context-aware hybrid classification system for fine-grained retail product recognition. **2016 IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop, IVMSP 2016**, France, 2016.

BELLAVISTA, P.; CORRADI, A.; FANELLI, M.; FOSCHINI, L. A survey of context data distribution for mobile ubiquitous systems. **ACM Comput. Surv.**, New York, NY, USA, v.44, n.4, p.24:1–24:45, Sept. 2012.

BETTINI, C. et al. A Survey of Context Modelling and Reasoning Techniques. **Pervasive Mob. Comput.**, Amsterdam, The Netherlands, v.6, n.2, p.161–180, Apr. 2010.

BIBRI, S. E. Context Modeling, Representation, and Reasoning: An Ontological and Hybrid Approach. In: **The Human Face of Ambient Intelligence**. Paris: Atlantis Press, 2015. p.197–257. (Atlantis Ambient and Pervasive Intelligence, v.9).

CABRERA, M. M.; EDYE, E. O. Integration of Rule Based Expert Systems and Case Based Reasoning in an Acute Bacterial Meningitis Clinical Decision Support System. **CoRR**, USA, v.abs/1003.1493, 2010.

CAN, O.; SEZER, E.; BURSA, O.; UNALIR, M. O. Comparing Relational and Ontological Triple Stores in Healthcare Domain. **Entropy**, Switzerland, v.19, n.1, p.30, 2017.

COMMISSION, U. S. F. T. **The Internet of Things**: Privacy and Security in a Connected World. USA: DIANE Publishing Company, 2015. (Federal Trade Commission staff reports).

COOK, D. Learning Setting-Generalized Activity Models for Smart Spaces. **IEEE Intelligent Systems**, USA, v.27, n.1, p.32–38, Jan 2012.

COSTA, C. A. **Continuum**: A Context-aware Service-based Software Infrastructure for Ubiquitous Computing. 2008. Tese (Doutorado em Ciência da Computação) — UFRGS, Porto Alegre-RS.

COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T.; BLAIR, G. **Sistemas Distribuídos**: Conceitos e Projeto. Porto Alegre – RS: Bookman Editora, 2013.

DEY, A. K. Understanding and Using Context. **Personal Ubiquitous Comput.**, London, UK, v.5, n.1, p.4–7, Jan. 2001.

DIAS, F. O. **Um Modelo Proativo de Antecipação de Ações de Times de Resposta Rápida Baseado em Análise Preditiva**. 2017. Dissertação de Mestrado em Computação Aplicada — Unisinos, São Leopoldo-RS.

DIKICI, A.; TURETKEN, O.; DEMIRORS, O. Factors influencing the understandability of process models: A systematic literature review. **Information and Software Technology**, ELSEVIER, v.93, p.112 – 129, 2018.

ELEKAR, K.; WAGHMARE, M.; PRIYADARSHI, A. Use of rule base data mining algorithm for intrusion detection. **Pervasive Computing (ICPC), International Conference on**, Pune, India, p.1–5, Jan 2015.

FILHO, M. A. P. M. **SQL X NOSQL**: Análise de Desempenho do Uso do MongoDB em Relaçãoo ao Uso do PostgreSQL. 2015. Graduação em Ciência da Computação — Universidade Federal de Pernambuco.

GILMAN, E. **EXPLORING THE USE OF RULEBASED REASONING IN UBIQUITOUS COMPUTING APPLICATIONS**. 2015. Academic dissertation of the Doctoral Training Committee of Technology and Natural Sciences — UNIVERSITY OF OULU, OULU-Finland.

GOLDBERGER, A. L. et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. **Circulation 101(23):e215-e220, Circulation Electronic Pages**, USA, 2000.

GOMES, P. et al. A semantic-based discovery service for the Internet of Things. **Journal of Internet Services and Applications**, Springer London, v.10, n.1, p.10, 2019.

GRUBER, T. R. A translation approach to portable ontology specifications. **Knowledge Acquisition**, London, UK, UK, v.5, n.2, p.199–220, June 1993.

GUIZZARDI, G. On Ontology, Ontologies, Conceptualizations, Modeling Languages, and (Meta)Models. **Conference on Databases and Information Systems IV: Selected Papers from the Seventh International Baltic Conference DB&IS'2006**, Amsterdam, The Netherlands, p.18–39, 2007.

HE, L. An Improved Intrusion Detection based on Neural Network and Fuzzy Algorithm. **Journal of Networks**, Finland, v.9, n.5, p.1274–1280, 2014.

HEEAGER, L. T.; NIELSEN, P. A. A conceptual model of agile software development in a safety-critical context: A systematic literature review. **Information and Software Technology**, ELSEVIER, 2018.

HENRICKSEN, K.; INDULSKA, J. Developing Context-aware Pervasive Computing Applications: Models and Approach. **Pervasive Mob. Comput.**, Amsterdam, The Netherlands, v.2, n.1, p.37–64, Feb. 2006.

HOSSEINZADEH, S. et al. Diversification and obfuscation techniques for software security: A systematic literature review. **Information and Software Technology**, ELSEVIER, 2018.

JAPKOWICZ, N.; SHAH, M. **Evaluating Learning Algorithms**: A Classification Perspective. New York, NY, USA: Cambridge University Press, 2011.

KÄMPPÄ, J. **Recommendation systems in the context of tourism**. 2016. Degree Programme in Business Information Technology — Haaga-Helia University of Applied Sciences, Helsinki-Finland.

KHAN, N.; ALEGRE, U.; KRAMER, D.; AUGUSTO, J. C. Is 'Context-Aware Reasoning = Case-Based Reasoning'? **Modeling and Using Context**, Cham, p.418–431, 2017.

KHATTAK, A. M. et al. Context Representation and Fusion: Advancements and Opportunities. **Sensors**, USA, v.14, n.6, p.9628–9668, 2014.

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. Durham: Keele University and Durham University, 2007. Joint Report.

KNAPPMEYER, M. et al. Survey of Context Provisioning Middleware. **Communications Surveys Tutorials, IEEE**, USA, v.15, n.3, p.1492–1519, Third 2013.

KOÇ, H.; HENNIG, E.; JASTRAM, S.; STARKE, C. State of the Art in Context Modelling – A Systematic Literature Review. **Advanced Information Systems Engineering Workshops: CAiSE 2014 International Workshops, Thessaloniki, Greece, June 16-20, 2014. Proceedings**, Switzerland, p.53–64, 2014.

KOTENKO, I.; POLUBELOVA, O.; CHECHULIN, A.; SAENKO, I. Design and Implementation of a Hybrid Ontological-Relational Data Repository for SIEM Systems. **Future Internet - Special Issue Security of Systems and Software Resiliency**, USA, 2013.

LEE, W.; STOLFO, S.; MOK, K. A data mining framework for building intrusion detection models. **Security and Privacy, 1999. Proceedings of the 1999 IEEE Symposium on**, Oakland, USA, p.120–132, 1999.

LI, X.; ECKERT, M.; MARTINEZ, J.-F.; RUBIO, G. Context Aware Middleware Architectures: Survey and Challenges. **Sensors**, USA, v.15, n.8, p.20570–20607, 2015.

LI, X.; MARTÍNEZ, J.-F.; RUBIO, G. Towards a Hybrid Approach to Context Reasoning for Underwater Robots. **Applied Sciences**, USA, v.7, n.2, 2017.

LOPES, J. et al. A Middleware Architecture for Dynamic Adaptation in Ubiquitous Computing. **Journal of Universal Computer Science**, J.UCS, v.20, n.9, p.1327–1351, sep 2014.

LOPES, J. L. et al. A distributed architecture for dynamic contexts composition in Ubicomp. **38th Latin America Conference on Informatics - CLEI 2012 - Conference Proceedings**, Colombia, 2012.

MACHADO, A. **Associação do Contexto de Interesse do Usuário às Atividades Clínicas na Arquitetura Clinicspace**. 2010. Dissertação de Mestrado em Ciência da Computação — Programa de Pós-Graduação em Informática/UFSM, Santa Maria-RS.

MACHADO, G. M. et al. A systematic mapping on adaptive recommender approaches for ubiquitous environments. **Computing**, Austria, Aug 2017.

MACHADO, R. et al. Reconhecimento de Atividades em Ambientes de Vivência Assistida: uma Proposta Explorando um Modelo Ontológico. **Seminário de Pesquisa em Ontologias no Brasil – ONTOBRAS 2019**, Porto Alegre, Brasil, 2019.

OCEGUEDA-MIRAMONTES, V.; SANCHEZ, M. A.; AGUILAR, L. Towards Intelligent Systems for Ubiquitous Computing: Tacit Knowledge-Inspired Ubicomp. **Applied Decision-Making: Applications in Computer Sciences and Engineering**, Cham, p.65–94, 2019.

OTEBOLAKU, A. M. **Context-Aware Personalization for Mobile Multimedia**. 2015. Thesis of Doctor of Philosophy in Telecommunications — Faculty of Engineering University of Porto, Porto-Portugal.

PATIDAR, A.; SUMAN, U. A survey on software architecture evaluation methods. **2nd International Conference on Computing for Sustainable Global Development (INDIACom)**, India, p.967–972, March 2015.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, USA, v.12, p.2825–2830, Oct 2010.

PERERA, C.; ZASLAVSKY, A.; CHRISTEN, P.; GEORGAKOPOULOS, D. Context Aware Computing for The Internet of Things: A Survey. **Communications Surveys Tutorials, IEEE**, USA, v.16, n.1, p.414–454, First 2014.

PETERSEN, K.; VAKKALANKA, S.; KUZNIARZ, L. Guidelines for conducting systematic mapping studies in software engineering: An update. **Information and Software Technology**, Butterworth-Heinemann Newton, MA, USA, v.64, n.Supplement C, p.1 – 18, 2015.

PITTOLI, F. **ChronicPrediction**: Um modelo Para Prognóstico Ubíquo de Fatores de Risco de Doenças Crônicas Não Transmissíveis. 2015. Dissertação de Mestrado Acadêmico em Computação Aplicada — Programa Interdisciplinar de Pós-Graduação em Computação Aplicada/UNISINOS, São Leopoldo-RS.

PLA, A.; COLL, J.; MORDVANIUK, N.; LÓPEZ, B. Context-Aware Case-Based Reasoning. **Mining Intelligence and Knowledge Exploration**, Cham, p.229–238, 2014.

RANGANATHAN, A.; AL-MUHTADI, J.; CAMPBELL, R. H. Reasoning about uncertain contexts in pervasive computing environments. **IEEE Pervasive Computing**, USA, v.3, n.2, p.62–70, April 2004.

RAZA, S. M. A. **COMPARISON OF GEOSPATIAL SUPPORT IN RDF STORES**: EVALUATION FOR ICOS CARBON PORTAL METADATA. 2019. Master in Geographical Information Sciences — Lund University, Sweden.

RAZZAQ, M. A.; AMIN, M. B.; LEE, S. An ontology-based hybrid approach for accurate context reasoning. **19th Asia-Pacific Network Operations and Management Symposium (APNOMS)**, South Korea, p.403–406, Sept 2017.

RIBONI, D.; BETTINI, C. COSAR: Hybrid reasoning for context-Aware activity recognition. **Personal and Ubiquitous Computing**, London, v.15, n.3, p.271–289, 2011.

ROSA, D. et al. Uma abordagem híbrida para armazenamento de dados de contexto no EXEHDA. **Escola Regional de Redes de Computadores**, Passo Fundo - RS, 2015.

ROSA, F. et al. Análise de Desempenho de Ferramentas para Persistência de Dados Ontológicos em Triplas: Experimentos e Resultados. **Seminário de Pesquisa em Ontologias no Brasil – ONTOBRAS 2019**, Porto Alegre, Brasil, 2019.

SAHU, C. G.; ADANE, D. S. A Survey on Context-Aware Middleware. **International Journal of Advanced Research in Computer and Communication Engineering**, USA, v.4, n.5, p.650–656, May 2015.

SÁNCHEZ GUINEA, A.; NAIN, G.; LE TRAON, Y. A Systematic Review on the Engineering of Software for Ubiquitous Systems. **J. Syst. Softw.**, New York, NY, USA, v.118, n.C, p.251–276, Aug. 2016.

SAP. STANDARDIZED Technical Architecture Modeling. Disponível em: <http://www.fmc-modeling.org/download/fmc-and-tam/SAP-TAM_Standard.pdf>, acesso em dezembro 2018.

SARWAR, B.; KARYPIS, G.; KONSTAN, J.; RIEDL, J. Item-based Collaborative Filtering Recommendation Algorithms. **Proceedings of the 10th International Conference on World Wide Web**, New York, NY, USA, p.285–295, 2001.

SCHERER, A. P. Z.; JACOBSEN, D. G.; SANTOS, M. L. dos. **PostgreSQL**: instalando e conhecendo seus recursos. Porto Alegre, RS, Brasil: Faculdade Dom Bosco de Porto Alegre, 2008.

SEZER, O. B.; DOGDU, E.; OZBAYOGLU, A. M. Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey. **IEEE Internet of Things Journal**, IEEE, v.5, n.1, p.1–27, Feb 2018.

SHANMUGAPRIYA, P.; SURESH, R. M. Software Architecture Evaluation Methods - A Survey. **International Journal of Computer Applications**, USA, v.49, n.16, p.19–26, July 2012.

SHARP, J. et al. **Data Access for Highly-Scalable Solutions**: Using SQL, NoSQL, and Polyglot Persistence. 1st.ed. USA: Microsoft patterns & practices, 2013.

SIRIN, E. et al. Pellet: A Practical OWL-DL Reasoner. **Web Semant.**, Amsterdam, The Netherlands, The Netherlands, v.5, n.2, p.51–53, June 2007.

STRANG, T.; LINNHOFF-POPIEN, C. A Context Modeling Survey. **In: Workshop on Advanced Context Modelling, Reasoning and Management**, Nottingham/England, 2004.

STROBBE, M. et al. Hybrid reasoning technique for improving context-aware applications. **Knowledge and Information Systems**, London, v.31, n.3, p.581–616, 2012.

SUBBARAJ, R.; VENKATARAMAN, N. REASONING IN CONTEXT AWARE COMPUTING - A REVIEW. **International Journal of Pharmacy and Technology**, USA, v.8, p.5021–5032, 11 2016.

TEMDEE, P.; PRASAD, R. **Context-Aware Communication and Computing**: Applications for Smart Environment. Switzerland: Springer International Publishing, 2018. n.1. (Springer Series in Wireless Technology).

VIEIRA, V.; SOUZA, D.; SALGADO, A. C.; TEDESCO, P. **Uso e Representação de Contexto em Sistemas Computacionais**. Brasil: UFSCAR, 2006. p.127–166.

WANG, X.; ONG, S.; NEE, A. A comprehensive survey of ubiquitous manufacturing research. **International Journal of Production Research**, UK, p.1–25, 2017.

WATSON, I. Case-based Reasoning is a Methodology Not a Technology. **Know.-Based Syst.**, Amsterdam, The Netherlands, The Netherlands, v.12, n.5-6, p.303–308, Oct. 1999.

WEISER, M. The Computer for the 21st Century. **Scientific American**, New York, NY, USA, v.265, n.3, p.66–75, January 1991.

XU, T.; DAVID, B.; CHALON, R.; ZHOU, Y. A context-aware middleware for ambient intelligence. **Proceedings of the Workshop on Posters and Demos Track, PDT'11 - 12th International Middleware Conference**, Lisbon, Portugal, 2011.

YAN, M. et al. Automated change-prone class prediction on unlabeled dataset using unsupervised method. **Information and Software Technology**, USA, v.92, p.1 – 16, 2017.

YUAN, B.; HERBERT, J. Context-aware hybrid reasoning framework for pervasive healthcare. **Personal and Ubiquitous Computing**, New York, NY, USA, v.18, n.4, p.865–881, 2014.

ZAVALA, E.; FRANCH, X.; MARCO, J. Adaptive Monitoring: A Systematic Mapping. **Information and Software Technology**, ELSEVIER, 2018.