

Autonomic Ranking of Resources in IoT Exploring Fuzzy Logic and Machine Learning

Renato Dilli and Amanda Argou and Renata Reiser and Adenauer Yamin

Universidade Federal de Pelotas - RS, Brazil

Email: {renato.dilli, aacardozo, reiser, adenauer}@inf.ufpel.edu.br

Resumo—Currently, billions of resources are connected to the Internet, many providing services. There is forecast to be a significant increase in the number of resources in the coming years. The adequate selection of resources that best meet the demands of users among a large number of options has been a relevant and current research challenge in the autonomic IoT management. This paper specifies and evaluates the pre-classification of new resources of the EXEHDA middleware based on the non-functional parameters of QoS. Fuzzy logic is used in the treatment of uncertainties in defining the importance weights of QoS attributes. The results obtained in the evaluation of the accuracy of the pre-classification through fuzzy logic and machine learning are presented.

Index Terms—IoT, resources ranking, MCDA, fuzzy logic, machine learning.

I. INTRODUÇÃO

O cenário atual contabiliza mais de seis bilhões de coisas conectadas à Internet, disponibilizando serviços aos clientes, e previsão de mais de 100 bilhões até 2025 [1]. O número de serviços será maior à medida que as coisas físicas poderão fornecer uma ou mais funcionalidades.

Um desafio a ser vencido após a identificação e localização dos recursos, é classificar os serviços para selecionar o que melhor atende à solicitação do usuário. Os processos de classificação se concentram nas preferências do usuário que, frequentemente estabelecem uma ordem baseada na Qualidade de Serviço (QoS) [2] ou propriedades não-funcionais (NFPs - *Non-Functional Properties*). Usando essas preferências, o conjunto de serviços descobertos pode ser classificado de modo que o melhor serviço possa ser escolhido [3].

As propostas atuais para descoberta e acesso aos serviços para a IoT têm como importante frente de pesquisa em aberto a grande quantidade de recursos, com elevada heterogeneidade, em um ambiente de composição dinâmica, características essas típicas da infraestrutura provida pela Internet [4].

Para atender o perfil operacional da infraestrutura da IoT, entretanto, o processo de descoberta de recursos deve considerar tanto os requisitos funcionais como os não-funcionais. Os funcionais descrevem explicitamente as funcionalidades dos recursos, ou seja, o que ele pode oferecer e os não-funcionais definem informações adicionais sobre os serviços, tais como: desempenho e segurança [5].

Os recursos, geralmente disponibilizados por meio de serviços, possuem atributos de QoS com valores apresentando unidades de medidas heterogêneas e faixas amplas nos valores mínimos e máximos. Para selecionar o recurso mais oportuno

para a requisição do cliente é essencial a correta especificação destes valores.

No ambiente computacional, os responsáveis por definir estes valores devem ter conhecimento de todas as características dos atributos de qualidade, e desta forma avaliar os valores para qualificar os recursos. Considerando que a definição destes valores pode ser feita por especialistas com percepções diferentes, surge a incerteza quando da definição dos valores a serem adotados como atributos de QoS. Neste contexto, este artigo explora a lógica fuzzy para o tratamento da incerteza na definição do grau de relevância dos atributos de qualidade dos recursos disponibilizados na Internet das Coisas. Ainda, este modelo considera as preferências do cliente e os atributos de qualidade dos serviços, agregando tecnologias de Web Semântica na especificação e consulta de recursos.

Como principais contribuições, este trabalho apresenta duas propostas para pré-classificação de recursos para o *middleware* EXEHDA, chamada EXEHDA-RD (*Resource Discovery*) brevemente descritas logo a seguir.

- A primeira, empregando **aprendizagem de máquina** (AM) no reconhecimento de padrões para a pré-classificação de recursos. Nesta proposta os recursos são inicialmente classificados por algoritmo de Análise de Decisão com Múltiplos Critérios (MCDA). A cada classificação, um novo treinamento do algoritmo de aprendizagem de máquina é realizado para pré-classificar novos recursos ao entrarem na infraestrutura computacional.
- E, a segunda proposta explorando a **lógica fuzzy** (LF) com o propósito de avaliar o impacto na precisão da classificação de recursos. Neste caso, o objetivo da avaliação é reduzir o conjunto de recursos através de pré-classificação com lógica fuzzy para posterior classificação com o algoritmo MCDA.

O diferencial destes processos de pré-classificação que considera a integração de conceitos destas áreas de pesquisa - AM e LF, é a significativa redução da necessidade de processar todos os cálculos dos algoritmos MCDA a cada requisição do cliente. Além destas contribuições, este artigo apresenta uma avaliação da acurácia no emprego destas duas abordagens.

Este trabalho está organizado da seguinte forma: na Seção II e III são apresentados conceitos sobre aprendizagem de máquina na seleção de recursos e lógica fuzzy, respectivamente; A Seção IV destaca o processo de classificação de recursos utilizado; A Seção V mostra a arquitetura do EXEHDA-RD, enquanto a Seção VI apresenta o classificador

de recursos do EXEHDA-RD e o resultado das avaliações; Na Seção VII são discutidos os Trabalhos Relacionados e ao final são feitas as Considerações Finais.

II. APRENDIZAGEM DE MÁQUINA NA SELEÇÃO DE RECURSOS

A aprendizagem de máquina visa compreender a lógica dos programas para melhorar o desempenho das máquinas em determinadas tarefas, através da experiência anterior [6]. Sendo assim, os algoritmos de aprendizagem automática têm-se revelado extremamente úteis em diversos domínios, desde logo têm sido especialmente importantes na resolução de problemas de mineração de dados, em que universos de dados extensos poderão conter implicitamente informações consideradas relevantes e que poderão ser descobertas automaticamente. Estes algoritmos são igualmente essenciais em domínios de compreensão difícil, em que os próprios humanos não possuem capacidade para o desenvolvimento de algoritmos eficazes.

Algoritmos de aprendizagem de máquina supervisionada podem ser utilizados para a geração automática de regras com o objetivo de classificar os serviços, considerando os requisitos não-funcionais.

Foram avaliados os algoritmos de árvore de decisão J48, LMT, RandomForest, SimpleCart, DecisionStump, RandomTree, HoeffdingTree e REPTree na ferramenta WEKA [7]. O objetivo da avaliação é verificar a acurácia dos algoritmos na classificação de serviços considerando os valores de atributos de QoS.

A classificação dos serviços foi realizada através de aprendizagem supervisionada para classificar o atributo “Service Classification” definido no *dataset* QWS [8].

Conforme os resultados apresentados na Tabela I, nota-se que o classificador LMT obteve uma acurácia de 85.99%. O LMT (*Logistic Model Trees*) constrói árvores do modelo logístico e usa validação cruzada para determinar quantas iterações executar apenas uma vez, e emprega o mesmo número em toda a árvore em vez de validação cruzada em cada nó. O mecanismo de poda utilizado produz uma estrutura de árvore compacta.

Tabela I
COMPARATIVO DOS CLASSIFICADORES

Classificador	Acertos	Erros	Acurácia
LMT	313	51	85.99%
RandomForest	294	70	80.77%
J48	258	106	70.88%
RandomTree	248	116	68.13%
SimpleCart	248	116	68.13%
REPTree	239	125	65.66%
HoeffdingTree	198	166	54.40%
DecisionStump	160	204	43.95%

Os testes dos classificadores foram realizados sem alterar os padrões da ferramenta WEKA, com exceção do J48. A acurácia do J48 com as configurações padrão foi de 69.50%, após ativar a opção “ReduceErrorPuning” a acurácia subiu para 70.88%.

Pode-se observar na Matriz de Confusão do algoritmo LMT, conforme Tabela II, que os erros ocorridos na predição foram

Tabela II
LMT E J48

a	b	c	d	classificado
LMT				
37	4	0	0	a = 1
3	86	11	0	b = 2
0	11	100	9	c = 3
0	0	13	90	d = 4
J48				
31	9	1	0	a = 1
15	66	18	1	b = 2
1	24	79	16	c = 3
0	2	19	82	d = 4

pequenos e aceitáveis na perspectiva da qualidade do serviço na escala de 1 a 4, considerando o valor 1 mais bem qualificado. Na primeira linha, onde todas as instâncias deveriam ser classificadas com valor 1, 4 foram classificadas com valor 2. O erro é pequeno, pois nenhuma instância classificada como 1 no *dataset* original foi classificada em 3 ou 4. Na segunda linha, todas as instâncias deveriam ser classificadas com valor 2, mas 3 foram classificadas com valor 1 e 11 classificadas com valor 3. Na terceira linha as instâncias deveriam ser classificadas com valor 3, mas 11 foram classificadas com valor 2 e 9 classificadas com valor 4. Na última linha as instâncias deveriam ser classificadas com valor 4, mas 13 instâncias foram classificadas com valor 3.

A Matriz de Confusão do algoritmo J48, apresenta uma maior imprecisão, se comparada à matriz de confusão do algoritmo LMT. Na primeira linha, onde todas as instâncias deveriam ser classificadas com valor 1, 9 foram classificadas com valor 2 e 1 classificada com valor 3. Na segunda linha, todas as instâncias deveriam ser classificadas com valor 2, mas 15 foram classificadas com valor 1, 18 classificadas com valor 3 e 1 classificada com valor 4. Na terceira linha as instâncias deveriam ser classificadas com valor 3, mas 1 foi classificada com valor 1, 24 com valor 2, 79 com valor 3 e 16 com valor 4. Na última linha as instâncias deveriam ser classificadas com valor 4, mas 2 foram classificadas com valor 2, 19 com valor 3 e 82 com valor 4.

Através da análise dos classificadores do tipo árvores de decisão, foi possível avaliar a acurácia do classificador e obter o conjunto de regras para o ranqueamento de novos recursos que satisfaçam as requisições dos clientes.

III. LÓGICA FUZZY

A teoria do conjunto fuzzy, que tem sido amplamente aplicada para modelar as ambiguidades do pensamento humano, também resolve efetivamente as incertezas nas informações disponíveis para tomada de decisão baseada em múltiplos critérios. A adequação de substituições versus critérios e o peso de significância de critérios são avaliados em termos de valores linguísticos representados por números difusos. No conjunto fuzzy, as variáveis linguísticas são usadas para descrever termos difusos que mapeiam variáveis linguísticas para variáveis numéricas. Os valores de verdade da lógica booleana são substituídos com intervalos de unidade no processo de tomada de decisão [9].

A principal vantagem de se utilizar lógica fuzzy deve-se a sua capacidade de lidar com incertezas, raciocínio aproximado,

termos vagos e ambíguos, o que não é possível de se fazer com as lógicas clássicas. O raciocínio humano envolve todos esses elementos tratados pela lógica fuzzy, por isso ela é de suma importância no contexto da computação, especialmente na inteligência artificial, que procura representar o raciocínio e o conhecimento humano da forma mais realística possível. Nos últimos anos, o potencial de manuseio de incertezas e de controle de sistemas complexos tem-se tornado possíveis com o uso da lógica fuzzy. Estes estão sendo combinados com métodos de aprendizagem de máquina, que por sua vez, possuem características de adaptação e aprendizagem. A palavra certa para isto é simbiose, que vem gerando novas classes de sistemas e de controladores neurodifusos, combinando desta forma os potenciais e as características individuais em sistemas adaptativos e inteligentes[10].

Assim, matematicamente, um conjunto é definido como uma coleção finita, infinita, ou contável infinita de elementos. Em cada caso, cada elemento é um membro do conjunto ou não. No entanto, em sistemas fuzzy, o elemento pode estar parcialmente dentro ou fora do conjunto. Assim, a resposta à pergunta: “x é membro de um conjunto A” não tem uma resposta definitiva, como verdadeiro ou falso.

A. Conjuntos Fuzzy

Um conjunto fuzzy A definido no universo de discurso $U \neq \emptyset$ é caracterizado por uma função de pertinência $\mu_A : U \rightarrow [0, 1]$, caracterizado pela expressão:

$$A = \{(x, \mu_A(x)) \mid x \in U \wedge \mu_A(x) \in [0, 1]\}$$

Para cada $x \in X$, $\mu_A(x)$ representa o grau de pertinência de x em A .

$$x \in (A, \mu) \iff x \in A \wedge \mu(x) \neq 0$$

Desta forma, pela função de pertinência cada elemento $x \in U$ tem um grau de pertinência em cada conjunto A , expressando o quanto é possível para o elemento x pertencer ao conjunto A . Assim, quando um elemento com grau de pertinência 0, significa que não está incluso no conjunto fuzzy. Enquanto que um elemento com grau 1 está totalmente incluído neste.

B. Fuzzificação

Considerando as especificidades da área de aplicação do presente trabalho foram adotadas funções de pertinência triangulares. Um número fuzzy triangular A pode ser definido por uma tripla (a, b, c) com a função de pertinência dada pela Eq.(1) dada a seguir:

$$\mu_A(x) = \begin{cases} 0, & \text{se } x \leq a; \\ \frac{x-a}{b-a}, & \text{se } a \leq x \leq b; \\ \frac{c-x}{c-b}, & \text{se } b \leq x \leq c; \\ 0, & \text{se } x \geq c. \end{cases} \quad (1)$$

C. Defuzzificação

A defuzzificação é o processo que produz um resultado quantificável na lógica fuzzy. Isso significa que a

defuzzificação transforma um número fuzzy em um único número. Existem muitos métodos diferentes de defuzzificação, sendo para este trabalho considerado o método da média ponderada dos máximos, conforme a Eq. (2) dada a seguir

$$Z_0 = \frac{\mu(x)_i w_i}{\mu(x)_i}, \quad (2)$$

De acordo com a Eq. (2), onde Z_0 é a saída defuzzificada, $\mu(x)_i$ é o grau de pertinência e w_i é o valor de peso de saída fuzzy.

IV. MCDA NA CLASSIFICAÇÃO DE RECURSOS

O processo de descoberta de recursos engloba a classificação e seleção dos recursos mais adequados à requisição do cliente. Nesta seção é apresentado o algoritmo MCDA para classificação e seleção de recursos utilizado neste trabalho.

Análise de Decisão de Múltiplos Critérios (MCDA) refere-se à tomada de decisões na presença de critérios múltiplos, geralmente conflitantes. Os algoritmos MCDA visam auxiliar no julgamento da tomada de decisão utilizando um conjunto de objetivos e critérios, estimando seus pesos de importância relativa e estabelecendo a contribuição de cada opção em relação a cada critério de desempenho. MCDA não é apenas um conjunto de teorias, metodologias e técnicas, mas também uma perspectiva específica para lidar com problemas de tomada de decisão [11].

O algoritmo *The Simple Additive Weighting* (SAW) usa uma pontuação de avaliação para classificar cada opção disponível. A pontuação é obtida através de valores de critérios normalizados multiplicados por um peso. As opções são classificadas em ordem decrescente de acordo com sua pontuação final, que é a soma das pontuações para cada critério [12].

O algoritmo MCDA utilizado neste artigo é uma adaptação dos algoritmos SAW e *Web Service Relevancy Function* (WsRF) [8], com a primeira etapa de normalização de dados proposta por [13].

Para a normalização da matriz são definidos dois vetores. O primeiro $N = \{n_1, n_2, \dots, n_m\}$. O valor de n_j pode ser 0 ou 1. Será 1 quando o aumento de $q_{i,j}$ beneficia a requisição do cliente e 0 quando o aumento de $q_{i,j}$ não beneficia a requisição do cliente. O segundo vetor $C = \{c_1, c_2, \dots, c_m\}$ contém constantes com o máximo valor normalizado para cada atributo.

As seguintes etapas devem ser realizadas para o cálculo da avaliação dos recursos através do algoritmo MCDA:

- 1) Normalizar a matriz $Q = (q_{ij})_{n \times m}$ de acordo com a Eq.(3) se o critério deve ser maximizado ou Eq.(4) se o critério deve ser minimizado. Nestas equações, $\frac{1}{n} \sum_{i=1}^n q_{i,j}$ é a média dos atributos de qualidade j na matriz Q .

$$v_{i,j} = \begin{cases} \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} & \text{if } \frac{1}{n} \sum_{i=1}^n q_{i,j} \neq 0 \\ & \text{and } \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} < c_j \\ & \text{and } n_j = 1 \\ c_j & \text{if } \frac{1}{n} \sum_{i=1}^n q_{i,j} = 0 \\ & \text{and } n_j = 1 \\ & \text{or } \frac{q_{i,j}}{\frac{1}{n} \sum_{i=1}^n q_{i,j}} \geq c_j \end{cases} \quad (3)$$

$$v_{i,j} = \begin{cases} \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} & \text{if } q_{i,j} \neq 0 \\ & \text{and } n_j = 0 \\ & \text{and } \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} < c_j \\ c_j & \text{if } q_{i,j} = 0 \\ & \text{and } n_j = 0 \\ & \text{or } \frac{\frac{1}{n} \sum_{i=1}^n q_{i,j}}{q_{i,j}} \geq c_j \end{cases} \quad (4)$$

- 2) Calcular o vetor pontuação de cada opção disponível. Cada pontuação pode ser calculada usando a Eq.(5) e o operador $\max(v_j)$ representando o maior valor de atributo normalizado na coluna j . Portanto, precisamos definir uma matriz que represente a contribuição de pesos para cada recurso, onde $w = \{w_1, w_2, w_3, \dots, w_j\}$. Cada peso nesta matriz representa o grau de importância ou fator de peso associado a uma propriedade QoS específica. Os valores de desses pesos variam de 0 a 1. A Eq.(6) soma todos os atributos de qualidade para o recurso R_i , onde N representa o número de atributos.

$$h_{i,j} = w_j \left[\frac{v_{i,j}}{\max(v_j)} \right]_{(5)} \quad R_i = \sum_{j=1}^N h_{i,j} \quad (6)$$

$$MCDA(A_i) = \left[\frac{100 * R_i}{\max(R)} \right] \quad (7)$$

- 3) O resultado final do algoritmo MCDA, Eq.(7), é reclassificado, como mostra a Figura. 1. As folhas da árvore são representadas por uma caixa contendo a classificação (1 a 4).

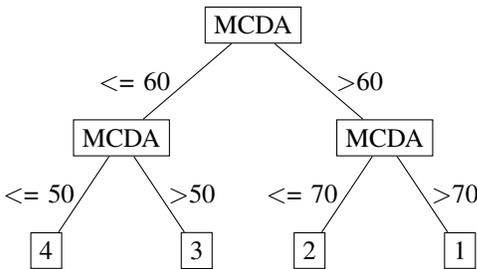


Figura 1. Reclassificação do algoritmo MCDA

V. EXEHDA-RD: ARQUITETURA E FUNCIONALIDADES

O EXEHDA [14] é o *middleware*, foco deste trabalho, responsável por prover a infraestrutura computacional básica para a IoT. Este ambiente computacional é constituído por células de execução, nas quais os dispositivos computacionais são distribuídos. Cada célula (Figura 2) é constituída dos seguintes componentes: (i) EXEHDABase, o elemento central da célula, sendo responsável por todos serviços básicos e constituindo referência para os demais elementos; (ii) o EXEHDAAnodo que corresponde aos dispositivos computacionais responsáveis

pela execução das aplicações; (iii) o EXEHDAAnodo móvel, um subcaso do anterior, que corresponde aos dispositivos tipicamente móveis que podem se deslocar entre as células do ambiente ubíquo, (iv) o EXEHDA Borda, responsável por fazer a interoperação entre os serviços do *middleware* e os diversos tipos de *gateways*; e (v) o EXEHDA Gateway, que consiste no elemento responsável por setorizar pontos de coleta e/ou atuação distribuídos, disponíveis no meio físico, realizando a interação destes com os outros componentes do *middleware*.

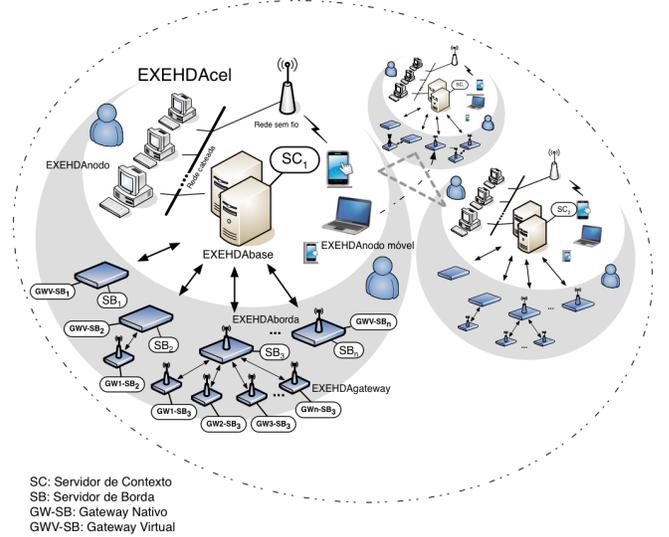


Figura 2. Organização celular do ambiente ubíquo gerenciado pelo EXEHDA

O EXEHDA-RD (*Resource Discovery*) é um novo serviço para descoberta de recursos proposto para o *middleware* EXEHDA. Sua arquitetura de software foi modelada considerando a dinamicidade em que os recursos entram e saem do ambiente, portanto, controla a presença de dispositivos, inclusive com baixo poder computacional, conectados à gateways. O aumento da expressividade na descrição e consulta por recursos foi provida através de tecnologias de Web Semântica com intuito de potencializar a descoberta de recursos no ambiente. A arquitetura de software do EXEHDA-RD possui três componentes distintos: (CD) Componente Diretório, (CR) Componente Recurso e (CC) Componente Cliente (Figura 3).

A. CC - Componente Cliente

O Componente Cliente (CC) é responsável pelo disparo da consulta por recursos, empregando critérios pertinentes às demandas do cliente. O CC é composto do módulo Gerador de Consultas.

B. CR - Componente Recurso

O Componente Recurso (CR) é responsável por notificar a presença do recurso. Isto é feito por troca de mensagens com o CD, dentro de um intervalo de tempo. Quando o recurso ingressa pela primeira vez em uma célula, o CR envia a descrição dos seus recursos para o CD da célula onde está localizado o recurso. O CD gera uma instância do recurso na

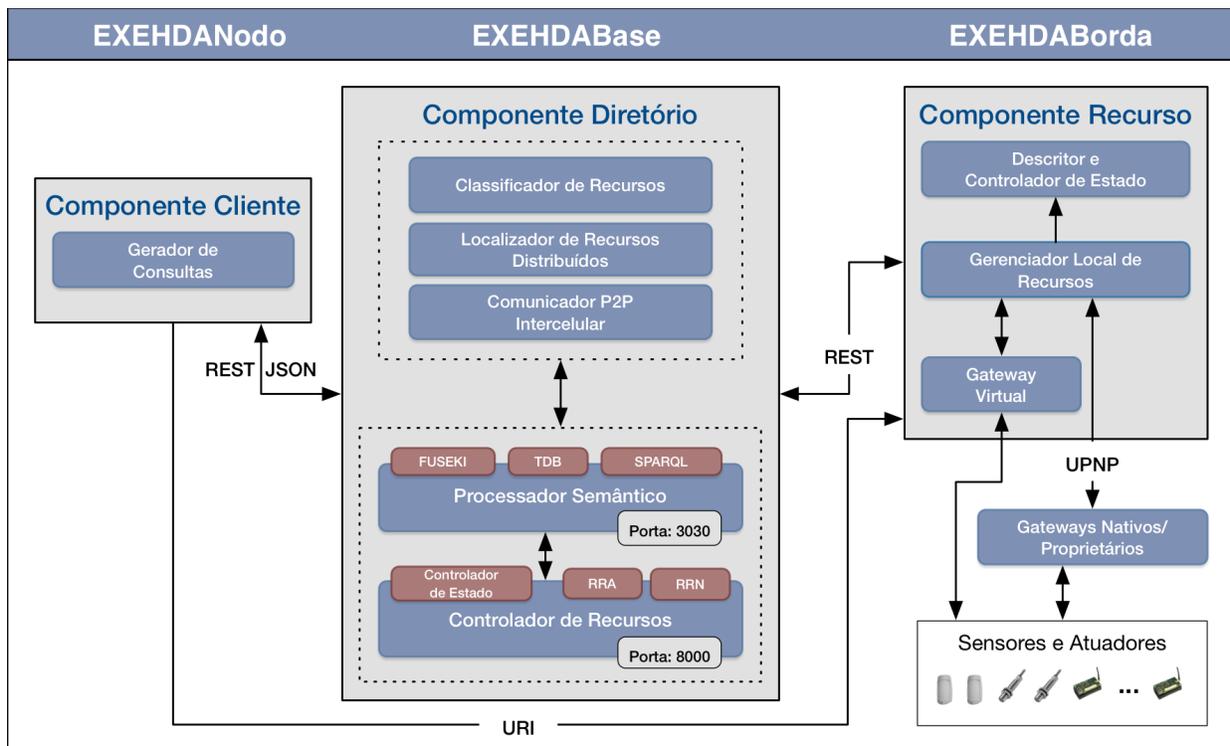


Figura 3. Arquitetura de software do EXEHDA-RD

ontologia local. O CR é composto pelos módulos Descritor e Controlador de Estado e Gerenciador Local de Recursos.

O módulo Descritor e Controlador de Estado é responsável por enviar mensagens ao CD com as características do recurso e anunciar sua presença no ambiente. O módulo Gerenciador Local de Recursos tem por objetivo realizar a descoberta de gateways através do protocolo UPnP.

O CR é ativado em nodos EXEHDABorda, cuja finalidade é anunciar a presença dos dispositivos confinados que estão conectados aos gateways da célula.

C. CD - Componente Diretório

O Componente Diretório (CD), instanciado no EXEHDA-Base é formado por cinco módulos:

- 1) Controlador de Recursos: o qual tem 5 objetivos; (i) realizar a manutenção de estado dos recursos. Os estados possíveis são: Ativo, Inativo e Negado; (ii) gerenciar o Repositório de Recursos Ativos (RRA); (iii) gerenciar o Repositório de Recursos Negados (RRN); (iv) receber as consultas dos Componentes Cliente e (v) receber as mensagens enviadas pelos Componentes Recurso. O *lease* é um intervalo de tempo que é gerenciado pelo Controlador de Recursos. O Componente Recurso precisa renovar o *lease* do recurso localizado no CD periodicamente, caso contrário será removido do Repositório de Recursos Ativos (RRA).
- 2) Processador Semântico: responsável pelo processamento das consultas em linguagem SPARQL. O processamento semântico segue o padrão arquitetural REST com a integração do Apache Jena Fuseki [15]. O componente

TDB realiza a persistência de dados e o motor de inferência foi configurado para utilizar o raciocinador genérico do Apache Jena. É possível definir regras adicionais para aumentar a expressividade nas consultas. A Figura 4 descreve uma requisição por recursos na Linguagem SPARQL. Nesta consulta são especificadas as preferências do cliente. Os recursos são classificados de 1 a 4 no Componente Diretório, sendo 1 melhor classificado e 4 o pior. A consulta é realizada em ordem ascendente de classificação e, após, em ordem de preferência do cliente. Na listagem o cliente priorizou *Response Time* (Ordem Ascendente), pois quanto menor o valor melhor e *Reliability* (Ordem Descendente), quanto maior o valor melhor desempenho terá o recurso. Na ontologia do Processador Semântico, cinco atributos de qualidade foram definidos como “Propriedades de Dados” para a classe “Recurso”: QoS_AV (Availability), QoS_LA (Latency), QoS_RE (Reliability), QoS_RT (Response Time) e QoS_TH (Throughput).

- 3) Comunicador P2P Intercelular: através deste módulo o mecanismo de descoberta processa a consulta de clientes em outras células do ambiente utilizando protocolo P2P.
- 4) Localizador de Recursos Distribuídos: responsável por receber e organizar as respostas das consultas realizadas em outras células, recebidas pelo módulo Comunicador P2P Intercelular.
- 5) Classificador de Recursos: tem por finalidade realizar a classificação dos recursos mais adequados à solicitação do cliente. Este módulo contém os seguintes submódulos:
 - Tratador Fuzzy. Tem por objetivo resolver as di-

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-
→ syntax-ns#>
PREFIX ont: <http://www.owl-ontologies.com/
→ Ontology1251223167.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
SELECT *
WHERE {
?resource rdf:type ont:Resource .
?resource ont:QoS_RT ?RT .
?resource ont:QoS_AV ?AV .
?resource ont:QoS_TH ?TH .
?resource ont:QoS_RE ?RE .
?resource ont:QoS_LA ?LA .
?resource ont:QoS_Classif ?Classif .
}
ORDER BY ASC(?Classif) ASC(?RT) DESC(?RE)
LIMIT 20

```

Figura 4. SPARQL Query Specification

vergências entre vários especialistas na atribuição dos pesos de importância de cada atributo de QoS.

- Classificador MCDA. Responsável por classificar todos os recursos descobertos através do algoritmo MCDA proposto, tendo como entrada os pesos definidos pelo Tratador Fuzzy.
- Pré-classificador. Os novos recursos recém descobertos são pré-classificados, visando reduzir o custo computacional gerado pelo algoritmo MCDA.

VI. EXEHDA-RD: CLASSIFICADOR DE RECURSOS

O módulo “Classificador de Recursos” é responsável pela classificação e seleção de recursos do *middleware* EXEHDA. Seus sub-módulos são apresentados neste artigo.

A. Tratador Fuzzy

No processo de classificação dos recursos que podem satisfazer a requisição do usuário, a avaliação dos atributos de QoS é uma etapa desafiadora. A definição do grau de importância dos atributos de QoS pelo usuário e/ou administradores da infraestrutura computacional é uma atividade que depende da experiência e conhecimento de cada indivíduo. A participação de várias pessoas neste processo pode apresentar divergências nas definições dos valores dos atributos de QoS. Este sub-módulo visa tratar estas divergências com o emprego de lógica fuzzy.

Para avaliar o emprego da lógica fuzzy na especificação dos atributos de QoS ideais, definidos pelos especialistas do ambiente computacional, foi utilizado o *dataset* QWS versão 2.0 disponibilizado por [8], com 2.505 recursos e 9 atributos de qualidade. A Tabela III descreve os 5 atributos utilizados.

A Figura 5 e a Tabela IV mostram os Termos Linguísticos e os pesos atribuídos.

A Tabela V simula a definição do grau de importância de cada atributo de qualidade usando os Termos de Linguísticos definidos por cinco especialistas. Essa avaliação visa definir

Tabela III
ATRIBUTOS SELECIONADOS DO DATASET QWS

Atributo	Descrição	Unid
RT - Response Time	Tempo de enviar uma requisição e receber sua resposta	ms
AV - Availability	Número de invocações corretas/total invocações	%
TH - Throughput	Total de invocações por um determinada período	%
RE - Reliability	Taxa de erro no total de mensagens	%
LA - Latency	Tempo que o servidor leva para processar uma requisição	ms

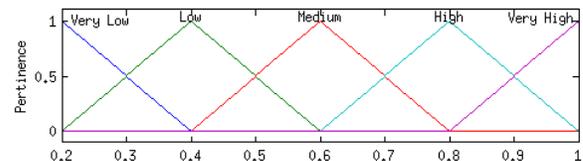


Figura 5. Representação do Conjunto Fuzzy

Tabela IV
GRAU DE IMPORTÂNCIA DOS ATRIBUTOS

Termo Linguístico	Valor
VH - Very High	(0.8, 1.0, 1.0)
H - High	(0.6, 0.8, 1.0)
M - Medium	(0.4, 0.6, 0.8)
L - Low	(0.2, 0.4, 0.6)
VL - Very Low	(0.0, 0.2, 0.4)

os pesos a serem atribuídos a cada atributo para fornecer os recursos mais adequados à solicitação do usuário.

Tabela V
AVALIAÇÃO DOS ESPECIALISTAS

Atrib	Esp1	Esp2	Esp3	Esp4
RT	M	H	L	H
AV	VH	VH	H	VH
TH	M	M	H	H
RE	H	H	VH	H
LA	VH	H	VL	H

Na Tabela VI é apresentada a conversão dos Termos Linguísticos, atribuídos pelos especialistas, em números fuzzy triangulares. A Tabela VII apresenta a média fuzzy entre os

Tabela VI
CÁLCULO DOS PESOS DE QoS

Atr	Esp1	Esp2	Esp3	Esp4
RT	(0.4, 0.6, 0.8)	(0.6, 0.8, 1.0)	(0.2, 0.4, 0.6)	(0.6, 0.8, 1.0)
AV	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.6, 0.8, 1.0)	(0.9, 1.0, 1.0)
TH	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0.6, 0.8, 1.0)	(0.6, 0.8, 1.0)
RE	(0.6, 0.8, 1.0)	(0.6, 0.8, 1.0)	(0.9, 1.0, 1.0)	(0.6, 0.8, 1.0)
LA	(0.9, 1.0, 1.0)	(0.6, 0.8, 1.0)	(0.0, 0.2, 0.4)	(0.6, 0.8, 1.0)

especialistas e a defuzzificação utilizando o método da média ponderada. Ao final os valores dos atributos são normalizados.

Os valores normalizados resultantes serão aplicados no processo de classificação e seleção de recursos do algoritmo MCDA.

Tabela VII
CÁLCULO DOS PESOS DE QoS

Atrib	Média Fuzzy	Defuzz	Norm
RT	(0.45, 0.65, 0.85)	0.65	0.70
AV	(0.83, 0.95, 1.00)	0.93	1.00
TH	(0.50, 0.70, 0.90)	0.70	0.75
RE	(0.68, 0.85, 1.00)	0.87	0.94
LA	(0.53, 0.70, 0.85)	0.70	0.75

B. Classificador MCDA

A aplicação do algoritmo MCDA é demonstrada na classificação de um dataset contendo 5 atributos de qualidade para 10 recursos. Cada linha representa um recurso e cada coluna um atributo de qualidade (Tabela VIII). Os valores dos atributos Resp.Time e Latency são melhores se forem baixos e os valores dos atributos Availability, Throughput e Reliability são melhores se forem altos.

Tabela VIII
EXEMPLO DO DATASET

Resp.Time	Availability	Throughput	Reliability	Latency
302.75	89	7.1	73	187.75
482	85	16	73	1
3321.4	89	1.4	73	2.6
126.17	98	12	67	22.77
107	87	1.9	73	58.33
107.57	80	1.7	67	18.21
255	98	1.3	67	40.8
136.71	76	2.8	60	11.57
102.62	91	15.3	67	0.93
200	40	13.5	67	41.66

A primeira etapa da classificação é a normalização dos dados. Para tanto, consideramos os vetores $N=\{0, 1, 1, 1, 0\}$ e $C=\{6, 2, 3, 2, 50\}$ e $w=\{0.70, 1.00, 0.75, 0.94, 0.75\}$. Os atributos Resp.Time e Latency qualificam o recurso com valores baixos, Eq.(4) e Availability, Throughput e Reliability qualificam com valores altos, Eq.(3). Todos os atributos são normalizados com valor máximo definido em "C", o resultado é apresentado na Tabela IX.

Tabela IX
ATRIBUTOS NORMALIZADOS

RT	AV	TH	RE	LA
1.70	1.07	0.97	1.06	0.21
1.07	1.02	2.19	1.06	38.56
0.15	1.07	0.19	1.06	14.83
4.07	1.18	1.64	0.98	1.69
4.80	1.04	0.26	1.06	0.66
4.78	0.96	0.23	0.98	2.12
2.02	1.18	0.18	0.98	0.95
3.76	0.91	0.38	0.87	3.33
5.01	1.09	2.10	0.98	41.46
2.57	0.48	1.85	0.98	0.93

A Tabela X apresenta os valores dos atributos após aplicar a Eq.(5), ou seja dividir o valor normalizado da Tabela IX pelo maior valor normalizado de cada coluna. Após é aplicada a Eq.(6), que irá somar todos os valores de atributos em cada linha. A seguir é aplicada a Eq.(7) que irá qualificar o recurso com um valor que vai de 0 a 100. Será dado o valor 100 para o melhor recurso do *dataset*. A classificação de 1 a 4 é atribuída através da regra ilustrada na Figura 1.

Tabela X
ATRIBUTOS CLASSIFICADOS

RT	AV	TH	RE	LA	MCDA	Classif
0.24	0.91	0.33	0.94	0.00	61.18	2
0.15	0.87	0.75	0.94	0.70	85.99	1
0.02	0.91	0.07	0.94	0.27	55.67	3
0.57	1.00	0.56	0.86	0.03	76.42	1
0.67	0.89	0.09	0.94	0.01	65.68	2
0.67	0.82	0.08	0.86	0.04	62.27	2
0.28	1.00	0.06	0.86	0.02	56.14	3
0.53	0.78	0.13	0.77	0.06	57.22	3
0.70	0.93	0.72	0.86	0.75	100	1
0.36	0.41	0.63	0.86	0.02	57.59	3

O algoritmo MCDA apresentado nesta seção foi utilizado para classificar os recursos e gerar a classificação de referência para os pré-classificadores propostos.

C. Pré-classificador

Na infraestrutura computacional provida pela IoT a entrada e saída de recursos é muito dinâmica e altamente escalável, desta forma, uma pré-classificação do recurso ao entrar no ambiente pode pré-selecionar os melhores recursos para serem ranqueados pelo algoritmo MCDA e disponibilizados ao cliente. Para o pré-classificador do EXEHDA-RD duas propostas para a pré-classificação de novos recursos foram avaliadas. A primeira utiliza o algoritmo LMT para a pré-classificação e a segunda lógica fuzzy.

1) *Pré-classificação de recursos explorando aprendizagem de máquina*: Com o emprego do algoritmo de aprendizagem de máquina LMT foram realizadas 4 avaliações da pré-classificação de recursos. Foi escolhido o algoritmo LMT por ter obtido a melhor acurácia dentre os algoritmos de árvore de decisão analisados. Em cada avaliação foram classificados 1000 recursos com valores de 1 a 4. Os recursos foram agrupados em lotes de 50, 100, 200 e 500 recursos.

Em cada avaliação, o grupo de recursos classificados através do algoritmo MCDA é retirado do conjunto de dados de treinamento e usado no conjunto de dados de teste do algoritmo de aprendizado de máquina.

As avaliações foram realizadas com os seguintes repositórios iniciais para treinamento do algoritmo LMT: (a) - repositório inicial com 100 recursos; (b) - repositório inicial com 200 recursos e (c) - repositório inicial com 300 recursos.

A Tabela XI apresenta as classificações incorretas dos recursos. Foram realizadas 4 avaliações:

A primeira avaliação (E1) considerou grupos de 50 recursos, totalizando 20 grupos. A segunda avaliação (E2) considerou grupos de 100 recursos, totalizando 10 grupos. A cada grupo avaliado, o algoritmo MCDA reclassifica os recursos e treina o algoritmo de aprendizagem de máquina.

Ao analisar a acurácia obtida pela pré-classificação do algoritmo de aprendizagem de máquina LMT, podemos observar que E1c, obteve a acurácia de 84,2%. O repositório inicial estava com 300 recursos que foram utilizados para treinamento e a cada 50 recursos é feito um novo treinamento.

As avaliações E2b e E2c também se mostram satisfatórias, com acurácia de 78,4% e 80,5%. Estas avaliações foram realizadas a cada 100 recursos, portanto os processos de

Tabela XI
 AVALIAÇÃO DA ACURÁCIA NA PRÉ-CLASSIFICAÇÃO COM APRENDIZAGEM DE MÁQUINA

Qtd	E1a	E1b	E1c	E2a	E2b	E2c	E3a	E3b	E3c	E4a	E4b	E4c
50	14	7	8									
100	13	8	10	26	21	21						
150	8	12	8									
200	11	10	11	20	24	16	41	45	42			
250	11	10	11									
300	14	10	11	22	17	18						
350	7	6	10									
400	11	8	10	13	14	15	35	34	28			
450	5	6	6									
500	6	6	5	14	12	13				109	106	108
550	7	6	14									
600	12	14	8	16	14	27	63	61	77			
650	20	21	6									
700	17	19	9	35	38	16						
750	17	16	5									
800	16	17	6	34	31	12	73	76	24			
850	11	14	11									
900	11	9	19	22	23	45						
950	14	10	7									
1000	7	11	5	26	22	12	46	50	87	213	193	356
TOTAL	232	223	180	228	216	195	258	266	258	322	299	464
Acurácia%	76.8	77.7	82	77.2	78.4	80.5	74.2	73.4	74.2	67.8	70.1	53.6

normalização de dados e cálculos do algoritmo MCDA só é realizado a cada 100 recursos.

A avaliação 4 mostrou-se imprópria para a pré-classificação de recursos, com acurácia abaixo de 54% (E4c). O intervalo de 500 recursos para recálculo do algoritmo MCDA e treinamento da aprendizagem de máquina mostrou-se muito grande, acarretando a perda de acurácia.

2) *Pré-classificação de recursos explorando lógica fuzzy:* Com o emprego de lógica fuzzy avaliamos o impacto na precisão da classificação de recursos. O objetivo da avaliação é reduzir o conjunto de recursos através de pré-classificação com lógica fuzzy para posterior classificação com o algoritmo MCDA.

Os testes foram realizados considerando um repositório de 10, 200, 400, 800 e 1600 recursos classificados através do algoritmo MCDA apresentado na seção 3. Os recursos possuem cinco critérios de qualidade (QoS). Todos os recursos foram avaliados através do algoritmo MCDA e classificados de 1 a 4.

Com o emprego de lógica fuzzy é possível pré-classificar os novos recursos sem a necessidade de compará-lo aos demais recursos, pois utiliza um conjunto de regras lógicas que irão pré-classificar os melhores recursos.

Para exemplificar o modelo proposto um conjunto de 10 recursos com seus atributos de qualidade são classificados inicialmente pelo algoritmo MCDA apresentado e após com o emprego de lógica fuzzy.

A Tabela XII contém cinco atributos de qualidade para dez recursos. Cada linha representa um recurso e cada coluna possui um atributo de qualidade. Nomes dos recursos foram omitidos para reduzir o espaço usado pelas tabelas.

A primeira etapa da classificação é a normalização dos dados. Para tanto, consideramos os vetores $N=\{0,1,1,1,0\}$ e $C=\{5,2,3,2,50\}$. Os atributos *Response Time* e *Latency* qualificam o recurso com valores baixos obtidos pela Eq.(4)

Tabela XII
 EXEMPLO DO DATASET

RT	AV	TH	RE	LA
302.75	89	7.10	81	187.75
482.00	85	16.00	73	1.00
3321.40	89	1.40	73	2.60
126.17	98	12.00	67	22.77
107.00	87	1.90	73	58.33
107.57	80	1.70	67	18.21
255.00	98	1.30	58	40.80
136.71	76	2.80	60	11.57
102.62	91	15.30	67	0.93
200.00	40	13.50	59	41.66

e *Availability*, *Throughput* e *Reability* qualificam com valores altos pela Eq.(3).

Tabela XIII
 ATRIBUTOS NORMALIZADOS

RT	AV	TH	RE	LA
1.70	1.07	0.97	1.19	0.21
1.07	1.02	2.19	1.08	38.56
0.15	1.07	0.19	1.06	14.83
4.07	1.18	1.64	0.98	1.69
4.80	1.04	0.26	1.06	0.66
4.78	0.96	0.23	0.98	2.12
2.02	1.18	0.18	0.98	0.95
3.76	0.91	0.38	0.87	3.33
5.00	1.09	2.10	0.98	41.46
2.57	0.48	1.85	0.98	0.93

A Tabela XIV apresenta os valores dos atributos após aplicar a Eq.(5), ou seja dividir o valor normalizado da Tabela XIII pelo maior valor normalizado de cada coluna, multiplicado pelo peso calculado através da lógica fuzzy $w=\{0.70, 1.00, 0.75, 0.94, 0.75\}$ (Tratador Fuzzy). O valor MCDA é obtido aplicando a Eq.(6), que irá somar todos os valores de atributos em cada linha. Após é aplicada a Eq.(7) que irá qualificar o recurso com um valor que vai de 0 a 100. Será dado o valor 100 para o melhor recurso do *dataset*. O resultado da

classificação está informado na coluna "R". Na coluna "C" está o resultado reclassificado com valores de 1 a 4.

Tabela XIV
ATRIBUTOS CLASSIFICADOS - MCDA

RT	AV	TH	RE	LA	R	C
0.24	0.91	0.33	0.94	0.00	63	2
0.15	0.87	0.75	0.85	0.70	85	1
0.02	0.91	0.07	0.85	0.27	54	3
0.57	1.00	0.56	0.78	0.03	76	1
0.67	0.89	0.09	0.85	0.01	65	2
0.67	0.82	0.08	0.78	0.04	61	2
0.28	1.00	0.06	0.67	0.02	52	3
0.53	0.78	0.13	0.70	0.06	57	3
0.70	0.93	0.72	0.78	0.75	100	1
0.36	0.41	0.63	0.68	0.02	54	3

A Tabela XV apresenta as regras que determinam os Termos Linguísticos e correspondentes valores numéricos, identificando os conjuntos fuzzy associados a cada atributo de QoS. Exemplificando, um serviço com valor no atributo RT está associado a um termo linguístico "High" e um valor linguístico igual a 0.75.

Tabela XV
TERMOS LINGUÍSTICOS

Score	Variables	Values
RT ≤ 100	Very High	1
100 < RT ≤ 300	High	0.75
300 < RT ≤ 500	Medium	0.5
RT > 500	Low	0.25
AV ≥ 90	Very High	1
90 > AV ≥ 80	High	0.75
80 > AV ≥ 70	Medium	0.5
AV < 70	Low	0.25
TH ≥ 20	Very High	1
20 > TH ≥ 13	High	0.75
13 > TH ≥ 9	Medium	0.5
TH < 9	Low	0.25
RE ≥ 80	Very High	1
80 > RE ≥ 70	High	0.75
70 > RE ≥ 60	Medium	0.5
RE < 60	Low	0.25
LA ≤ 1	Very High	1
1 < LA ≤ 2	High	0.75
2 < LA ≤ 4	Medium	0.5
LA > 4	Low	0.25

Tabela XVI
ATRIBUTOS CLASSIFICADOS - FUZZY

RT	AV	TH	RE	LA	R	C
0.5	0.75	0.25	1	0.25	77	1
0.5	0.75	0.75	0.75	1	100	1
0.25	0.75	0.25	0.75	0.5	70	1
0.75	1	0.5	0.5	0.25	82	1
0.75	0.75	0.25	0.75	0.25	75	1
0.75	0.5	0.25	0.5	0.25	60	3
0.75	1	0.25	0.25	0.25	68	2
0.75	0.5	0.25	0.5	0.25	60	3
0.75	0.75	0.75	0.5	1	98	1
0.75	0.25	0.5	0.25	0.25	50	3

Para realizar a classificação final é utilizada a equação $h_{i,j} = w_j v_{i,j}$ e após são aplicadas as Eq.(6) e Eq.(7).

Para avaliar o emprego da lógica fuzzy, no processo de pré-classificação de novos recursos, foram classificados conjuntos de 10, 100, 200, 400, 800 e 1600 recursos do *dataset* QWS.

Inicialmente, os recursos foram classificados através do algoritmo MCDA proposto e o resultado final reclassificado de 1 a 4, conforme Figura 1. O mesmo conjunto foi classificado usando lógica fuzzy e o resultado final reclassificado de 1 a 4.

A Tabela XVII apresenta a acurácia obtida após o ranqueamento de todos os grupos de recursos. Foi avaliada a acurácia entre os dois classificadores (MCDA e Fuzzy), na classificação dos melhores recursos do conjunto, ou seja, dos avaliados com valor "1".

Tabela XVII
ACURÁCIA DA CLASSIFICAÇÃO

Recursos	Acurácia	Selecionados	Excluídos	Redução
10	100%	6	4	40.00%
100	92.00%	72	28	28.00%
200	95.33%	150	50	25.00%
400	93.71%	248	152	38.00%
800	95.90%	350	450	56.25%
1600	96.00%	469	1131	70.69%

Com o resultado obtido pode-se observar que, com o emprego de lógica fuzzy na pré-classificação de novos recursos foi possível uma acurácia de mais de 92% e uma redução dos recursos aptos em até 70,69%.

As técnicas apresentadas para pré-classificar os recursos, no momento em que entram na infraestrutura computacional, evita que seja necessário avaliar todos os recursos presentes a cada requisição do cliente.

VII. TRABALHOS RELACIONADOS

A Tabela XVIII apresenta uma comparação dos principais Trabalhos Relacionados, empregando os seguintes critérios: (i) expressividade na representação de recursos; (ii) utilização de algoritmos MCDA; (iii) aplicação das Preferências do Cliente; (iv) emprego de QoS; (v) emprego de Lógica Fuzzy; e (vi) emprego de Aprendizagem de Máquina (AM). A presença do critério é representada pelo caracter "+" e a ausência por "-".

Tabela XVIII
COMPARAÇÃO DOS TRABALHOS RELACIONADOS

Trabalhos	Expr.	MCDA	Pref.	QoS	Fuzzy	AM
Maheswari [16]	+	+	+	+	+	-
Salah [17]	-	+	+	+	+	-
Perera [18]	+	+	+	-	-	-
Gomes [19]	+	-	-	-	-	-
Almulla [20]	-	+	+	+	+	-
Nunes [21]	-	+	+	-	-	-
Suchithra [22]	-	+	+	+	-	-
Vaadaala [23]	-	+	+	+	-	+
EXEHDA-RD	+	+	+	+	+	+

A especificação de recursos através de linguagens de alta expressividade é realizada nos trabalhos de [16], [18] e [19].

Somente o trabalho de [19] não considera as preferências do cliente, pois trabalha com dados de contexto.

Critérios de Qualidade de Serviço (QoS) são avaliados nos trabalhos de [16], [17], [20], [22] e [23].

A lógica fuzzy é discutida nas obras [16], [17] e [20]. Nenhum deles aplica a lógica fuzzy para resolver a incerteza entre os especialistas.

Pode-se observar que, apenas o trabalho de [23] utiliza Aprendizagem de Máquina. Mas, com o objetivo de mensurar a acurácia de um único atributo de qualidade com o algoritmo J48. Os autores consideraram os resultados satisfatórios e obtiveram uma acurácia de 63%.

VIII. CONSIDERAÇÕES FINAIS

Este artigo introduz a arquitetura de software do EXEHDA-RD contribuindo no tratamento da incerteza referente aos especialistas na definição de pesos para atributos de QoS e ainda, reduzindo o número de recursos qualificados por meio da pré-classificação de novos recursos ao considerar o emprego de aprendizagem de máquina e lógica fuzzy. Como principal contribuição do trabalho desenvolvido até o momento, pode ser destacada a especificação e avaliação do módulo “Classificador de Recursos”.

O modelo de ranqueamento de recursos proposto considera a dinamicidade da infraestrutura computacional provida pela IoT, com uma grande quantidade de consumidores e provedores de recursos.

Com o emprego de lógica fuzzy, na resolução de conflitos entre os especialistas na definição de graus de importância para cada atributo de QoS, foi possível promover uma adequação dos pesos utilizados no algoritmo MCDA, considerando assim os recursos de melhor qualidade no resultado das requisições dos clientes. Também conquistou-se uma redução do esforço computacional dos algoritmos MCDA na classificação dos recursos a cada requisição dos clientes.

O aprendizado de máquina supervisionado, a partir de um conjunto de dados, classificado pelo algoritmo MCDA, promovendo-se melhorias significativas na classificação de recursos com considerável precisão. O algoritmo LMT foi adequado, com até 82% de acerto na pré-classificação.

Com o uso da lógica fuzzy, a partir de um conjunto de dados, classificado pelo algoritmo MCDA, foi possível classificar de forma semelhante os melhores recursos do conjunto de dados com considerável precisão. O uso da lógica fuzzy foi adequado, com 92% a 100% de acurácia na pré-classificação.

As duas propostas de pré-classificação apresentaram resultados satisfatórios para integração ao sub-módulo “Pré-classificador” do EXEHDA-RD. Na continuidade, destacamos duas importantes metas:

- (i) a extensão da avaliação do modelos para classificação considerando o emprego de um *dataset* contendo uma grande quantidade de recursos, e
- (ii) e promoção de novas métricas de avaliação do esforço computacional bem como da acurácia.

REFERÊNCIAS

- [1] B. Research, “Global Sensors in Internet of Things (IoT) Devices Market, Analysis & Forecast: 2016 to 2022,” Tech. Rep., 2017. [Online]. Available: https://www.researchandmarkets.com/research/bvgxvl/global_sensors_in
- [2] L. H. V. Nakamura, S. Carlos, J. C. Estrella, S. Carlos, M. J. Santana, S. Carlos, and S. Carlos, “Using Semantic Web for Selection of Web Services with QoS,” *WebMedia'11: Proceedings of the 17th Brazilian Symposium on Multimedia and the Web.*, pp. 4–7, 2011.
- [3] C. Schröpfer, M. Schönherr, P. Offermann, and M. Ahrens, “A Flexible Approach to Service Management-Related Service Description in SOAs,” in *CEUR Workshop Proceedings*, vol. 234, 2006.
- [4] P. Barnaghi and A. Sheth, “On Searching the Internet of Things : Requirements and Challenges,” *IEEE Intelligent Systems*, vol. 31, no. 6, pp. 71–75, 2016.
- [5] P. A. Khutade and R. Phalnikar, “QoS Aware Web Service Selection and Ranking Framework Based on Ontology,” *International Journal of Soft Computing and Engineering (IJSCE)*, no. 3, pp. 77–81, 2014.
- [6] H. A. Simon, “Why Should Machines Learn?” *Machine Learning: An Artificial Intelligence Approach*, 1983.
- [7] I. H. Witten, E. Frank, and M. a. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. Morgan Kaufmann Publishers Inc., 2011.
- [8] E. Al-Masri and Q. H. Mahmoud, “QoS-based discovery and ranking of Web services,” in *Proceedings - International Conference on Computer Communications and Networks, ICCCN, 2007*, pp. 529–534.
- [9] H. M. Alabool and A. K. Mahmood, “Trust-Based Service Selection in Public Cloud Computing Using Fuzzy Modified VIKOR Method,” *Australian Journal of Basic and Applied Sciences*, vol. 7, no. 9, pp. 211–220, 2013.
- [10] F. A. C. Gomide, R. R. Gudwin, and R. Tanscheit, “Conceitos Fundamentais Da Teoria De Conjuntos Fuzzy , Lógica Fuzzy E Aplicações,” 1988.
- [11] J. Figueira, S. Greco, and M. Ehrgott, “Multiple criteria decision analysis: state of the art surveys,” *Multiple Criteria Decision Analysis State of the Art Surveys*, vol. 78, 2005. [Online]. Available: <http://www.springerlink.com/content/vj266112x6072u62>
- [12] G. H. Tzeng and J. J. Huang, *Multiple Attribute Decision Making: Methods and Applications*, ser. A Chapman & Hall book. Taylor & Francis, 2011.
- [13] Y. Liu, A. H. Ngu, and L. Z. Zeng, “QoS computation and policing in dynamic web service selection,” *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pp. 66–73, 2004. [Online]. Available: <http://doi.acm.org/10.1145/1013367.1013379>
- [14] J. Lopes, R. Souza, C. Geyer, C. Costa, J. Barbosa, A. Pernas, and A. Yamin, “A Middleware Architecture for Dynamic Adaptation in Ubiquitous Computing,” *J-Jucs*, vol. 20, no. 9, pp. 1327–1351, 2014.
- [15] R. Dilli, H. Filho, G. Cavalheiro, A. M. Pernas, and A. Yamin, “Uma Abordagem Dinâmica para Descoberta de Recursos na IoT Explorando Processamento Semântico,” *Simpósio Brasileiro de Computação Ubiqua - CSBC*, pp. 908–917, 2017.
- [16] S. Maheswari and G. R. Karpagam, “Comparative analysis of semantic web service selection methods,” pp. 159–169, 2015.
- [17] N. B. Salah and I. B. Saadi, “Fuzzy AHP for Learning Service Selection in Context-Aware Ubiquitous Learning Systems,” *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Comp., Adv. and Trusted Comp., Scalable Comp. and Communications, Cloud and Big Data Comp., Internet of People, and Smart World Congress*, pp. 171–179, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7816842/>
- [18] C. Perera, A. Zaslavsky, P. Christen, M. Compton, and D. Georgakopoulos, “Context-aware sensor search, selection and ranking model for internet of things middleware,” in *Proceedings - IEEE International Conference on Mobile Data Management*, vol. 1, 2013, pp. 314–322.
- [19] P. Gomes, E. Cavalcante, T. Batista, C. Taconet, S. Chabridon, D. Conan, F. Delicato, and P. Pires, “A QoS-Aware Discovery Service for the Internet of Things,” *Ubiquitous Computing and Ambient Intelligence*, vol. 7656, pp. 344–355, 2016. [Online]. Available: <http://link.springer.com/10.1007/978-3-642-35377-2>
- [20] M. Almulla, H. Yahyaoui, and K. Al-Matori, “A new fuzzy hybrid technique for ranking real world Web services,” *Knowledge-Based Systems*, vol. 77, pp. 1–15, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.knosys.2014.12.021>
- [21] L. H. Nunes, J. C. Estrella, C. Perera, S. Reiff-Marganiec, and A. N. Delbem, “Multi-criteria IoT Resource Discovery: A Comparative Analysis,” *Wiley InterScience*, pp. 1–16, 2016.
- [22] M. Suchithra and M. Ramakrishnan, “A survey on different web service discovery techniques,” 2015.
- [23] V. Vaadaala, “Classification of Web Services Using JForty Eight,” *A National Conference on Recent Trends in Computer Science and Technology International Journal of Electronics Communication and Computer Engineering*, vol. 4, no. 6, pp. 181–184, 2013.