



UTILIZAÇÃO DO MÉTODO ELECTRE TRI-ME PARA CLASSIFICAÇÃO DE MÉTRICAS DE SOFTWARE

Mara Regina dos Santos Barcelos
marabarceloss@gmail.com
UFF

Adriana Manzollilo Sanseverino
adrianams@id.uff.br
UFF

Carlos Francisco Simões Gomes
cfsg1@bol.com.br
UFF

Resumo: O software é um produto usado em larga escala por empresas dos mais diversos ramos, sendo assim, algumas de suas características são cruciais. As empresas desenvolvedoras dispõem de técnicas que podem ser usadas a fim de assegurar a usabilidade, a funcionalidade, a confiabilidade, dentre outros atributos. As métricas de software são uma dessas técnicas, através delas é possível identificar erros de maneira precoce, controlar o número de horas usadas no desenvolvimento, além de outros benefícios de acordo com o tipo de métrica. Neste contexto, o presente trabalho propõe uma abordagem multicritério para classificar as métricas de software, em relação a alguns de seus atributos de qualidade, de acordo com a opinião dos especialistas. Para tanto o método Electre Tri-Me foi utilizado. Os resultados demonstram que as métricas de software podem ser classificadas através da utilização do método Electre Tri-Me, demonstrando que todas as alternativas permanecem na mesma classe nos procedimentos otimista e pessimista. Sendo que a alternativa a1 foi classificada na classe A (contribui muito), enquanto as alternativas a2 e a3 foram classificadas na classe B (Contribui pouco), ou seja, nenhuma alternativa foi classificada na classe C (não contribui).

Palavras Chave: Métrica de software - Atributo de software - Multicritério - ELECTRE Tri-Me -



1. INTRODUÇÃO

O aumento do uso da tecnologia por diversas empresas, de setores distintos, contribui para que a inovação ocorra em períodos cada vez menores, e com o aumento da demanda e exigência em relação aos produtos de *software*. Empresas de diversos ramos utilizam softwares diversificados em suas atividades, visando aumento da produtividade, controle de processos, gestão de recursos, dentre outras atividades. Neste contexto o produto de *software* deve atender as necessidades das empresas de acordo com as especificações solicitadas pela mesma.

De acordo com Figueira Filho *et al.* (2015) o Brasil é uma economia emergente com muitas iniciativas na área de tecnologia da informação (TI), tanto no setor público quanto no privado. Além disso, a produção de *software* pode viabilizar a incorporação do conhecimento em produtos, serviços e sistemas, contribuindo ainda para a difusão da TI entre organizações, instituições e a população em geral (BRITTO & STALLIVIERI, 2010).

As características do ambiente estratégico de TI o tornam muito complexo em situações em que as decisões são rotineiramente requeridas. Essas características incluem mudança acelerada; altos níveis de incerteza; a escolha entre grandes e novos sistemas tecnológicos com poucos precedentes históricos; a priorização de novos projetos e novos sistemas; terceirização versus decisões de centralização versus descentralização; e poucos erros podem ser cometidos (DE SOUZA *et al.*, 2018).

Existem técnicas que podem ser usadas para alcançar esses objetivos, como métricas, testes, metodologias de desenvolvimento, ciclos de vida, dentre outras. A questão é que as técnicas a serem utilizadas não são definidas pelo cliente, e sim pela equipe de desenvolvimento.

No que diz respeito ao uso de métricas, existem vários tipos que podem contribuir para um determinado atributo de qualidade, e que, são utilizadas em diferentes fases do desenvolvimento. Segundo Pressman (2011) métricas de *software* fornecem uma maneira quantitativa de avaliar a qualidade de *software*.

Neste contexto, o presente trabalho tem o objetivo de classificar as métricas de *software*, em relação a atributos de qualidade do mesmo, mediante uma abordagem multicritério, utilizando o método Electre Tri-Me.

Este artigo está dividido em cinco seções. Além dessa introdução, um referencial teórico sobre *software*, métricas e multicritério é apresentado na seção 2, a metodologia utilizada é descrita na seção 3, enquanto os resultados são discutidos na seção 4, e na seção 5 as conclusões do trabalho são relacionadas.

2. REFERENCIAL TEÓRICO

2.1 SOFTWARE

Softwares podem ser usados como produtos ou como serviços, para os mais diversos objetivos, oferecendo às empresas uma possibilidade de ganho de tempo, dinheiro, ou padronização de suas operações, dependendo das necessidades da mesma.

Steinmueller (1995) define *software* como instruções que controlam a operação da tecnologia de informação (TI) de *hardware*. Já para Pressman (2011) um *software* consiste em instruções que fornecem características, funções e desempenho desejados, além de ser possuir uma estrutura de dados que possibilita aos programas manipular informações adequadamente.



2.1.1 ATRIBUTOS DE QUALIDADE DE *SOFTWARE*

A norma ISO/IEC 250120 (2011) define atributo como a propriedade que é inerente ou uma característica de uma entidade que pode ser distinguida de maneira quantitativa ou qualitativa por meios humanos ou automatizados. Os atributos de qualidade usados neste trabalho têm suas definições apresentadas na Tabela 1.

Tabela 1: Definições dos atributos de qualidade de *software*.

Atributo	Definição
Funcionalidade	Grau em que o conjunto de funções abrange todas as tarefas especificadas e os objetivos do usuário.
Confiabilidade	Grau em que um usuário tem confiança de que um produto ou sistema se comportará como pretendido.
Usabilidade	Grau em que um produto ou sistema pode ser usado para atingir metas especificadas com efetividade, eficiência e satisfação em um contexto específico de uso.
Eficiência	Recursos gastos em relação à exatidão e completude com que os usuários alcançam metas especificadas.
Manutenibilidade	Grau de facilidade que um produto ou sistema pode ser modificado por seus desenvolvedores.
Portabilidade	Grau de facilidade com que um sistema, produto ou componente pode ser transferido de um hardware, <i>software</i> ou outro ambiente operacional ou de uso para outro.

Fonte: ISSO-IEC 250120 (2011).

2.1.2 MÉTRICAS DE *SOFTWARE*

A medição pode ser definida como um processo que converte símbolos ou números aos atributos de uma entidade do mundo real (FENTON & BIEMAN, 2014). O processo de medição pode ser realizado tanto no produto, quanto no processo.

Segundo Pressman (2011), o *software* pode ser medido por várias razões, a saber: verificar a qualidade do produto ou a produtividade da equipe; formular estimativas; e, ajudar a justificar os pedidos de novas ferramentas ou treinamento adicional, dentre outros.

- **Métricas de produto**

As métricas de produto (também conhecidas como métricas de qualidade) têm como objetivo medir a qualidade dos sistemas desenvolvidos em um determinado momento, seja na fase de desenvolvimento, ou na fase de manutenção, quando o processo de desenvolvimento estiver finalizado (SOMMERVILLE, 2011).

- **Métricas de processo**

As métricas de processo são um conjunto de teorias e práticas que se relacionam com medidas, permitindo assim, realizar uma estimativa de custo, desempenho e cronograma de um projeto, a fim de levantar uma série de indicadores que podem levar à melhoria do processo. (PRESSMAN, 2011).

- **Métricas de modelo**

Mediante o uso de métricas de modelo é possível mensurar a estrutura do *software* durante o desenvolvimento do mesmo. Sendo assim, a identificação e correção de possíveis



erros acontece no início do ciclo de vida, reduzindo custos e retrabalho. As métricas de modelo podem ser usadas antes da implementação do código e, por isso, permitem uma redução maior no custo de sua correção. (PRESSMAN, 2011). Vários tipos de métricas têm sido utilizados, com objetivos distintos, conforme ilustrado na Tabela 2.

Tabela 2: Trabalhos relacionados a métricas de *software*.

Autores	Ano	Breve descrição do trabalho em relação às métricas
HILTON, R.; GETHNER, E.	2018	Propuseram um método que permite prever a qualidade externa dos módulos do <i>software</i> , correlacionando com métricas internas de qualidade que podem ser coletadas por meio de análise do código-fonte.
KIM, J.; SUNGWON, K.; JONGSUN, A.	2018	Propuseram a métrica de extensibilidade para arquitetura de <i>software</i> (EMSA), que representa o grau de extensibilidade de um sistema baseado em sua arquitetura.
DJEBAR, Y.; KIMOUR, M.; GUERSI, N.	2017	Propuseram uma abordagem baseada na análise e avaliação do processo do produto de <i>software</i> . As métricas são usadas para medir a similaridade entre os sistemas, a fim de racionalizar a linha de produção e promover reutilização.
VENKATARAMAN, S.; AL HUSSEIN, A.; SIDDAPPA, M.	2017	Propuseram 44 tipos de métricas para aumentar a qualidade dos projetos estudantis na indústria de <i>software</i> na Arábia Saudita.
PLAKIDAS, K.; SCHALL, D.; ZDUN, U.	2017	Usaram métricas para medir a atividade e impacto de ecossistemas de <i>software</i> , baseando-se nos metadados da documentação. Isso permitiu categorizar os participantes do ecossistema, tanto no mercado de <i>software</i> quanto na comunidade de desenvolvedores.

Fonte: Barcelos *et al.* (2018).

2.2 MÉTODOS MULTICRITÉRIO

Segundo Gomes, Araya e Carignano (2004) o apoio multicritério à decisão (AMD) pode ser definido como a atividade que, baseada em modelos claramente apresentados, ajuda na obtenção de resposta às questões de um agente de decisão no decorrer de um processo.

O Decisor possui:

- conjunto de alternativas $A = \{a_1, a_2, \dots, a_n\}$, A é um conjunto finito e estável de possíveis alternativas a serem classificadas.
- conjunto de critérios $C = \{c_1, c_2, \dots, c_j\}$, C é um conjunto de critérios de desempenho; uma pré ordem total em cada atributo $c_j(a)$, $a \in A$, a melhor alternativa a no critério c_j ;
- conjunto de m agentes decisórios (D_1, D_2, \dots, D_m) expressando sua opinião sobre as alternativas, mediante as ordenações de preferência (Franceschini e Maisano, 2015).
- conjunto de pesos de critérios $W = \{w(c_1), w(c_2), \dots, w(c_j)\}$, associa cada preferência de agente W é um vetor de peso de atributo satisfazendo $w(c) > 0$ (ANGILELLA *et al.*, 2018).

Os métodos de AMD apoiam o processo decisório, considerando a influência dos critérios e os julgamentos de valor dos decisores, assumindo a presença de subjetividade no



mesmo (Gomes; Gomes, 2014). Os métodos AMD podem ser agrupados em quatro tipos de problemática: (1) problemática $P.\alpha$ para escolha; (2) problemática $P.\beta$ para classificação; (3) problemática $P.\gamma$ para ordenação; (4) problemática $P.\delta$ para descrição (Gomes & Gomes, 2014).

Os resultados obtidos mediante a utilização de métodos AMD podem ser comparados, a fim de assegurar resultados mais confiáveis. Vieira Junior (2008) utilizou a metodologia MACBETH como uma ferramenta para descobrir os limites dos pesos a serem usados na abordagem de restrições de pesos, alcançando um resultado que demonstrou ótima correlação com os métodos ELECTRE III, SMART e PROMETHEE I e II.

Os métodos da família ELECTRE (*Elimination Et Choix Traduisant la Réalité*) utilizam o conceito de sobreclassificação, mediante a verificação das relações de subordinação entre as alternativas (COSTA *et al.*, 2014). Trata-se de uma combinação que inclui três relações do sistema fundamental de relações de preferência: indiferença (aIb), preferência fraca (aQb) e preferência estrita (aPb) (GOMES; ARAYA & CARIGNANO, 2004).

Esses métodos são utilizados em duas etapas: (1) construção da relação de sobreclassificação; (2) exploração da relação de sobreclassificação construída na primeira etapa com objetivo de identificar o conjunto de alternativas dominantes (GOMES; ARAYA & CARIGNANO, 2004).

Govindan e Jepsen (2016) realizaram um levantamento sobre a utilização dos métodos da família Electre, e chegaram a conclusões interessantes: (i) o Electre III é o mais popular; (ii) o Electre I continua sendo utilizado apesar de ter sido desenvolvido a mais de 40 anos; (iii) o Electre Tri tem sido mais utilizado em problemas nas áreas de gestão financeira e de energia; (iv) já as versões Electre Tri-C e Electre Tri-nC possuem aplicações incipientes; (v) vários trabalhos fazem adaptações dos métodos Electre para utilizá-los.

2.2.1 MÉTODO ELECTRE TRI

O método Electre Tri é um dos métodos da escola francesa, da família Electre. É um método de sobreclassificação, como ilustrado na Figura 1. Este método vem sendo utilizado em diversas áreas, em vários problemas de classificação, por exemplo: Educação, Riscos industriais, Satisfação de consumidores, Construção civil, dentre outros ((FREITAS & SILVA, 2014); (COSTA *et al.*, 2007a); (COSTA *et al.*, 200b); (SZAJUBOK, MOTA & ALMEIDA, 2006)).

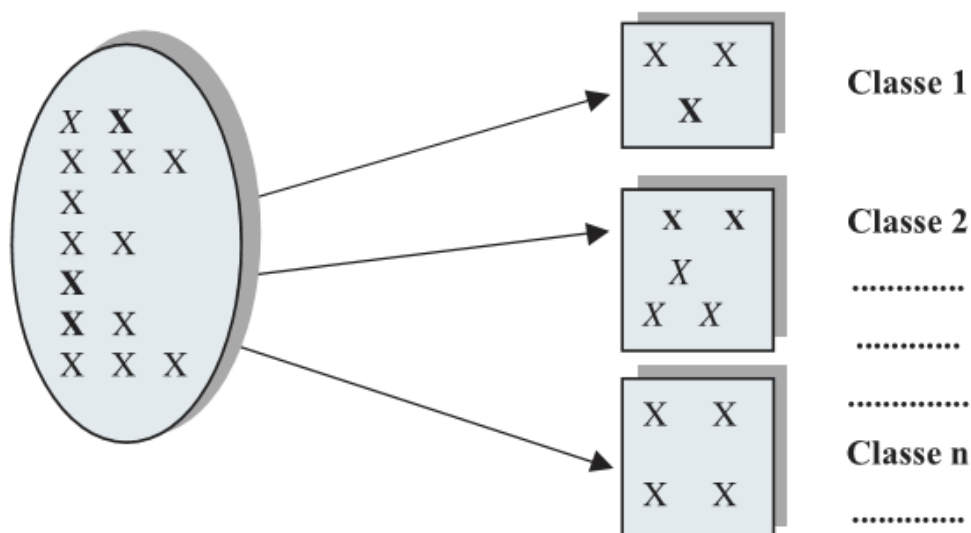


Figura 1: Problemática de classificação do Electre Tri.

Fonte: Costa *et al.* (2007b).



Apesar da larga utilização deste método, Zheng *et al.* (2014) propuseram um algoritmo capaz de destacar os exemplos de atribuição que contraditórios ou não representáveis utilizando o modelo de preferência ELECTRE TRI, pois os autores afirmam que devido a restrições temporais e limitações cognitivas, os decisores podem expressar informações contraditórias, já que suas preferências mudam com o tempo.

O Electre Tri é um método de classificação, ou seja, aloca as alternativas em categorias pré-definidas, a partir da comparação dessas alternativas com perfis definidos de limites das classes (MOUSSEAU & SLOWINSKI, 1998). A Figura 2 ilustra essa questão.

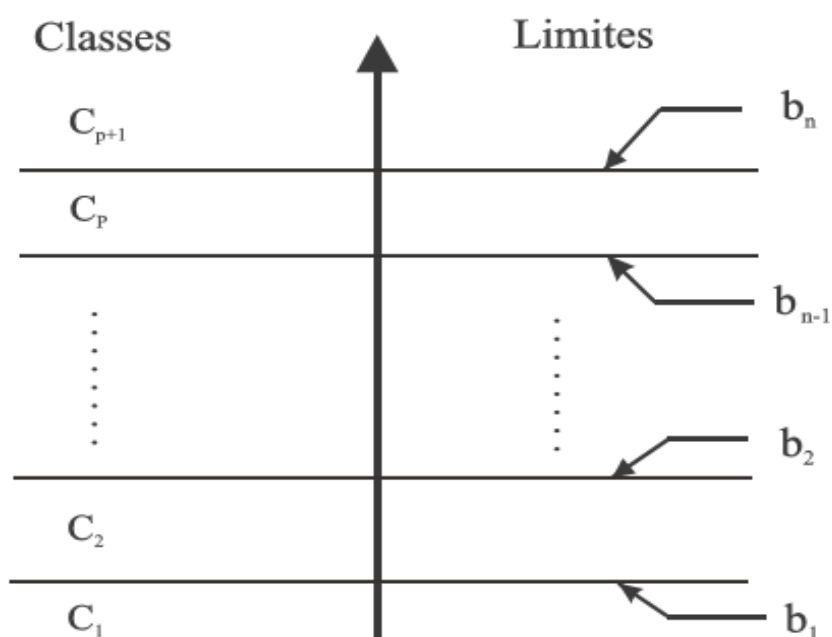


Figura 2: Definição de categorias.

Fonte: Costa *et al.* (2007a).

Esse método constrói relações de sobreclassificação (S), confirmando ou não a informação que a alternativa *a* sobreclassifica a alternativa *b* (aSb), e isto significa que *a* é pelo menos tão boa quanto *b*.

Concordância: para que a afirmação (aSb) seja aceita, a maioria dos critérios deve ser a favor da mesma. A concordância parcial, e a concordância global são calculadas mediante o uso das fórmulas 1 e 2 (MOUSSEAU & SLOWINSKI, 1998), respectivamente.

$$c_j(a, b) = \begin{cases} 0 & \text{se } g_j(b_h) - g_j(a) \geq p_j(b_h) \\ 1 & \text{se } g_j(b_h) - g_j(a) \leq q_j(b_h) \\ \frac{p_j(b_h) + g_j(a) - g_j(b_h)}{p_j(b_h) - q_j(b_h)} & \text{caso contrário} \end{cases}$$

(1)



$$c(a, b) = \frac{\sum_{j \in F} k_j c_j(a, b_h)}{\sum_{j \in F} k_j} \quad (2)$$

Não-discordância: caso a condição de concordância não seja atendida, nenhum critério deve ser contrário a afirmação (aSb). A concordância parcial é calculada mediante o uso da fórmula 3 (MOUSSEAU & SLOWINSKI, 1998).

$$d_j(a, b) = \begin{cases} 0 & \text{se } g_j(b_h) - g_j(a) \leq p_j(b_h) \\ 1 & \text{se } g_j(b_h) - g_j(a) > q_j(b_h) \\ \frac{g_j(b_h) + g_j(a) - p_j(b_h)}{v_j(b_h) - p_j(b_h)} & \text{caso contrário} \end{cases} \quad (3)$$

Veto: um conjunto de limiares de veto é utilizado no teste de discordância, para a construção da subordinação ($v_1(b_h), v_2(b_h), \dots, v_m(b_h)$). (MOUSSEAU & SLOWINSKI, 1998).

Índice de credibilidade: representa o grau de credibilidade da afirmação (aSb). $\sigma(a, b_h) \in [0, 1]$. (MOUSSEAU & SLOWINSKI, 1998). O índice é calculado mediante o uso da fórmula 4.

$$\sigma(a, b_h) = c(a, b_h) \cdot \prod_{j \in F} \frac{1 - d_j(a, b_h)}{1 - c(a, b_h)} \quad (4)$$

As relações de preferência são construídas a partir desses valores. Os valores de $\sigma(a, b_h)$, $\sigma(b_h, a)$ e λ .

- $\sigma(a, b_h) \geq \lambda$ e $\sigma(b_h, a) \geq \lambda \rightarrow aSb_h$ e $b_hSa \rightarrow a$ é *indiferente* a b_h .
- $\sigma(a, b_h) \geq \lambda$ e $\sigma(b_h, a) < \lambda \rightarrow aSb_h$ e *não* $b_hSa \rightarrow a$ é *preferível* a b_h .
- $\sigma(a, b_h) < \lambda$ e $\sigma(b_h, a) \geq \lambda \rightarrow$ *não* aSb_h e $b_hSa \rightarrow b_h$ é *preferível* a a .
- $\sigma(a, b_h) < \lambda$ e $\sigma(b_h, a) < \lambda \rightarrow$ *não* aSb_h e *não* $b_hSa \rightarrow a$ é *incomparável* a b_h .

Segundo Mousseau e Slowinski (1998) a classificação no Electre Tri é feita de 2 maneiras:

- (i) Otimista: consiste em comparar a alternativa a sucessivamente à alternativa b , começando do último perfil (categoria, classe);
- (ii) Pessimista: consiste em comparar a alternativa a sucessivamente à alternativa b , começando do primeiro perfil (categoria, classe). Essa classificação é mais exigente.

2.2.1.1 VARIAÇÕES DO MÉTODO ELECTRE TRI

O método Electre Tri foi alterado ao longo dos anos por diversos pesquisadores, resultando em novas versões, conforme ilustrado na Tabela 3.



Tabela 3: Variações do método Electre Tri.

Método	Breve descrição
ELECTRE Tri-B	Compara as alternativas a alternativas de referência que contêm as características representativas de cada categoria. Cada categoria é definida por meio de uma única alternativa de referência característica. Usa perfis (classes) limitantes.
ELECTRE Tri-C	Compara as alternativas a alternativas de referência que contêm as características representativas de cada categoria. Cada categoria é definida por meio de uma única alternativa de referência característica. Usa perfis (classes) centrais.
ELECTRE Tri-nC	Compara as alternativas a alternativas de referência que contêm as características representativas de cada categoria. Cada categoria pode ser definida por meio de uma ou mais alternativas de referência característica.
ELECTRE Tri-Me	Compara as alternativas a alternativas de referência que contêm as características representativas de cada categoria. Realiza a classificação ordenada de um conjunto de alternativas, considerando a opinião/avaliação/percepção de múltiplos avaliadores sobre o desempenho dos critérios utilizados.

Fonte: (NEPOMUCENO, 2012); (BOUYSSOU & MARCHANT, 2015).

3. METODOLOGIA

Neste contexto, o método ELECTRE Tri-Me será usado neste trabalho a fim de promover a classificação das métricas de *software* na opinião dos especialistas, pois existe mais de um avaliador.

Para este fim foram utilizados dados coletados a partir da aplicação de um questionário, desenvolvido para este fim. A pesquisa em questão foi aplicada no período de 19/02/2018 a 15/03/18, contando com a participação de quatro especialistas.

3.1 DEFINIÇÃO DOS CRITÉRIOS E ALTERNATIVAS

As alternativas consideradas neste trabalho estão dispostas na Tabela 4, no que diz respeito aos critérios estão relacionados na Tabela 5.

Tabela 4: Alternativas utilizadas.

Alternativas	
a ₁	Métricas de Produto
a ₂	Métricas de Processo
a ₃	Métricas de Modelo

Fonte: Elaboração própria (2018).

Tabela 5: Critérios utilizados.

Critérios	
C ₁	Funcionalidade
C ₂	Confiabilidade
C ₃	Usabilidade
C ₄	Eficiência
C ₅	Manutenibilidade
C ₆	Portabilidade

Fonte: Elaboração própria (2018).



3.2 PREFERÊNCIA, VETO E PESO

A Tabela 6 traz os valores usados para preferência forte (P); preferência fraca (Q); Veto (V); Peso (W) para cada critério. Além disso, o valor usado para lambda (λ) foi 1.

Tabela 6: Valores de preferência, veto e peso.

	g1	g2	g3	g4	g5	g6
Q	1.5	1.5	1.5	1.5	1.5	1.5
P	3	3	3	3	3	3
V	1	1	1	1	1	1
W	1	1	1	1	1	1

Fonte: Elaboração própria (2018).

3.3 DEFINIÇÃO DOS LIMITES INFERIORES DAS CLASSES

Foram definidas 3 classes para a utilização do método, a Tabela 7 ilustra os limites inferiores das mesmas.

Tabela 7: Limites inferiores das classes.

Classes	Limite Inferior	Conceitos
A	2,5	Contribui muito
B	1	Contribui pouco
C	0	Não contribui

Fonte: Elaboração própria (2018).

4. RESULTADOS

Para obter a classificação das métricas de *software*, o algoritmo de classificação do ELECTRE TRI-ME foi executado com o apoio do sistema computacional J-Electre (PEREIRA, *et al.*, 2018), que permite a execução dos algoritmos de alguns métodos multicritério da escola francesa, já tendo sido testado e validado para a execução dos métodos: ELECTRE I; I_S; I_V; II; III; IV; TRI e TRI-ME.

Com base nos parâmetros estabelecidos a priori, foi obtida a classificação das métricas de *software*, conforme ilustra a Tabela 8. Como é possível perceber, apenas a alternativa a_1 (métricas de produto) ficou classificada na classe A (contribui muito). A alternativa em questão possui a mesma classificação tanto na visão otimista quanto na pessimista.

Além disso, as alternativas a_2 (métricas de processo) e a_3 (métricas de modelo), permanecem na mesma classe (B) de acordo com os procedimentos pessimista e otimista.

Tabela 8: Classificação das métricas.

Alternativas	Classificação	
	Pessimista	Otimista
a_1	A	A
a_2	B	B
a_3	B	B

Fonte: Elaboração própria (2018).



5. CONCLUSÕES

A demanda de utilização dos recursos de TI está aumentando, e com isso, seus usuários (clientes) estão cada vez mais exigentes. Mas, existem técnicas que podem ser utilizadas para auxiliar na atividade de desenvolvimento de *software*, para assegurar as especificações solicitadas pelos clientes.

De acordo com a opinião dos especialistas, os três tipos de métricas contribuem para a qualidade do *software*, sendo que, a alternativa a_1 contribui mais que as outras.

Sendo assim, o presente trabalho teve como objetivo principal classificar as métricas de *software* mediante o uso do método multicritério Electre Tri-Me. Foram utilizados seis critérios (atributos de qualidade de *software*), e três alternativas (tipos de métrica). Os resultados demonstram que é possível classificar as métricas de *software* com a utilização do método Electre Tri-Me.

Existem outros tipos de métricas que podem ser considerados na pesquisa, com potencial de fornecer um parâmetro mais amplo para as empresas desenvolvedoras de *software*, sendo esta uma limitação do trabalho.

Como sugestão para trabalhos futuros, considera-se ampliar o número de critérios e alternativas, bem como ampliar a participação dos especialistas.

6. AGRADECIMENTOS

Os autores agradecem o apoio fornecido pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERÊNCIAS

- ANGILELLA, S.; CORRENTE, S.; GRECO, S. & SŁOWIŃSK, R. (2016) Robust Ordinal Regression and Stochastic Multiobjective Acceptability Analysis in multiple criteria hierarchy process for the Choquet integral preference model Omega, Volume 63, September, Pages 154-169.
- BARCELOS, M. R. S.; SANSEVERINO, A. M.; GOMES, C. F. S. & SANTOS, M. Um estudo bibliométrico sobre métricas de software. XXXVIII Encontro Nacional de Engenharia de Produção, 2018.
- BOUYSSOU, D & MARCHANT, T. On the relations between ELECTRE TRI-B and ELECTRE TRI-C and on a new variant of ELECTRE TRI-B. European Journal of Operational Research, v. 242, n. 1, p. 201-211, 2015.
- BRITTO, J. & STALLIVIERI, F. Inovação, cooperação e aprendizado no setor de software no Brasil: análise exploratória baseada no conceito de Arranjos Produtivos Locais (APLs). Economia e Sociedade, v. 19, n. 2, p. 315-358, 2010.
- COSTA, H. G.; VILAS BOAS, G. A. de R.; POLICANI, A. L. & GOMES, C. F. S. G. Modelo multicritério para avaliação e classificação da gestão organizacional: proposta e caso de uso. Production, v.24, n.3, p. 521-535, jul./set. 2014.
- COSTA, H. G.; SANTAFÉ JÚNIOR, G. H. P. & HADDAD, A. N. Uma contribuição do método ELECTRE TRI à obtenção da classificação de riscos industriais. Investigação Operacional, v. 27, n. 2, p. 179-197, 2007(a).
- COSTA, H. G.; MANSUR, A. F. U.; FREITAS, A. L. P. & DE CARVALHO, R. A. ELECTRE TRI aplicado a avaliação da satisfação de consumidores. Production, v. 17, n. 2, p. 230-245, 2007(b).
- DE SOUZA, L. P.; GOMES, C. F. S. & DE BARROS, A. P. Implementation of new Hybrid AHP-TOPSIS-2N method in sorting and prioritizing of an it capex project portfolio. INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY & DECISION MAKING, v. 18, p. S0219622018500207, 2018.
- DJEBAR, Y.; KIMOUR, M. & GUERSI, N. A Feature Model Metrics-Based Approach to Develop a Software Product Line. International Arab Journal of Information Technology (IAJIT), v. 14, n. 3, 2017.
- FENTON, N. & BIEMAN, J. Software metrics: A rigorous and practical approach. CRC Press, 2014.



- FIGUEIRA FILHO, F.; PERIN, M. G.; TREUDE, C.; MARCZAK, S.; MELO, L.; DA SILVA, I. M., & DOS SANTOS, L. B.** A study on the geographical distribution of Brazil's prestigious software developers. *Journal of Internet Services and Applications*, v. 6, n. 1, p. 17, 2015.
- FREITAS, A. L. P. & SILVA, V. B.** Avaliação e classificação de instituições de ensino médio: um estudo exploratório. *Educação e Pesquisa*, v. 40, n. 1, 2014.
- FRANCESCHINI, F. & MAISANO, D.** (2015) Checking the consistency of the solution in ordinal semi-democratic decision-making problems. *Original Research Article. Omega*, Volume 57, Part B, December, Pages 188-195.
- GOMES, L. F. A. M. & GOMES, C. F. S.** (2014). *Tomada de decisão gerencial: enfoque multicritério (5ª ed.)*. São Paulo: Atlas.
- GOMES, L. F. A. M.; ARAYA, M. C. G. & CARIGNANO, C.** *Tomada de decisões em cenários complexos: introdução aos métodos discretos de apoio multicritério à decisão*. São Paulo: Pioneira Thomson Learning, 2004.
- GOVINDAN, K. & JEPSEN, M. B.** ELECTRE: A comprehensive literature review on methodologies and applications. *European Journal of Operational Research*, v. 250, n. 1, p. 1-29, 2016.
- HILTON, R. & GETHNER, E.** Predicting Code Hotspots in Open-Source Software from Object-Oriented Metrics Using Machine Learning. *International Journal of Software Engineering and Knowledge Engineering*, v. 28, n. 03, p. 311-331, 2018.
- ISO.** IEC25010: 2011 Systems and software engineering—systems and software quality requirements and evaluation (square)—system and software quality models. *International Organization for Standardization*, v. 34, p. 2910, 2011.
- KIM, J.; SUNGWON, K. & JONGSUN, A.** EMSA: Extensibility Metric for Software Architecture. *International Journal of Software Engineering and Knowledge Engineering*, v. 28, n. 03, p. 371-405, 2018.
- MOUSSEAU, V. & SLOWINSKI, R.** Inferring an ELECTRE TRI model from assignment examples. *Journal of Global Optimization*, v.12, p. 157–174, 1998.
- NEPOMUCENO, L, D, O.** *Electre Tri Me aplicado à classificação de percepções sobre a influência do mestrado no desempenho profissional dos egressos*. Tese (Doutorado em Engenharia de Produção), Universidade Federal Fluminense, Niterói, Brasil, 2012. 126 p.
- PEREIRA, V.; COSTA, H. G. & NEPOMUCENO, L. D. de O.** *Software J-ELECTRE-v1.0*. Disponível em: <<https://sourceforge.net/projects/j-electre/files/>>. Acesso em: 17. mar 2018.
- PLAKIDAS, K.; SCHALL, D. & ZDUN, U.** Evolution of the R software ecosystem: Metrics, relationships, and their impact on qualities. *Journal of Systems and Software*, v. 132, p. 119-146, 2017.
- PRESSMAN, R. S.** *Engenharia de software: uma abordagem profissional*. 7ª Edição. McGraw Hill, 2011.
- SOMMERVILLE, I.** *Engenharia de software*. 9ª Edição. Pearson Prentice Hall, 2011.
- STEINMUELLER, W. E.** *The US software industry: an analysis and interpretative history*. 1995.
- SZAJUBOK, N. K.; MOTA, C M. M. & ALMEIDA, A. T.** Uso do método multicritério ELECTRE TRI para classificação de estoques na construção civil. *Pesquisa Operacional*, v. 26, n. 3, p. 625-648, 2006.
- VENKATARAMAN, S.; AL HUSSEIN, A. & SIDDAPPA, M.** Development of software metrics for improving the quality of the under graduate student projects in computer science/information science/information technology/computer engineering. *IJCSNS*, v. 17, n. 10, p. 212, 2017.
- VIEIRA JUNIOR, H.** Multicriteria approach to data envelopment analysis. *Pesquisa Operacional*, v. 28, n. 2, p. 231-242, 2008.
- ZHENG, J; TAKOUGANG, S. A. M. MOUSSEAU, V. & PIRLOT, M.** Learning criteria weights of an optimistic Electre Tri sorting rule. *Computers & Operations Research*, v. 49, p. 28-40, 2014.