UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ARTES, CIÊNCIAS E HUMANIDADES
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

THAIS RODRIGUES NEUBAUER

Agrupamento interativo aplicado à mineração de processos de negócio

São Paulo

2020

THAIS RODRIGUES NEUBAUER

Agrupamento interativo aplicado à mineração de processos de negócio

Dissertação apresentada à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo para obtenção do título de Mestre em Ciências pelo Programa de Pós-graduação em Sistemas de Informação.

Área de concentração: Metodologia e Técnicas da Computação

Versão corrigida contendo as alterações solicitadas pela comissão julgadora em 10 de março de 2020. A versão original encontra-se em acervo reservado na Biblioteca da EACH-USP e na Biblioteca Digital de Teses e Dissertações da USP (BDTD), de acordo com a Resolução CoPGr 6018, de 13 de outubro de 2011.

Orientador: Profa. Dra. Sarajane Marques Peres

Coorientador: Prof. Dr. Marcelo Fantinato

São Paulo

2020

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

CATALOGAÇÃO-NA-PUBLICAÇÃO

(Universidade de São Paulo. Escola de Artes, Ciências e Humanidades. Biblioteca)

CRB 8 - 4936

Neubauer, Thais Rodrigues

Agrupamento interativo aplicado à mineração de processos de negócio / Thais Rodrigues Neubauer ; orientadora, Sarajane Marques Peres ; coorientador, Marcelo Fantinato. – 2020.
129 f.

Dissertação (Mestrado em Ciências) - Programa de Pós-Graduação em Sistemas de Informação, Escola de Artes, Ciências e Humanidades, Universidade de São Paulo.

Versão corrigida

1. Mineração de dados. 2. Negócios - Processos. I. Peres, Sarajane Marques, orient. II. Fantinato, Marcelo, coorient. III. Título

CDD 22.ed.– 006.312

Dissertação de autoria de Thais Rodrigues Neubauer, sob o título “**Agrupamento interativo aplicado à mineração de processos de negócio**”, apresentada à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo, para obtenção do título de Mestre em Ciências pelo Programa de Pós-graduação em Sistemas de Informação, na área de concentração Metodologia e Técnicas da Computação, aprovada em 10 de março de 2020 pela comissão julgadora constituída pelos doutores:

Profa. Dra. Sarajane Marques Peres

Universidade de São Paulo

Presidente

Prof. Dr. Sylvio Barbon Junior

Universidade Estadual de Londrina

Prof. Dr. Ronaldo Cristiano Prati

Universidade Federal do ABC

Dedico este trabalho à minha família, que possibilitou que eu vencesse mais essa etapa.

Agradecimentos

Agradeço à toda minha família, que valoriza a educação, que me incentivou e me deu possibilidades financeiras de estudar e que sempre me apoiou psicologicamente com muitas horas de conversas sobre todos meus anseios (que nunca foram poucos).

Agradeço à minha mãe, Andréa Rodrigues, por tudo que me proporcionou, mas, principalmente, pelos valores morais e exemplos de conduta, por ter sempre me feito enxergar as belezas da carreira acadêmica e por ter me inspirado, por tantas vezes, no desenvolvimento de habilidades de discurso. Agradeço à minha queridíssima avó, Maria Aparecida Coutinho Cardoso, ou Maricy para os íntimos (e os não tão íntimos assim também), por ter me ensinado e proporcionado tantas coisas, entre elas o exemplo e o gosto por estudar e por ensinar, tão essenciais na carreira que escolhi. Obrigada, vó, por todas as fichas de estudo que fizemos e por todas as horas de estudo juntas.

Ao meu companheiro de vida, meu amor, meu primeiro e único namorado, meu Alexandre Rossi Alvares: muitíssimo obrigada pelo apoio sempre gigantesco em todas as minhas empreitadas, mas especialmente nesses dois últimos anos de muita ansiedade e pouco dinheiro, muitas madrugadas que acabei não dormindo junto com você por ficar trabalhando neste projeto e muitas conversas sobre ele.

À minha orientadora, Profa. Dra. Sarajane Marques Peres, muito obrigada por ser essa profissional que inspira. Sou imensamente grata por termos podido trabalhar juntas todos esses anos. Obrigada por todos os ensinamentos técnicos e não-técnicos, por todas as conversas formais e informais, por toda a paciência e dedicação. Com toda certeza, todo esse mestrado, desde decisão de ingressar, aconteceu em grande devido ao seu envolvimento.

Agradeço ainda ao meu co-orientador, Prof. Dr. Marcelo Fantinato, por toda a dedicação e envolvimento no trabalho, por todas as opiniões e sugestões e por ter sido nosso guia da área de mineração de processos. Obrigada também aos colegas Claudio e Alexandre, pelo tempo dedicado para auxiliar no aprimoramento deste trabalho, e a minha grande amiga Vivian pela paciência de ajudar na revisão deste documento.

Por fim, mas não menos importante, ressalto que o presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo

NEUBAUER, Thais Rodrigues. **Agrupamento interativo aplicado à mineração de processos de negócio**. 2020. 129 f. Dissertação (Mestrado em Ciências) – Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo, 2020.

Os modelos de processos são ferramentas essenciais para alcançar o sucesso no gerenciamento de negócio. No entanto, devido diversos fatores, é comum que as organizações não formalizem esses modelos e, portanto, desconheçam o real processo executado. Em vista disso, a mineração de processo desempenha um papel central e estratégico, pois fornece meios para que os processos organizacionais sejam descobertos, analisados e aprimorados automaticamente. Contudo, a complexidade inerente dos processos organizacionais - especialmente aqueles que não são estruturados - impede que a mineração de processos automatizada seja realizada com total sucesso. Para ajudar na minimização dessa complexidade, a tarefa de agrupamento, denominada agrupamento de *traces* na área de mineração de processos, fornece uma maneira de identificar padrões existentes nos dados mesmo sem informações prévias sobre eles. Os padrões encontrados podem contribuir com a obtenção de resultados de maior qualidade ou mais aderentes a determinados aspectos organizacionais, pois podem auxiliar na redução de um problema grande e complexo em problemas menores e mais simples, referentes a cada perfil de comportamento encontrado. Embora seja claro que muitos avanços têm sido obtidos com agrupamento de *traces*, seu potencial ainda não está totalmente explorado. Por um lado, as soluções trazidas ainda não atendem a todas as expectativas e, por outro, é possível identificar iniciativas para melhoria da aplicação da tarefa de agrupamento na área de mineração de dados que ainda são pouco exploradas na área de mineração de processos. Um exemplo disso é a inserção de conhecimento de um ou mais especialistas humanos diretamente nas decisões tomadas no processo de agrupamento de dados, iniciativa denominada *agrupamento interativo*. O resultado da união de agrupamento interativo e agrupamento de *traces* está sendo chamado neste trabalho de agrupamento de *traces* interativo. Este trabalho explorou como a inclusão do conhecimento de especialistas afeta a qualidade do agrupamento de *traces*, tanto do ponto de vista de índices de avaliação da área de mineração de dados, quanto da área de mineração de processos. Para tanto, a abordagem de agrupamento interativo utilizada foi de restrições *must-link/cannot-link* e foram consideradas representações de *traces* baseadas em modelos semânticos de contagem. Diferentes cenários de experimento no contexto de log de eventos sintéticos e de log de eventos reais provenientes de um processo de gerenciamento de incidentes foram elaborados. A abordagem proposta demonstrou que, apesar de ser observada queda no valor de algumas métricas de qualidade de mineração de processos, é possível obter resultados em que as expectativas de especialistas são atendidas sem causar degradação significativa na qualidade dos grupos resultantes do ponto de vista de mineração de dados e sem perda no aprendizado do algoritmo com a aplicação das restrições. Ainda, no contexto do log de eventos reais, alguns dos resultados de agrupamento de *traces* interativo obtidos demonstraram capacidade de auxiliar no aumento da precisão na tarefa de predição de tempo de resolução de incidentes.

Palavras-chaves: Agrupamento de traços. Agrupamento interativo. Agrupamento de traços interativo. Mineração de processos. Mineração de dados. Gerenciamento de processos de negócio.

Abstract

NEUBAUER, Thais Rodrigues. **Interactive clustering applied in process mining**. 2020. 129 p. Dissertation (Master of Science) – School of Arts, Sciences and Humanities, University of São Paulo, São Paulo, 2020.

Process models are essential tools for achieving business management success. However, due to several factors, it is common for organizations not to formalize these models and thus to be unaware of the actual process being performed. In view of this, process mining plays a central and strategic role as it provides the means for organizational processes to be discovered, analyzed, and improved automatically. However, the inherent complexity of organizational processes - especially those that are unstructured - prevents automated process mining from being performed with complete success. To assist in minimizing this complexity, the clustering task, called trace clustering in the process mining field, provides a way to identify existing patterns in data without the need for prior information about them. These patterns can contribute to the achievement of higher quality or more adherent results to certain aspects of an organizational process, they can help to reduce a large and complex problem in smaller and simpler problems, referring to each behavioral profile encountered. Although it is clear that many advances have been obtained in the area of trace clustering, the potential of the area is not yet fully exploited. On the one hand, it can be seen in the literature in the area that the solutions brought by the trace clustering implementation still do not meet all expectations. On the other hand, it is possible to identify initiatives to improve the application of the clustering task in the area of data mining that are still little explored in the area of process mining, such as the interactive clustering, which inserts the knowledge of one or more human specialists in decision-making process. The result of the union of interactive clustering and trace clustering is here called interactive trace clustering. This paper explored how inserting expert knowledge affects the quality of the trace clustering, both from a data mining and process mining evaluation point of view. To this end, the interactive clustering approach used was a constrained clustering, with must-link/cannot-link constraints, and representations of traces based on semantic counting models were considered. Different experiment scenarios in the context of a synthetic event log and an real-world event log from an incident management process were elaborated. The proposed approach has shown that while some process mining quality metrics were worse, it is possible to achieve results where expert expectations are met without causing significant degradation in the quality of the resulting clusters from data mining perspective and without loss in algorithm learning by applying the constraints. Also, in the context of the real-world event log, some of the interactive trace clustering results demonstrated the ability to assist in increasing the accuracy of the incident resolution time prediction task.

Keywords: Trace clustering. Interactive clustering. Interactive trace clustering. Process mining. Data mining. Business process management.

Lista de figuras

Figura 1 – Esquema da mineração de processos incluindo agrupamento de <i>traces</i> .	16
Figura 2 – Esquema do agrupamento de <i>traces</i> interativo com a inserção do conhecimento especialista no processo de agrupamento de <i>traces</i>	20
Figura 3 – Resultados de agrupamento utilizando diferentes medidas de similaridade	26
Figura 4 – Esquema da interação com especialista na tarefa de agrupamento . . .	27
Figura 5 – Exemplo de log de eventos e de modelo de processo	35
Figura 6 – Exemplos de modelos de processos gerados automaticamente a partir de log de eventos	36
Figura 7 – Exemplo de representação vetorial dos <i>traces</i> presentes em um log de eventos	45
Figura 8 – Esquema de execução dos experimentos	49
Figura 9 – Exemplo de visualização de resultados baseada no valor das características	50
Figura 10 – Exemplo de como a repetição de dados pode alterar o resultado de agrupamento	52
Figura 11 – Exemplo da diferença de movimentação do protótipo com dados sem repetição ou com repetição	53
Figura 12 – Exemplo de log mapeado para o espaço vetorial com perfil em atividades	54
Figura 13 – Modelo de processo usado no exemplo de agrupamento de <i>traces</i>	55
Figura 14 – Modelos de processos gerados com a ferramenta DISCO	56
Figura 15 – Exemplos das nuvens de atividades geradas para três dos grupos resultantes da aplicação do <i>k-means++</i> no log de eventos sintéticos	58
Figura 16 – Índice Silhouette dos agrupamentos do log sintético no cenário 1	59
Figura 17 – Resultados em termos de acertos dos agrupamentos do log sintético no cenário 1	61
Figura 18 – Variação da quantidade de acertos dos agrupamentos do log sintético no cenário 1	63
Figura 19 – Medidas de qualidade de modelo de processos para os resultados do cenário 1	64
Figura 20 – Índice Silhouette dos agrupamentos do log sintético no cenário 2	66

Figura 21 – Medidas de qualidade de modelo processos para os resultados de agrupamento do log sintético no cenário 2	66
Figura 22 – Índice Silhouette dos agrupamentos do log sintético no cenário 3	68
Figura 23 – Resultados em termos de acertos nos agrupamentos com log sintético no cenário 3	69
Figura 24 – Variação da quantidade de acertos dos agrupamentos do log sintético no cenário 3, em 10 execuções	69
Figura 25 – Medidas de qualidade de modelo de processos para os resultados de agrupamento do log sintético no cenário 3	71
Figura 26 – Comparação dos resultados obtidos nos cenários de experimentação com o log sintético	72
Figura 27 – Fragmento do formato original do log real	74
Figura 28 – Resultados obtidos com o algoritmo <i>K-means++</i> e diferentes representações vetoriais do log de incidentes em termos de duração dos incidentes	76
Figura 29 – Valores para o índice Silhouette obtidos com o algoritmo <i>K-means++</i> e diferentes representações vetoriais do log de incidentes	77
Figura 30 – Visualizações das respostas dos especialistas ao primeiro questionário .	77
Figura 31 – Seleção dos resultados obtidos com o algoritmo <i>K-means++</i> e representações vetoriais do log de incidentes com o conjunto de atributos “espec”	80
Figura 32 – Resultados obtidos com a abordagem CMCR no log de incidentes . . .	83
Figura 33 – Resultados obtidos com o algoritmo <i>Cop-Kmeans</i> no log de incidentes .	84
Figura 34 – Resultado de agrupamentos do log sintético do cenário 2	100
Figura 35 – Variação de resultado dos agrupamentos log sintético do cenário 2, em 10 repetições das execuções indicadas	101
Figura 36 – Medidas de qualidade de modelo de processo para os resultados do log sintético do cenário 2	101

Lista de algoritmos

Algoritmo 1 – COP-Kmeans	32
Algoritmo 2 – Procedimento para criação da estrutura de restrições <i>must-link</i>	51
Algoritmo 3 – Procedimento para criação da estrutura de restrições <i>cannot-link</i>	52

Lista de quadros

Quadro 1 – Variações de representação experimentadas para o log sintético	57
Quadro 2 – Exemplo das formas de mapeamento de atributo para atividade no log real de incidentes	75
Quadro 3 – Variações de cada quesito para o experimento do log real de incidentes	75

Lista de tabelas

Tabela 1 – Exemplo de log com atividade, timestamp e recurso	45
Tabela 2 – Exemplo de perfil de <i>trace</i> por atividade	45
Tabela 3 – Exemplo de perfil de <i>trace</i> por recurso	46
Tabela 4 – Exemplo de perfil de <i>trace</i> por transição	46
Tabela 5 – Exemplo de perfil de <i>trace</i> por desempenho	47
Tabela 6 – Exemplo de distribuição dos casos nos grupos	60
Tabela 7 – Comparações dos valores obtidos nas métricas de mineração de processos	71
Tabela 8 – Restrições levantadas junto aos especialistas e selecionadas para aplicação	81
Tabela 9 – Representação canônica da string de busca da revisão sistemática . . .	98
Tabela 10 – Critérios de Inclusão (CI)	98
Tabela 11 – Critérios de Exclusão (CE)	98
Tabela 12 – Critérios de qualidade (CQ)	99

Sumário

1	Introdução	15
1.1	<i>Definição do problema</i>	18
1.2	<i>Hipótese</i>	19
1.3	<i>Objetivos</i>	20
1.4	<i>Métodos</i>	21
1.5	<i>Organização deste documento</i>	23
2	Referencial teórico	24
2.1	<i>Agrupamento interativo</i>	24
2.1.1	<i>Avaliação</i>	28
2.1.2	<i>Formas de interação</i>	29
2.1.3	<i>Abordagem restrições must-link/cannot-link</i>	31
2.2	<i>Mineração de Processos</i>	33
2.2.1	<i>Avaliação</i>	35
2.2.2	<i>Agrupamento de <i>traces</i></i>	41
2.2.3	<i>Representação de <i>traces</i> no espaço vetorial</i>	43
2.3	<i>Agrupamento interativo em mineração de processo</i>	47
3	Agrupamento de <i>traces</i> interativo	49
3.1	<i>Modelagem adotada para a inclusão do especialista em agrupamento de <i>traces</i></i>	50
3.2	<i>Experimentos com logs de eventos sintéticos</i>	54
3.2.1	<i>Os logs de eventos sintéticos</i>	54
3.2.2	<i>Log de eventos com $m = 100$ réplicas do processo base</i>	57
3.2.3	<i>Log de eventos com $m = 5$ réplicas do processo base</i>	66
3.2.4	<i>Considerações finais sobre os experimentos com logs de eventos sintéticos</i>	70
3.3	<i>Experimentos com log de eventos de um processo de gerenciamento de <i>incidentes</i></i>	72
3.3.1	<i>O log de eventos - incidentes</i>	74
3.3.2	<i>Definição de atividades e escolha da representação vetorial</i>	74

3.3.3	Análise dos resultados de agrupamento de <i>traces</i> interativo	79
3.3.4	Considerações finais sobre os experimentos com o log de eventos reais	85
4	Conclusão	86
4.1	<i>Contribuições</i>	88
4.2	<i>Limitações</i>	89
4.3	<i>Trabalhos futuros</i>	90
	Referências¹	91
	Apêndice A – Protocolo para revisão sistemática: agrupamento interativo	97
	Apêndice B – Resultados complementares	100
	Apêndice C – Questionário - interação 1	102
	Apêndice D – Questionário - interação 2	107
	Apêndice E – Questionário - interação 2 - grupo de controle .	121
	Anexo A – Atributos log de dados reais	127

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

1 Introdução

Os modelos de processos são ferramentas essenciais para alcançar o sucesso na gestão de negócios (DUMAS et al., 2018; WESKE, 2007). No entanto, devido razões culturais ou falta de recursos humanos ou materiais adequados, é comum que as organizações não formalizem esses modelos e, por isso, muitas vezes desconheçam o processo realmente realizado em suas operações diárias.

A área de mineração de processos é capaz de fornecer às empresas informação sobre o que de fato ocorre em seus processos, dado que o foco da área é a extração de conhecimento dos logs de eventos gerados durante a execução de diferentes fases do processo de negócio para que seja possível aprimorar o entendimento sobre ele (AALST, 2016; MARUSTER et al., 2002). Dessa forma, a mineração de processos desempenha um papel central e estratégico, pois fornece meios para que os processos organizacionais sejam automaticamente descobertos, analisados e aprimorados.

Em Maita et al. (2017), os autores realizaram um mapeamento sistemático da literatura que mostra como as áreas de mineração de dados e de mineração de processos estão inter-relacionadas e concluíram que a tarefa de agrupamento, que se origina na primeira, está entre as três tarefas mais utilizadas na segunda (23% de 705 estudos primários mapeados) considerando um espectro de nove tarefas de mineração de dados. O interesse nessa tarefa pode ser explicado principalmente por dois fatores: (i) por sua natureza descritiva que permite a identificação de padrões existentes nos dados e, em seguida, fornece condições para compreensão do contexto no qual os padrões ocorrem e porquê eles ocorrem; e (ii) não requer informação prévia sobre esses padrões para que sejam descobertos (SILVA; PERES; BOSCARIOLI, 2016; HAN; KAMBER; PEI, 2011). Para a área de mineração de processos, essas vantagens são especialmente interessantes por conta do desconhecimento supracitado das organizações sobre seus processos. Os perfis descobertos pelo agrupamento podem fornecer às organizações conhecimento sobre aspectos particulares dos processos ou facilitar tarefas posteriores de mineração de processos, reduzindo um problema grande e complexo a problemas menores, mais simples e localizados nos diferentes perfis encontrados. Neste sentido, a aplicação de agrupamento em mineração de processos contribui com a obtenção de resultados de maior qualidade ou mais aderentes a determinados aspectos de um processo organizacional (LEONI; AALST; DEES, 2015).

A tarefa de agrupamento no contexto de mineração de processos é conhecida como agrupamento de *traces*¹ (LEONI; AALST; DEES, 2015; WEERDT et al., 2013; BOSE; AALST, 2009a; SONG; GUNTHER; AALST, 2008). Um *trace* é um “caminho” no processo para chegar do seu ponto inicial ao seu ponto final, consistindo em uma sequência de eventos distintos e ordenados de forma crescente no tempo, de forma que o momento em que o primeiro evento ocorre é anterior ou igual ao momento de ocorrência do segundo. Um *log* de eventos consiste em um conjunto de *traces* e é a fonte de dados para a tarefa de agrupamento de *traces* (LU, 2018), que beneficia diferentes problemas da área de mineração de processos. Alguns exemplos são: descoberta de processos (WEERDT et al., 2013; APPICE; MALERBA, 2016; GRECO et al., 2006; KONINCK et al., 2017), melhoria de qualidade em log de eventos (LU, 2018), busca por variantes de processos em “chão de fábrica” (MEINCHEIM et al., 2017), sistemas de recomendação em processos de negócio (YANG et al., 2017), *concept drift* em processos de negócio (HOMPES et al., 2017).

A figura 1 mostra que a descoberta do modelo de processo pode ser realizada diretamente a partir do log gerado durante as operações diárias da organização. Contudo, o modelo descoberto dessa maneira pode ser mais complexo do que o analista de processos necessita, levando à sua insatisfação com o resultado, representada na figura pela cor vermelha no círculo indicativo da satisfação do analista. Com os *traces* devidamente agrupados com a aplicação de agrupamento de *traces* e, portanto, com a satisfação do analista de dados, representada na figura pela cor verde no círculo indicativo da satisfação do analista, a descoberta de processos aplicada aos sublogs pode auxiliar na obtenção de modelos mais simples para cada grupo e potencialmente melhores do ponto de vista do analista de processos, representada na cor amarela do círculo de indicação de satisfação.

Figura 1 – Esquema da mineração de processos incluindo agrupamento de *traces*

Fonte: Thais Rodrigues Neubauer, 2020

¹ Neste trabalho, a palavra *trace*, escrita na língua inglesa, será usada no decorrer do texto por ser amplamente conhecida na área de mineração de processos.

Embora seja claro que muitos avanços têm sido obtidos na área de agrupamento de *traces*, o potencial da área ainda não está totalmente explorado. Por um lado, percebe-se diante da literatura na área, que as soluções trazidas pela execução de agrupamento de *traces* ainda não atendem a todas as expectativas. Como exemplo, no trabalho de Appice e Malerba (2016), os autores apresentam uma estratégia de co-treinamento que visa maximizar a extração de informação de um log de eventos, fazendo uso de representações de *traces* que capturem diferentes perfis do processo de negócio sob análise. A abordagem dos autores é sempre superior a abordagens correlatas no que diz respeito a índices de qualidade de agrupamento, porém, não apresenta os melhores resultados para todos os casos quando a avaliação considera aspectos de qualidade de modelos de processos. Esse caso, em especial, mostra que qualidade de agrupamento, como classicamente analisada, não necessariamente garante qualidade de solução para mineração de processos.

Por outro lado, é possível identificar iniciativas para melhoria da aplicação da tarefa de agrupamento na área de mineração de dados que ainda são pouco exploradas na área de mineração de processos, como é o caso da inclusão de especialista na execução da tarefa de agrupamento, procedimento chamado de *agrupamento interativo*, ou *interactive clustering* (HU; MILIOS; BLUSTEIN, 2014).

A tarefa de agrupamento é originalmente aplicada em um cenário de aprendizado sem supervisão alguma e, por isso, os dados são particionados com base em informações de similaridade. Decisões, como a função de similaridade a ser utilizada, podem levar a casos de ambiguidade em relação aos grupos aos quais os dados são alocados, como exemplificado na seção 2.1. Como não há supervisão, essas ambiguidades são usualmente resolvidas por meio de tomadas de decisão e suposições estatísticas (AWASTHI; BALCAN; VOEVODSKI, 2017; CODEN et al., 2017; LEI et al., 2017; VIKRAM; DASGUPTA, 2016; CORREA et al., 2015). Um exemplo disso é a geração de dados por uma mistura de gaussianas ou a busca por grupos com certa densidade, sem comprovação de que tais suposições são razoáveis no contexto real dos dados (AWASTHI; BALCAN; VOEVODSKI, 2017; RINALDO, 2010; ACHLIOPTAS; MCSHERRY, 2005; BRUBAKER; VEMPALA, 2008; KALAI; MOITRA; VALIANT, 2010; MOITRA; VALIANT, 2010; BELKIN; SINHA, 2010; CHAUDHURI; DASGUPTA, 2010).

Além da questão de ambiguidade, a escolha da representação de dados usada também pode não possibilitar, diretamente, a análise desejada (LU; FAHLAND; AALST, 2014). Como exemplo, considere um contexto no qual relações de ordem são importantes e uma

representação que distribui os dados em um espaço vetorial de forma independentemente das relações de ordem. Essas e outras decisões técnicas, como a escolha do algoritmo a ser usado e seus hiperparâmetros, influenciam na qualidade dos resultados obtidos (CODEN et al., 2017; LEI et al., 2017; VIKRAM; DASGUPTA, 2016; CORREA et al., 2015).

O escopo deste trabalho de mestrado está limitado à dificuldade inerente às ambiguidades advindas da escolha de funções de similaridade. Um caminho para minimização dessa dificuldade, estudada neste trabalho, é a inserção do conhecimento de especialistas como complemento às funções de similaridade.

No contexto de mineração de processos, mesmo diante da suposição que as organizações não tenham total conhecimento sobre o modelo de seus processos de negócio, os profissionais atuantes nas organizações, certamente, possuem algum conhecimento relevante sobre aspectos e características associadas aos processos. Assim, é provável que a aplicação do conhecimento desses profissionais no processo de agrupamento de *traces* possa trazer resultados mais adequados do que assumir suposições realizadas totalmente de forma não supervisionada. Além disso, com a inclusão do especialista na tarefa de agrupamento, são mais elevadas as chances de que a resposta alcançada vá ao encontro da expectativa “do negócio” e da necessidade de quem vai de fato aplicar as soluções propostas em operações organizacionais. A expectativa na junção de agrupamento interativo e agrupamento de *traces* é a oferta de uma alternativa de solução que crie condições de guiar o processo de agrupamento não apenas por decisões algorítmicas puramente baseadas em suposições iniciais e fixas sobre como *traces* devem ser analisados e considerados similares, mas inserir uma flexibilidade no processo que permita incorporar conhecimento especializado sobre o negócio, diminuindo eventuais diferenças entre qualidade de agrupamento e qualidade de modelos de processos associados ao grupos.

1.1 Definição do problema

A resolução da tarefa de agrupamento de *traces* gera informação que possibilita condições favoráveis à execução de outras tarefas de mineração de processos, permitindo uma compreensão mais refinada de diferentes aspectos e características de um processo. Essa melhora é decorrente da exploração das semelhanças entre os *traces* de um processo. Contudo, algoritmos para resolução da tarefa de agrupamento são guiados por funções

objetivo relacionadas a essas semelhanças, i.e., otimizam uma solução para o problema de forma não supervisionada, e, portanto, a qualidade do agrupamento obtido depende fortemente de como a semelhança entre os *traces* é estabelecida.

Na prática, mesmo bons resultados obtidos na tarefa de agrupamento dos *traces* falham em propiciar condições para que modelos de processos úteis e de qualidade no contexto do negócio sejam descobertos, ou que análises mais precisas e refinadas sejam produzidas diante da execução de algum outro tipo de mineração de processos. Potencialmente, nestes casos, a essência totalmente não supervisionada aplicada ao processo de agrupamento de *traces* exige suposições, eventualmente errôneas, sobre funções de similaridade usadas para análise e comparação dos dados (*traces*).

Assim, o problema delineado para este trabalho de mestrado é: como minimizar o efeito nocivo de escolhas de funções de similaridade não exatamente adequadas ao contexto de negócios no qual agrupamento de *traces* está sendo aplicado.

1.2 Hipótese

O problema delineado para estudo nesta pesquisa advém da natureza não supervisionada da tarefa de agrupamento de *traces*, pois a falta da supervisão exige o estabelecimento de suposições estatísticas. Tais suposições podem degradar a qualidade da solução obtida e a solução que está sendo proposta neste trabalho para resolver esse problema é baseada na hipótese que tais suposições podem ser complementadas por escolhas embasadas em conhecimento de especialistas.

A figura 2 ilustra o contexto, o problema de pesquisa e a hipótese deste trabalho. Ela mostra que o processo descoberto diretamente a partir do log completo pode ser complexo e, por isso, gerar insatisfação do analista de processos, indicada na figura pelo círculo vermelho. O problema de pesquisa é ilustrado pela caixa pontilhada azul, resultante da aplicação de agrupamento de *traces* para facilitar a descoberta de modelos de processo mais simples. Os grupos resultantes são satisfatórios para o analista de dados (círculo verde) e insatisfatórios para o analista de processos (círculo vermelho).

A hipótese, destacada na figura pela caixa pontilhada laranja, é que incluir especialistas no processo de agrupamento de *traces*, implementando o que é denominado neste trabalho agrupamento de *traces* interativo, levará à obtenção de resultados mais aderentes

ao contexto do processo de negócio, sem que a qualidade do agrupamento sofra alguma degradação significativa. Assim, os resultados serão satisfatórios para o analista de dados e para o analista de processos, indicado na figura pelos círculos verdes. Além disso, a solução proposta neste trabalho defende que a qualidade dos resultados de agrupamento de *traces* pode ser melhorada ainda que pouco esforço seja exigido do especialista.

Figura 2 – Esquema do agrupamento de *traces* interativo com a inserção do conhecimento especialista no processo de agrupamento de *traces*

Fonte: Thais Rodrigues Neubauer, 2020

1.3 Objetivos

O objetivo geral deste trabalho de mestrado é aplicar estratégias de agrupamento interativo ao contexto de mineração de processos, implementando agrupamento de *traces* interativo, para verificar se as estratégias de inclusão do conhecimento de especialista encontradas na literatura de mineração de dados se aplicam no contexto de mineração de processos e como essa inclusão pode afetar a qualidade dos: (a) agrupamentos obtidos, considerando o ponto de vista de análise de agrupamento comumente usado por analistas de dados; (b) resultados obtidos em problemas de mineração de processos resolvidos a partir do uso dos grupos descobertos, considerando o ponto de vista de modelos de processos, comumente usado por analistas de processos.

Diante do contexto deste trabalho, o processo para o alcance do objetivo geral delineado permite ainda a elaboração dos seguintes objetivos específicos:

- Avaliar se os modelos de representação de *traces* utilizados influenciam o processo de agrupamento e o processo de agrupamento interativo.

- Quantificar o custo do envolvimento do especialista no processo de agrupamento interativo, em termos do tempo (em minutos ou horas) que dispendido em cada sessão de participação, considerando o contexto de mineração de processos.

1.4 Métodos

A pesquisa proposta neste trabalho de mestrado é de natureza aplicada, estando direcionada a melhorar soluções para problemas da área de mineração de processos, por meio da implementação de agrupamento de *traces* interativo. Em termos de objetivo metodológico, classifica-se como uma pesquisa explicativa, que segundo Gerhardt e Silveira (2009) e Gil (2008), é uma pesquisa na qual a preocupação está em identificar fatores que determinam ou contribuem para a ocorrência de fenômenos. Entende-se como fenômeno, neste trabalho, a maior adequação dos resultados obtidos a partir da inserção de especialistas no processo de agrupamento. É um estudo empírico, dado que as conclusões são baseadas em análise de resultados obtidos a partir de experimentos computacionais, podendo então ser classificada como um pesquisa de Sistemas de Informação dentro da Ciência da Computação experimental (DODIG-CRANKOVIC, 2002).

Visto ser um estudo empírico, o principal procedimento metodológico adotado foi o experimento, neste caso, computacional. Porém, considerando o processo de desenvolvimento da pesquisa como um todo, procedimentos de revisão exploratória da literatura e revisão sistemática da literatura também foram adotados. A revisão exploratória da literatura foi aplicada para exploração de conceitos fundamentais de agrupamento interativo, mineração de processos e agrupamento de *traces*, e para levantamento bibliográfico na área de mineração de processos e agrupamento de *traces*. A revisão sistemática da literatura foi usada para exploração do estado da arte da área de agrupamento interativo. Detalhes do protocolo de revisão sistemática para essa área são fornecidos no Apêndice A.

Em termos de experimentos, para alcançar os objetivos delineados na seção 1.3, este estudo apresentou e analisou resultados referentes às seguintes atividades:

1. Mapeamento vetorial de log de eventos de processos sintéticos e log de eventos de um processo real, utilizando três diferentes modelos semânticos baseados em contagem - binário, *tf* e *tfidf*, definidos na seção 2.2.3 - e, a depender do contexto de cada log de

eventos, diferentes formas de construção das características as quais esses modelos de contagem foram aplicados, como especificado nas seções 3.2 e 3.3.

2. Aplicação de agrupamento de *traces* sobre os mesmos logs e representações vetoriais.
3. Inserção do especialista no processo de agrupamento, aplicando, portanto, o que neste trabalho é chamado de agrupamento de *traces* interativo, sobre os mesmos logs e representações vetoriais, usando a abordagem de *restrições must-link/cannot-link*, detalhada na seção 2.1.
4. Aplicação de soluções para problemas de mineração de processos nos logs originais e reduzidos por meio de agrupamento de *traces*;
5. Avaliação da qualidade dos resultados obtidos tanto do ponto de vista de mineração de dados quanto de mineração de processos.

Para os experimentos de agrupamento de *traces*, foi utilizado o algoritmo *K-means++* e para os experimentos de agrupamento de *traces* interativo, foram aplicados a abordagem de agrupamento interativo *restrições must-link/cannot-link* e o algoritmo *Cop-Kmeans*. A escolha do K-means++ está pautada na popularidade desse algoritmo tanto em mineração de processos, cf. (MAITA et al., 2017), quanto entre os estudos explorados de agrupamento interativo. A escolha da abordagem e do algoritmo utilizados em agrupamento de *traces* interativo está pautada também na popularidade entre estudos explorados de agrupamento interativo. As implementações próprias (da autora) foram desenvolvidas na linguagem Python. A geração de modelos de processos, usando algoritmos de descoberta de modelos, foram realizada a partir de três plataformas: ProM – *Process Mining Workbench*, Disco e Apromore – *Advanced Process Analytics Platform*².

Para medir a qualidade dos agrupamentos obtidos, foi aplicado o índice de validação interna *Silhouette*, definido na seção 2.1.1. Para avaliar resultados obtidos em problemas reais de mineração de processos, foi necessário usar índices de avaliação aderentes ao problema sob resolução. Considerando a tarefa de descoberta de modelos de processo, foram usadas medidas para avaliação dos aspectos de qualidade referentes a completude, precisão, simplicidade e generalização (AALST, 2011), definidas na seção 2.2.1.

Como apresentado na seção 3.3, o log de eventos reais utilizado nos experimentos de agrupamento de *traces* interativo foi o mesmo utilizado por Amaral, Fantinato e Peres (2018a) para aprimoramento da tarefa de predição de tempo de resolução de incidentes.

² ProM (www.promtools.org), Disco (www.fluxicon.com/disco) e Apromore (www.apromore.org).

Trabalhos como Folino, Guarascio e Pontieri (2014) indicam que a aplicação de agrupamento no contexto de predição de séries temporais pode levar a predições mais precisas. Por isso, um dos objetivos desses experimentos foi avaliar se agrupamento de *traces* interativo pode ser mais efetivo do que o agrupamento clássico na obtenção de grupos de incidentes com menor variação de tempo de resolução em cada grupo. Para tanto, além das avaliações supracitadas, os resultados foram também analisados sob ponto de vista da variação do tempo de resolução dos incidentes em cada grupo, com medidas de estatísticas descritiva. Ainda, para alguns dos resultados de agrupamento, foi realizada uma avaliação sob o ponto de vista de um preditor do tempo de resolução³, em termos da medida RMSPE (Root Mean Squared Percentage Error), uma das mais utilizadas no contexto de predição de tempo em séries temporais (ARMSTRONG; COLLOPY, 1992).

1.5 Organização deste documento

Esta dissertação está organizada em quatro capítulos. No capítulo 2, são apresentados a apresentação dos principais conceitos relacionados a este trabalho - sobre a tarefa de agrupamento, as motivações e formas de aplicação de agrupamento interativo e o contexto da área de mineração de processos. No capítulo 3, são apresentados os experimentos realizados com logs de eventos sintéticos e com log de eventos reais de um processo de gerenciamento de incidentes. Por fim, no capítulo 4 são apresentadas as conclusões, acompanhadas de suas principais contribuições, limitações e trabalhos futuros.

O conjunto de apêndice e anexos constantes neste trabalho são descritos abaixo:

- Apêndice A: disponibilização do protocolo da revisão sistemática realizada.
- Apêndice B: apresentação de visualizações complementares dos experimentos.
- Apêndices C e D: disponibilização integral dos primeiro e segundo questionários, respectivamente, aplicados junto aos especialistas.
- Apêndice E: disponibilização do questionário aplicado a um grupo de controle, juntamente com modelo de planilha criado para parte das respostas.
- Anexo A: disponibilização da descrição dos atributos disponíveis no log de eventos reais utilizado.

³ Para a tarefa de predição, foi utilizado o preditor aplicado por Fernandes (2019), baseado em Sistemas de Transição Anotados (STA), proposto por Aalst, Schonenberg e Songa (2011) que enriquece um modelo de processo baseado em transições com informação de tempo aprendida por instâncias anteriores).

2 Referencial teórico

Este capítulo tem o objetivo de apresentar os principais conceitos relacionados a este trabalho. Na seção 2.1, é definida a tarefa de agrupamento e é apresentada a motivação para agrupamento interativo, bem como conceitos relacionados e as principais abordagens encontradas na literatura até o momento. Na seção 2.2, são apresentados os principais conceitos relacionados à mineração de processos, as formas de avaliação dos resultados de descoberta de processos, as estratégias de agrupamento de *traces* que vêm sendo utilizadas e diferentes formas de representação de *traces* mapeados para o espaço vetorial. Por fim, a seção 2.3 discorre sobre a inserção de conhecimento de especialistas no contexto de mineração de processos.

2.1 Agrupamento interativo

A tarefa de agrupamento de dados é definida como o processo de agrupar dados de acordo com a similaridade existente entre eles de forma que dados alocados a um mesmo grupo são mais similares entre si e dados alocados em grupos diferentes são dissimilares entre si. Nesse contexto, similaridades (e dissimilaridades) são geralmente calculadas por meio de medidas de distâncias (HAN; KAMBER; PEI, 2011).

A solução para a tarefa de agrupamento pode ser organizada como um particionamento das linhas de uma matriz que representa um conjunto de dados. Essa matriz é construída de forma que cada linha representa um dado referente a um contexto sob análise, e os valores em suas colunas representam as características que descrevem os dados (HAN; KAMBER; PEI, 2011).

Formalmente, os algoritmos de agrupamento implementados sob estratégias de particionamento têm como entrada uma matriz de dados $X \in \mathbb{R}^{n \times m}$, com n linhas e m colunas representando um conjunto de dados em certo domínio de aplicação. Cada uma das n linhas representam um dado e as m colunas, as características dos dados. Essa matriz é formada por um conjunto de vetores de linhas $\mathcal{N} = \{\vec{x}_1, \dots, \vec{x}_n\}$ e o objetivo é encontrar k partes de $\mathcal{N}$, denotadas por subconjuntos ordenados $\mathcal{K}_p \subseteq \mathcal{N}$, sendo $p \in \{1, \dots, k\}$. Então, o conjunto $\mathcal{K} = \{\mathcal{K}_1, \dots, \mathcal{K}_k\}$ é justamente os grupos de vetores de linhas resultantes de um algoritmo que soluciona a tarefa de agrupamento.

A fim de solucionar a tarefa de agrupamento, são ajustados os parâmetros de uma função com base na maximização de medidas de qualidade aplicadas ao resultado apresentado com o uso de tal função. Assim, dado o conjunto de dados de entrada ($X \in \mathbb{R}^{n \times m}$), é encontrada uma função:

$$\mathcal{G}(X) : \mathbb{R}^{n \times m} \times W \rightarrow \mathcal{K}, \quad (1)$$

na qual W é um vetor de parâmetros ajustáveis por meio de um algoritmo não supervisionado e $\mathcal{K}$ é como definido anteriormente. A função $\mathcal{G}$ mapeia os dados de entrada em X para um conjunto de vetores de linhas distribuídos nos grupos de $\mathcal{K}$, tal que¹:

1. $\mathcal{K}_p \neq \emptyset, p \in \{1, \dots, k\}$;
2. $\bigcup_{p=1}^k \mathcal{K}_p = X$;
3. $\mathcal{K}_p \cap \mathcal{K}_q = \emptyset, p, q \in \{1, \dots, k\}$ e $p \neq q$.

Funções de similaridade podem ser adotadas de forma a guiar o processo de obtenção do parâmetro W . Essas funções são comumente implementadas usando métricas de distância² e representam a maneira como os dados sob análise serão interpretados na tomada de decisão sobre sua pertinência a um determinado grupo. Ainda, outras estratégias podem ser aplicadas em substituição às métricas de distância, tais como a adoção de critérios de densidade de ocupação do espaço dos dados ou informação probabilística de ocorrência conjunta de dados³ (ESTER et al., 1996).

A resolução dessa tarefa pode ser útil em diversas áreas do conhecimento, tais como Mineração de Dados, Estatística, Tomada de Decisão etc. uma vez que revela informação descritiva sobre o conjunto de dados analisado sem exigir informações prévias sobre a organização desses dados, isto é, de forma não supervisionada.

Contudo, a descoberta de perfis referentes aos subconjuntos de dados revelados tem como base medidas de similaridade e, por isso, a divisão dos dados nos grupos depende da função de similaridade escolhida. Isso pode levar a situações como casos de ambiguidade, como ilustrado na figura 3. Nela, são apresentados dois agrupamentos diferentes para o mesmo conjunto de dados no espaço bidimensional, inicializados com as mesmas sementes: em 3(a) utilizando a distância euclidiana e em 3(b), a similaridade cosseno. Nesse exemplo,

¹ Considerando o problema de agrupamento em sua forma básica.

² Para um estudo aprofundado sobre métricas de distâncias, veja Deza e Deza (2009).

³ O uso de uma ou outra estratégia caracteriza o algoritmo de agrupamento como pertencente a classes diferentes de soluções para o problema de agrupamento.

fica clara a diferença entre os agrupamentos obtidos, visto que para o mesmo conjunto de dados, a separação dos três grupos foi diferente.

Figura 3 – Resultados de agrupamento por meio do algoritmo *K-means++*, com três grupos, utilizando como medida de similaridade a distância euclidiana em (a) e a similaridade cosseno em (b). Os círculos menores representam os dados e os maiores, com circunferência escura, representam os centroides (representantes dos grupos). Cada cor representa um grupo.

Fonte: Thais Rodrigues Neubauer, 2020

Ainda que usando uma mesma função de similaridade, diferentes resultados de agrupamento podem ser obtidos a depender de outros fatores (AWASTHI; BALCAN; VOEVODSKI, 2017; HU; MILIOS; BLUSTEIN, 2014), como por exemplo a inicialização de vetores representantes dos grupos (protótipos) ou a ordem dos dados do conjunto analisados em cada iteração do processo de agrupamento. Essa ambiguidade é resolvida em abordagens tradicionais de agrupamento por escolhas de algoritmos e seus hiperparâmetros com base suposições sobre os processos de geração de dados, como supor a geração por uma mistura de gaussianas, ou encontrar grupos com uma certa densidade (AWASTHI; BALCAN; VOEVODSKI, 2017; ACHLIOPTAS; MCSHERRY, 2005; BRUBAKER; VEMPALA, 2008; KALAI; MOITRA; VALIANT, 2010; MOITRA; VALIANT, 2010; BELKIN; SINHA, 2010; RINALDO, 2010; CHAUDHURI; DASGUPTA, 2010).

Porém, essas suposições são feitas sem comprovação ou verificação de que são razoáveis no contexto real dos dados analisados. Além disso, ainda resta nessas alternativas outros problemas, como da representação não refletir aspectos importantes de similaridade existentes em um certo conjunto de dados e seu contexto real (CODEN et al., 2017; LEI et al., 2017; VIKRAM; DASGUPTA, 2016; CORREA et al., 2015).

Assim, ainda cabem discussões no sentido de buscar alternativas aos problemas decorrentes das escolhas que são necessárias realizar para resolver a tarefa de agrupamento.

Uma das formas propostas na literatura para minimizar esses problemas é a inclusão limitada de conhecimento de especialista, combinando julgamentos humanos com aprendizado de máquina. Essa abordagem, conhecida como agrupamento interativo, tem como objetivo ajudar o algoritmo a alcançar resultados mais adequados, levando em conta a expectativa e o conhecimento de especialistas a respeito do conjunto de dados e o contexto real no qual esses dados estão inseridos (CHANG et al., 2016; HU; MILIOS; BLUSTEIN, 2014).

De forma geral, como apresentado no esquema gráfico da figura 4, é obtido um primeiro agrupamento de forma totalmente não supervisionada, ou seja, da mesma forma como é realizado nas abordagens tradicionais de agrupamento: a partir do conjunto de dados, é realizado um pré-processamento que gera uma representação vetorial dos dados; essa representação é utilizada como entrada pelo processo de agrupamento, que resulta em um modelo. A inclusão do especialista na tarefa de agrupamento é realizada apresentando ao especialista informações selecionadas sobre os resultados obtidos até o momento e, a partir disso, o conhecimento do especialista é coletado e inserido no processo de agrupamento. Em geral, as abordagens de agrupamento interativo permitem que o especialista escolha, a cada apresentação de resultados, se deseja incluir conhecimento e executar o processo de agrupamento novamente ou encerrar o processo, gerando o agrupamento final⁴. Quando obtido o resultado final, atividades de pós-processamento podem ser aplicadas.

Figura 4 – Esquema da interação com especialista na tarefa de agrupamento

Fonte: Thais Rodrigues Neubauer, 2020

A formalização da tarefa de agrupamento interativo deve incluir a influência do especialista no processo de agrupamento. Assim, segundo a formalização da tarefa de

⁴ Em algumas abordagens, é definida uma quantidade máxima de iterações com o especialista por restrições de recursos, inclusive restrição do próprio tempo de supervisão disponível

agrupamento apresenta na equação 1, para este trabalho, a tarefa de agrupamento iterativo é definida como:

$$\mathcal{G}_I(X, \oplus) : \mathbb{R}^{n \times m} \times W \rightarrow \mathcal{K}, \quad (2)$$

na qual X , W e $\mathcal{K}$ são como definidos anteriormente, e $\oplus$ representa o conhecimento especializado que é embutido no processo.

A implementação da tarefa de agrupamento iterativo deve incluir a execução do conjunto $A = \{a_1, \dots, a_i\}$ de ações do especialista em cada iteração do processo de agrupamento, com $i \in \{1, \dots, f\}$ e $f \geq 0$, sendo portanto a_i cada uma das ações f possíveis. Note que $f = 0$ significa que a inserção de ações do especialista não é mandatória para todas as iterações do processo de agrupamento.

2.1.1 Avaliação

Independentemente das escolhas relacionadas à execução do agrupamento, na maior parte das vezes, a solução precisa ser avaliada, como no caso de comparações entre diferentes resultados de agrupamentos que serão realizadas neste trabalho. Sem uma definição *a priori* dos grupos, a avaliação de resultados de agrupamento são verificadas usando abordagens apropriados para esse cenário não supervisionado, seguindo os critérios de *compacidade*, pois os dados de um mesmo grupo devem ser o mais similares possíveis, e *separabilidade*, pois os grupos devem ser espaçados (HALKIDI; BATISTAKIS; VAZIRGIANNIS, 2001). Segundo HALKIDI, BATISTAKIS e VAZIRGIANNIS (2001), existem três abordagens para avaliar agrupamentos: (i) índices externos, que comparam o resultado obtido com uma estrutura pré-definida; (ii) índices internos, que fazem a avaliação envolvendo somente os próprios vetores dos dados e os grupos aos quais foram alocados, como na utilização da matriz de similaridade para avaliar quão similar é cada dado aos dados alocados ao mesmo grupo que ele; e (iii) índices relativos, que comparam os resultados obtidos de um agrupamento com os de outros.

O índice Silhouette foi escolhido para avaliação dos agrupamentos obtidos nos experimentos realizados neste trabalho. Dado o conjunto C de grupos obtidos e sendo c

cada um dos grupos e ci o grupo ao qual o dado i foi alocado, índice Silhouette de i é definido como (ROUSSEEUW, 1986):

$$s(i) = \begin{cases} 1 - a(i)/b(i) & \text{se } a(i) < b(i) \\ 0 & \text{se } a(i) = b(i) \\ b(i)/a(i) - 1 & \text{se } a(i) > b(i) \end{cases} \quad (3)$$

na qual $a(i)$ é a dissimilaridade média entre i e todos os dados alocados a ci ; $b(i) = \min(d(i, c)) \forall c \neq ci \in C$; e $d(i, c)$, a média da dissimilaridade entre i e todos os dados alocados ao grupo c . Em outras palavras, $b(i)$ é a dissimilaridade média do grupo competidor mais próximo de i , ao qual, na ausência de ci , i seria alocado.

Em geral, as formas de avaliação dos resultados alcançados com a aplicação de abordagens de agrupamento iterativo são realizadas utilizando os mesmos índices de validação utilizados nas abordagens tradicionais de agrupamento. Em alguns trabalhos, como em Cao et al. (2011) e Cavallo e Demiralp (2018), são aplicadas *surveys* junto aos especialistas em uma tentativa de quantificar como eles avaliam o resultado alcançado.

2.1.2 Formas de interação

Várias são as aplicações encontradas na literatura que incluem interação limitada no contexto da tarefa de agrupamento: mineração de texto (CORREA et al., 2015; AGRAWAL; BAYARDO; SRIKANT, 2000), reconhecimento de atividades humanas em sinais de sensores (HOSSAIN; KHAN; ROY, 2017), identificação de interações humanas em vídeos (KHODABANDEH et al., 2015), análise de gestos (JANG; ELMQVIST; RAMANI, 2014), agrupamento de fotos (WANG et al., 2012), diagnóstico médico (LEI et al., 2017), resultados de busca por termo (CHANG et al., 2016), arqueologia (MAUDER; BOBKOVA; NTOUTSI, 2016), comportamento de motor de aeronaves (GEORGESCU et al., 2015). Essa variedade de domínios de aplicação acrescida fatores como algoritmos subjacentes de aprendizado de máquina amplia a variedade de formas como a interação com especialistas no processo de resolução da tarefa de agrupamento é realizada (CHANG et al., 2016).

Um levantamento bibliográfico que incluiu a exploração de 50 estudos foi realizado para mapear as formas existentes na literatura. Para tanto, as formas de interação foram divididas em duas etapas: (i) a forma como o conhecimento do especialista é coletado; e (ii) a forma como o conhecimento coletado é incluído no processo de agrupamento.

Entre as formas de coleta do conhecimento, as mais comumente utilizadas entre os estudos explorados foram (ordenadas de mais para menos comum):

1. *Redistribuição de dados*. Consiste em redistribuir dado aos grupos, seja por meio da troca de um dado de um grupo para outro grupo, seja por meio da exclusão ou inclusão de um dado a um grupo.
2. *Requisições split/merge*. O especialista pode fazer requisições do tipo *split* ou do tipo *merge* nos grupos apresentados. Uma requisição do tipo *split* (“quebra” ou “divisão”) a um grupo significa que o especialista deseja que o grupo seja dividido em mais de um grupo e uma requisição do tipo *merge* (“junção” ou “união”) a dois ou mais grupos significa que o especialista deseja que esses grupos sejam unidos para formar um único grupo.
3. *Restrições must-link/cannot-link*. O especialista pode adicionar restrições *must-link* para pares de dados (quando ambos devem ser atribuídos a um mesmo grupo) e restrições *cannot-link* (quando aquele par de dados não puder ser atribuído a um mesmo grupo).
4. *Seleção de características*. Para o conjunto de características apresentado, o especialista indica qual(is) delas são mais relevantes.
5. *Ranqueamento de características*. Para o conjunto de características apresentado, o especialista indica a ordem de relevância delas, da mais para a menos relevante.
6. *Ranqueamento de agrupamentos*. Vários resultados de agrupamento são apresentados ao especialista e ele os ordena de forma que o mais adequado (mais próximo do resultado esperado por ele) fique na primeira posição.

Entre as formas de inclusão do conhecimento coletado no processo de agrupamento, as mais comumente utilizadas entre os estudos explorados foram (ordenadas da mais para a menos comum):

1. Aprendizado de medida de similaridade. O conhecimento do especialista é modelado de forma que a similaridade entre os pares de dados supervisionados reflitam a relação de similaridade indicada pela supervisão.
2. Execução de *split/merge*. Em geral, são utilizados algoritmos de agrupamento espectral⁵ para aplicar as requisições *split* (AWASTHI; BALCAN; VOEVODSKI, 2017;

⁵ Agrupamento espectral é baseado na ideia de particionamento de um grafo gerado partir da matriz de similaridade (LUXBURG, 2007).

CHAUDHURI; DASGUPTA, 2010) e a aplicação das requisições *merge* é feita pela união dos dados dos grupos envolvidos em um só grupo.

3. Aplicação de restrições *must-link/cannot-link*. Para sua implementação, o algoritmo utilizado é da classe denominada de *constrained clustering*, ou, em tradução livre, *agrupamento com restrições*. Nesse tipo de algoritmo, em cada interação e para cada dado do conjunto de dados, é verificado se existe alguma restrição do tipo *must-link* com algum dos dados já alocados e, nesse caso, o grupo ao qual o dado em questão deve ser alocado já está definido. Ainda, antes de alocar um dado a um grupo, é verificado se existe alguma restrição do tipo *cannot-link* para que isso ocorra, ou seja, se já foi alocado ao grupo algum dado com o qual o dado a ser alocado tem uma restrição do tipo *cannot-link*.

Vale mencionar que a apresentação dos resultados aos especialistas é bastante dependente do tipo de dado no qual o agrupamento é aplicado e do objetivo de agrupamento. Quando o agrupamento é feito sobre dados na forma de imagens, por exemplo, a representação gráfica do resultado do agrupamento pode ser feita utilizando os próprios dados, ou seja, as próprias imagens, mostrando quais delas foram alocadas em cada grupo.

De modo geral, para dados que não abrangem vídeos e imagens, o que foi encontrado na literatura para apresentar o resultado do agrupamento foi a utilização de projeções dos dados para espaços dimensionais menores (na maior parte dos casos, em 2D). Além disso, dependendo do contexto, do objetivo do agrupamento e da quantidade de dados envolvidos, em alguns estudos é feita a opção de apresentar para o especialista somente partes dos resultados. Essa é uma proposta para reduzir o esforço do especialista, escolhendo os dados com os quais há maior ganho na supervisão, ou seja, os dados que são mais difíceis de terem seu grupo definido. Esse tipo de iniciativa é o assunto da área de pesquisa denominada *active learning* (CORREA et al., 2015).

2.1.3 Abordagem *restrições must-link/cannot-link*

A abordagem de agrupamento interativo *restrições must-link/cannot-link*, encontrada na literatura de mineração de dados para incluir o conhecimento humano no processo de agrupamento, foi a escolhida neste trabalho para ser aplicada nos experimentos de agrupamento de *traces* interativo. Para tanto, o conhecimento humano proveniente dos

especialistas envolvidos, coletado conforme detalhado no capítulo 3, foi modelado para restrições do tipo *must-link/cannot-link*. Neste trabalho, $d_i \sim d_j$ denota a existência de uma restrição do tipo *must-link* entre os dados d_i e d_j e $d_i \bowtie d_j$, a existência de uma restrição do tipo *cannot-link*.

Essas restrições, juntamente com a representação vetorial do conjunto de dados, serviram de entrada para o algoritmo de agrupamento com restrições *Cop-Kmeans* utilizado. Seja D a representação vetorial do conjunto de dados, m o conjunto de restrições *must-link* $\subseteq D \times D$ e c o conjunto de restrições *cannot-link* $\subseteq D \times D$, o algoritmo *Cop-Kmeans* definido por Wagstaff et al. (2001) é ilustrado na adaptação do pseudo-código 1.

Algoritmo 1 COP-Kmeans(D, m, c)

$C_1 \dots C_k$: centroides iniciais dos k grupos.

Enquanto não há convergência **faça**:

Para $i \leftarrow 0$ **até** n **faça**:

 Associe o dado d_i ao grupo C_j mais próximo que não viole m e c .

Se C_j não existir **então**

Retorne $\{\}$

Para $i \leftarrow 0$ **até** k **faça**:

 Atualize o centroide do grupo C_i .

Retorne $\{C_1 \dots C_k\}$

Fonte: Adaptado de Wagstaff et al. (2001)

Como é apresentado no pseudo-código, uma nova interação acontece enquanto não há convergência. A convergência pode ser definida de várias formas e a escolhida neste trabalho foi a de alteração da localização dos centroides ser menor do que um limiar estabelecido em $1e^{-4}$. Em cada iteração, para cada dado d_i do conjunto de dados, d_i é associado ao grupo C_j mais próximo que não viole as restrições *must-link* presentes em m ou *cannot-link* presentes em c e, se C_j não for encontrado, o agrupamento não tem solução. Todas as relações de similaridade foram calculadas neste trabalho utilizando a distância euclidiana.

Durante os estudos desse algoritmo, foi notado que a ordem em que é realizada a associação dos dados aos grupos pode evitar a situação de não ser encontrado grupo para algum dos dados por conta de violação de restrições. Para exemplificar, suponha $d_i \bowtie d_j$ e $d_i \bowtie d_k$ e que o agrupamento realizado seja de dois grupos. Se os dados d_j e d_k forem associados antes de d_i e cada um a um grupo distinto, não haverá grupo no qual d_i possa ser associado. Porém, se o dado d_i for associado antes de d_j e d_k , certamente d_j e d_k serão associados ao outro grupo e há solução para agrupamento. A fim de evitar que a ordem

interfira em situações como essa, todos os dados que contém alguma restrição do tipo *cannot-link* com outro dado podem ser associados antes dos que não estão envolvidos em restrições desse tipo (RUTAYISIRE et al., 2011). Neste trabalho, não houve casos em que a solução não fosse encontrada e o algoritmo original foi utilizado.

A implementação do *Cop-Kmeans* utilizada neste trabalho foi a disponível em um repositório público de códigos-fonte⁶ por Babaki (2017), criada com base trabalho de Wagstaff et al. (2001). Quanto a essa implementação, foi realizada uma adaptação com relação ao formato de fornecimento dos conjuntos M e N de restrições *must-link* e *cannot-link*, respectivamente, para otimizar o tempo de execução do algoritmo.

Originalmente, a implementação tinha como entrada as listas de restrições M e N em termos de pares de dados e a partir delas eram geradas estruturas do tipo chave-valor (ECV), nas quais a chave correspondia ao identificador de cada um dos dados e o valor correspondia a lista de dados com os quais o dado identificado na chave tinha relação por restrição. Contudo, durante a criação dessa ECV, eram verificadas ainda: (i) as relações de transitividade decorrentes das restrições do conjunto M , pois se $d_i \sim d_j$ e $d_j \sim d_k$, então $d_i \sim d_k$; e (ii) a consistência das restrições para evitar situações do tipo $d_i \sim d_j$ e $d_i \bowtie d_j$ e assegurar $d_i \sim d_j$ e $d_j \sim d_i$. A adaptação foi realizada para já fornecer as restrições no formato ECV, com garantia das devidas transitividades e consistências.

2.2 Mineração de Processos

Um *processo* pode ser genericamente descrito como um conjunto de *atividades* (ou tarefas), como elas devem ser executadas e em qual ordem. De acordo com Weske (2007), um *processo de negócio* consiste, também, em um conjunto de atividades as quais são executadas de forma coordenada em um ambiente organizacional e técnico para alcançar, conjuntamente, uma meta de negócio. Para Dumas et al. (2018), um processo é uma cadeia de eventos, atividades e decisões.

Considerando um processo em particular, define-se como um *caso*, ou *trace*, uma sequência de execuções das atividades relacionadas ao processo. Cada uma dessas execuções é dita ser um *evento*, o qual pode ser descrito em termos de atividade executada, hora e data de execução, recursos envolvidos na execução, entre outras informações. Uma coleção de *traces* compõe um *log de eventos*, no qual são registradas as execuções de um processo

⁶ <https://github.com/Behrouz-Babaki/COP-Kmeans>

observado durante um período de tempo. Uma *trace* de processo de negócio é um caso concreto nos negócios operacionais de uma empresa.

Do ponto de vista organizacional, um bom gerenciamento dos processos de negócio garante resultados consistentes e permite tirar vantagens de oportunidades de melhoria, agregando valor para a organização e seus clientes (DUMAS et al., 2018). Uma das principais ferramentas para seguir com o gerenciamento de processos é o modelo do processo. De acordo com Weske (2007), um *modelo de processo de negócio* é um conjunto de modelos de atividade e restrições de execução entre esses modelos. Segundo Dumas et al. (2018), a principal razão pela qual se modela processos é para facilitar o entendimento deles e para compartilhar esse entendimento com as pessoas que estão envolvidas com eles. Os autores acrescentam que, mais do que isso, modelar os processos permite identificar situações típicas e situações adversas e tomar decisões para gerenciá-los.

Mesmo com a importância reconhecida dos modelos de processo, por vezes o modelo de processo formal não está disponível em uma organização. Essa ausência caracteriza a existência de um conjunto de *traces* de processo que seguem um modelo de processo desconhecido. Essa situação ocorre, em grande parte, devido ao grande volume de atividades realizadas no contexto dos processos de negócio, em todos os tipos de organizações.

Ademais, mesmo quando o modelo de processo formal está disponível, ele raramente está em *fitness* com o que ocorre na vida real, ou seja, o modelo provavelmente não corresponde à execução real das atividades organizacionais. Os processos da vida real ligados às atividades organizacionais derivam principalmente do comportamento de agentes humanos que, por sua vez, precisam entender os padrões e detalhes dos processos para que as organizações possam atingir pleno sucesso. É nesse contexto que surge a motivação para a área de mineração de processo.

De acordo com Aalst (2011), o objetivo da área, mais precisamente descrita por mineração de processos de negócio, é melhorar a compreensão desse processo por meio da extração de conhecimento de logs de eventos, gerados durante a execução de um processo de negócio. Nessa área, os dados relacionados a eventos e os modelos de processos são confrontados: i) para a descoberta desses modelos (gráficos) de processo a partir de uma análise dos logs de eventos (*descoberta*); ii) para reprodução e análise de dados provenientes dos, ou relacionados aos, gráficos de processo (*fitness*); e iii) para fornecimento de informações para melhorar os processos (*melhoria*) (AALST, 2016). Essas atividades caracterizam os três tipos básicos de mineração de processos: descoberta, conformidade

e aprimoramento de processos. Na figura 5(b), é apresentado um modelo de processo descoberto a partir do log da figura 5(a), revelando um modelo de comportamento estruturado mesmo tendo sido extraído a partir de uma análise do log de eventos, que pode ser visto como uma análise de dados não estruturados.

Figura 5 – (a) Exemplo de um log de eventos simplificado; (b) Exemplo de modelo de processo extraído do log de eventos apresentado em (a). O processo começa com a atividade A e termina com a atividade G. As atividades B e C representam um *gateway* do tipo AND, representado pelo símbolo “+”, e atividades E e F representam um *gateway* do tipo XOR, representado pelo símbolo “X”.

Trace	Eventos
1	A,B,C,D,E,G
2	A,C,B,D,E,G
3	A,B,C,D,F,G
4	A,C,B,D,F,G

(a)

Fonte: Thais Rodrigues Neubauer, 2020

2.2.1 Avaliação

Em estudos de mineração de processos, os seguintes tipos de processos são discutidos: “lasanha” e “espaguete”. Exemplos de modelos de processos dos dois tipos, gerados automaticamente, são ilustrados na figura 6(a) e 6(b), respectivamente. De acordo com Aalst (2016), os processos do tipo lasanha têm uma estrutura clara, com relativamente poucas exceções, e as partes interessadas têm uma compreensão clara do fluxo de trabalho envolvido no processo. Neste caso, todas as atividades são repetíveis e possuem entradas e saídas bem definidas. Segundo o mesmo autor, os processos do tipo espaguete têm características que não permitem que as pré e pós-condições para as atividades sejam definidas, e que são orientadas por experiência, intuição, julgamento e erro, regras heurísticas e qualitativas ou informação vaga. Há também uma taxonomia que é considerada como um *continuum*, e inclui os tipos de processos “estruturados”, “semi-estruturados” e “não estruturados”. Processos estruturados correspondem à categoria “lasanha”. Processos não estruturados correspondem à categoria “espaguete”. Aalst (2016) afirma que em processos semiestruturados, os requisitos para as atividades são conhecidos e é possível esboçar modelos para eles, mas algumas atividades no processo requerem julgamento humano, o que pode ser um obstáculo para seguir o curso modelado.

Figura 6 – Exemplos de modelos de processos gerados automaticamente a partir de log de eventos: (a) Exemplo de contexto estruturado com modelo do tipo “lasanha”; (b) Exemplo de contexto não-estruturado com modelo do tipo “espaguete”. A ferramenta ProM⁷ foi utilizada para geração desses modelos.

Fonte: Adaptado de Maita (2016)

Várias medidas de qualidade foram propostas para analisar os modelos de processo descobertos a partir de logs de eventos (LU, 2018). Aalst (2011) apresenta quatro critérios principais para verificação da qualidade de modelos descobertos a partir de logs de eventos: *fitness*, *simplicidade*, *generalização* e *precisão*. Em Aalst, Adriansyah e Dongen (2012) foram definidas formas de calcular os valores para cada um desses critérios. Para isso, o conceito de *alinhamento* foi definido como a relação entre os eventos no log de eventos L e os elementos do modelo M analisados, lembrando que nem todas as atividades presentes no log, conjunto A_L de atividades, estão também presentes no modelo, que contém um conjunto A_M de atividades. Para alinhar um modelo e um log, foram também definidos *movimentos*: $x \in A_L$ para “mover x no log”; $y \in A_M$ para “mover y no modelo”; e $\perp$ para denotar *nenhum movimento*, quando os movimentos no log não podem ser imitados pelo modelo e vice-versa. Considere σ_L um *trace* no log e σ_M um *trace* no modelo. Um alinhamento de σ_L e σ_M é uma sequência $\gamma \in A_{LM}$ tal que a projeção do primeiro (ignorando $\perp$) leva a σ_L e a projeção do segundo (ignorando $\perp$) leva a σ_M . Abaixo, são apresentados dois exemplos de alinhamento:

$$\gamma 1 = \begin{array}{|c|c|c|c|c|} \hline a & c & d & e & h \\ \hline a & c & d & e & h \\ \hline \end{array} \text{ e } \gamma 2 = \begin{array}{|c|c|c|c|c|c|c|} \hline a & b & \perp & d & e & g & \perp \\ \hline a & \perp & c & d & e & \perp & h \\ \hline \end{array}$$

nos quais $\gamma 1$ representa o alinhamento dos *traces* $\sigma_L = abdeh$ e $\sigma_M = abdeh$ e $\gamma 2$ representa o alinhamento dos *traces* $\sigma_L = abdeg$ e $\sigma_M = acdeh$. Com os movimentos indicados na

vertical, o primeiro movimento de γ é (a, a) , indicando que tanto o log L quanto no modelo M fazem o movimento a . O movimento $(\perp, \perp)$ é um movimento ilegal, que deve ser excluído do conjunto de movimentos para que o alinhamento seja válido.

Mais de um alinhamento pode ser possível entre dois *traces* (σ_L e σ_M) e um que contenha mais movimentos nos dois *traces* (do tipo (a, a)) e/ou menos $\perp$ pode ser preferível. Para qualificar um alinhamento γ , uma função de distância em movimentos legais é também definida: $\delta \in A_{LM} \rightarrow \mathbb{N}$, associando custos para os movimentos do alinhamento da seguinte forma, considerando a função de distância definida em Aalst, Adriansyah e Dongen (2012):

- $\delta(x, \perp) = 1$;
- $\delta(\perp, y) = 1$;
- $\delta(x, y) = 0$ se $x = y$;
- $\delta(x, y) = \infty$ se $x \neq y$.

A distância total de um alinhamento pode ser generalizada pela soma dos custos dos movimentos individuais: $\delta(\gamma) = \sum_{(x,y) \in \gamma} \delta(x, y)$. Ainda, $\lambda_M(\sigma_L)$ define um alinhamento ótimo (menor custo e menor distância) de qualquer *trace* $\sigma_L \in L$ com o modelo M .

Abaixo, cada um dos quatro critérios principais para verificação da qualidade de modelos descobertos a partir de logs de eventos definido por Aalst (2011) serão apresentados, juntamente com a formulação de seu cálculo correspondente:

- *Fitness*. É considerado bom quando o modelo permite todos os comportamentos existentes no log de eventos, ou seja, se todos os *traces* do log podem ser reproduzidos do início ao fim utilizando o modelo descoberto, o *fitness* é máximo. Seu valor neste trabalho é definido como em Aalst, Adriansyah e Dongen (2012), o custo total de alinhamento do log L com o modelo M (numerador) normalizado por meio da divisão desse custo total pelo maior custo possível (denominador):

$$Fitness(L, M) = 1 - \frac{fcost(L, M)}{move_L(L) + |L| \times move_M(M)} \quad (4)$$

sendo L o log; M o modelo descoberto; $fcost(L, M)$ o custo total de alinhamento do log L com o modelo M , definido por $fcost(L, M) = \sum_{\sigma_L \in L} \delta(\lambda_M(\sigma_L))$; $move_L(L)$ o custo total de movimentos por todo o log sem nenhum movimento no modelo, definido por $move_L(L) = \sum_{\sigma_L \in L} \sum_{x \in \sigma_L} \delta(x, \perp)$; e $move_M(M)$ o custo total de realizar movi-

mento somente no modelo, definido por $move_M(M) = \min_{\sigma_M \in \beta(M)} \sum_{y \in \sigma_M} \delta(\perp, y)$, sendo $\beta(M)$ o conjunto de todas as sequências completas de execução do modelo M . O resultado desse cálculo resulta em um valor no intervalo de 0 a 1, no qual 1 significa que todos os *traces* do log podem ser reproduzidos a partir do modelo descoberto.

- Simplicidade. É baseada no princípio da *Navalha de Occam*, que afirma que “não se deve aumentar, além do necessário, o número de entidades necessárias para explicar alguma coisa”, ou seja, deve-se procurar o “modelo mais simples” que explica o que é observado no conjunto de dados. O princípio *Minimal Description Length* (MDL) tenta operacionalizar o princípio da *Navalha de Occam* (GRÜNWALD, 2007). Qualquer regularidade nos dados deve ser usada para comprimí-los, ou seja, descrevê-los usando menos símbolos do que o número de símbolos necessários para descrever os dados literalmente. Quanto mais regularidades houver, mais os dados poderão ser compactados e, quanto mais compactados, mais aprendemos sobre os mesmos. Várias são as formas de quantificar a complexidade de um modelo. Neste trabalho, a complexidade será definida pela contagem de nós e arcos nele presentes.
- Generalização. O critério de *generalização* pode ser obtido através do cálculo apresentado por Aalst, Adriansyah e Dongen (2012) e está relacionado à necessidade dos modelos de processo em encontrar um equilíbrio entre sobreajuste e subajuste. Sobreajuste, é o problema de um modelo muito específico que só explica o log de uma amostra particular, só permite o comportamento exato observado; já subajuste ocorre quando o modelo super generaliza os exemplos vistos no log, ou seja, permite mais comportamentos do que os observados mesmo quando não há indicações no log que sugiram esse comportamento adicional. Um modelo no qual subajuste ocorre não é um modelo *preciso* (AALST, 2011). A forma de quantificação desse critério neste trabalho será a definida em (AALST; ADRIANSYAH; DONGEN, 2012):

$$generalizacao(L, M) = 1 - \frac{1}{|\epsilon|} \sum_{e \in \epsilon} p_{new}(w, n) \quad (5)$$

sendo L o log; M o modelo; e um evento; ϵ o conjunto de eventos diferentes presentes no log L ; $p_{new}(w, n)$ a probabilidade estimada de que a próxima visita ao estado s^8 revele um novo caminho ainda não conhecido; w a quantidade de atividades

⁸ Todos os eventos de um log de eventos podem ser vistos como uma observação de uma atividade no estado s

diferentes que saem do estado s ; e n a quantidade de vezes que o estado s foi visitado no log de eventos.

- **Precisão.** Um modelo preciso é aquele no qual todos os comportamentos permitidos pelo modelo que também são observados no log. Para sua quantificação, neste trabalho será usada a definição de Munoz-Gama (2014), baseada no conceito de arcos do tipo *escaping*, arcos (transições) presentes no modelo que não são observados no log ou que são observados no log mas não estão presentes no modelo:

$$precisao(L, M) = 1 - \frac{\sum_{s \in S_{LM}} \omega(s) \times |esc(s)|}{\sum_{s \in S_{LM}} \omega(s) \times |mod(s)|} \quad (6)$$

sendo L o log; M o modelo; s um estado; S_{LM} o conjunto de todos os estados no log e no modelo; $\omega(s)$ o peso do estado s , determinado pela quantidade de ocorrências de s no log L , ou seja, se o estado s existe somente no modelo, seu peso $\omega(s)$ é 0; $|esc(s)|$ quantidade de arcos do tipo *escaping*; e $|mod(s)|$ quantidade de *arcos modelados*. Essa quantificação resulta em valores no intervalo de 0 a 1, quanto mais próximo de 1 é o valor de precisão obtido, maior é a parte dos comportamentos permitidos pelo modelo que também são observados no log.

A qualidade de modelos descobertos pode ainda ser avaliada em relação ao sistema que produziu o log de eventos a partir do qual o modelo foi descoberto, usando as medidas de precisão e revocação (*recall* em língua inglesa) do sistema (LU, 2018). Para tanto, Aalst e Weijters (2006) definem essas duas medidas da seguinte forma:

- **Precisão Estrutural.** Mede a fração do conjunto $C2$ de conexões existentes no modelo original $M1$ simplificado que também aparecem no novo modelo $M2$ (também simplificado), que apresenta o conjunto $C2$ de conexões:

$$precisao^E(M1, M2) = \frac{|C1 \cap C2|}{|C2|} \quad (7)$$

O modelo original $M1$ refere-se ao modelo do sistema e o novo modelo $M2$, ao modelo descoberto. Os modelos simplificados correspondem a um grafo simples derivado do modelo completo com conectores do tipo AND, XOR etc. no qual cada nó representa uma atividade e cada arco entre as atividades correspondem a algum tipo de conexão, qualquer que seja ela.

- *Revocação*. Mede a fração do conjunto $C1$ de conexões existentes no modelo original $M1$ simplificado que também aparecem no novo modelo $M2$ (também simplificado), que apresenta o conjunto $C2$ de conexões:

$$revocacao(M1, M2) = \frac{|C1 \cap C2|}{|C1|} \quad (8)$$

Um fator que impacta fortemente os modelos descobertos é a questão da acurácia da ordenação dos eventos no log estudado, pois os algoritmos para descoberta de modelos de processos deduzem dependências causais entre atividades com base nessa ordem de execução dos eventos do log. Vários fatores podem impedir que a ordenação total de eventos seja conhecida, dificultando a exatidão do modelo de processo descoberto (LU; FAHLAND; AALST, 2014):

- A granularidade da informação temporal do evento pode não ser capaz de diferenciar os momentos de execução das atividades. Por exemplo, se dois eventos, a e b , ocorreram no mesmo dia, mas em horários diferentes, na ordem $\{b, a\}$ e somente o dia de ocorrência é armazenado, o *trace* pode conter $\{a, b\}$ mesmo que seja ordenado pelo *timestamp* armazenado.
- Eventos contendo exatamente a mesma informação temporal podem ser ordenados aleatoriamente em *traces* sequenciais. Dessa forma, três eventos, a, b , e c , que ocorreram exatamente no mesmo horário, podem ser ordenados como $\{a, b, c\}$ em um *trace* e como $\{b, a, c\}$ em outro.
- A ordenação de eventos totalmente baseada em informação temporal pode ser ilusória, pois dois eventos podem não ter relação causal mas acontecerem em certa ordem.

Uma ordenação parcial de eventos, na qual eventos a e b podem ocorrer não-ordenadamente ou concorrentemente, alivia os problemas de exatidão do modelo de processo descoberto que podem surgir por conta dos fatores apresentados (LU; FAHLAND; AALST, 2014), pois ajuda a diminuir o número de dependências causais falso positivas impostas a tais eventos por uma ordenação total.

Uma forma de simplificar esse complexo processo de descoberta de modelos, análise e melhoria de processos é a divisão do log em perfis de *traces* que corresponderão a sublogs, ou seja, a partes do log que têm comportamento semelhante. Com essa divisão, o contexto para outras tarefas de mineração de processo pode ser reduzido a cada perfil encontrado,

o que potencialmente facilita a resolução da tarefa. Na subseção 2.2.2, são apontadas estratégias de agrupamento de *traces* que vêm sendo utilizadas e na 2.2.3, diferentes formas de representação de *traces* mapeados para o espaço vetorial são discutidas.

2.2.2 Agrupamento de *traces*

As técnicas de mineração de processos existentes alcançam resultados bons quando aplicadas no cenário de processos estruturados, contudo, as tarefas ficam mais complexas e problemáticas quando estão contextualizadas em cenários de processos não-estruturados (SONG; GUNTHER; AALST, 2008). A área de mineração de processos tem maior interesse justamente nos cenários de mais diversidade, variabilidade e flexibilidade, que tendem a levar a modelos de processos mais complexos (SONG; GUNTHER; AALST, 2008). Um exemplo típico seria um processo de tratamento no ambiente de um hospital, no qual é vital que as pessoas desviem de procedimentos para lidar com diferentes circunstâncias que podem mudar com certa facilidade ou frequência. A tarefa conhecida na mineração de processos como agrupamento de *traces* é especialmente útil para decompor os logs de eventos em sublogs, potencialmente contendo perfis de *traces* de processos simplificados em relação à caracterização do processo como um todo, revelando padrões de comportamentos. Sobre esses sublogs, os diferentes tipos de mineração de processos podem ser implementados mais facilmente (LEONI; AALST; DEES, 2015).

As estratégias de agrupamento de *traces* que vêm sendo utilizadas em mineração de processos podem ser divididas em três categorias (APPICE; MALERBA, 2016; LU, 2018): as baseadas em sequência de *traces*, as baseadas em modelo e as baseadas em vetor de características. É importante ressaltar que essas categorias não são excludentes, ou seja, uma representação de *traces* pode ser baseada em mais de uma delas.

Estratégias baseadas em sequência de *traces* tratam o *trace* da forma como ele é apresentado no log de eventos, operando em toda sua extensão e levando em conta a sintaxe das sequências de *traces*. Para análise de similaridade, essas estratégias fazem uso de métricas de distância de sequências, como a distância de Levenshtein. Um *trace* pode ser editado de forma a se igualar a outro *trace*, adicionando e removendo eventos, e a distância entre dois *traces* é medida pelo número de operações de edição necessárias para que eles tornem-se iguais. Depois de calcular as medidas, os algoritmos de agrupamento

de propósito geral⁹ baseados em distância podem ser implantados. Lu (2018) aponta os estudos de Bose e Aalst (2009a) e Bose e Aalst (2009b) como exemplos de utilização dessa categoria de estratégia, sendo que em ambos foi identificada a utilização de algoritmos hierárquicos de agrupamento para descoberta de modelos de processo.

Levando em conta o objetivo de descobrir modelos melhores por meio do agrupamento de *traces*, a definição de distância entre *traces* foi alterada em alguns trabalhos para a qualidade do modelo descoberto a partir de tais *traces*, caracterizando as estratégias de agrupamento de *traces* como baseadas em modelos. A ideia principal é que a distância de um *trace* para um grupo é menor se um modelo mais completo, preciso e simples pode ser descoberto a partir do grupo. Assim, o objetivo para agrupamento de *traces* é definido como agrupar *traces* de forma que os modelos intermediários usados para representar os grupos sejam de alta qualidade e satisfaçam critérios predefinidos (por exemplo, *recall* e precisão). Segundo Lu (2018), os trabalhos de Cadez et al. (2003), Weerdt et al. (2013), Ferreira et al. (2007), Hompes et al. (2017) e Koninck et al. (2017) aplicam estratégias dessa categoria para resolver os problemas de agrupamento de *traces* propostos. Em Cadez et al. (2003), foi proposto o aprendizado de uma mistura de modelos de Markov de primeira ordem a partir do comportamento do usuário em um conjunto de dados de navegação da web, aplicando o problema *Expectation Maximization*. Os demais trabalhos, em termos de algoritmos de agrupamento, utilizam algoritmos baseados no *K-means*, mas seus objetivos são diferentes: Ferreira et al. (2007) e Weerdt et al. (2013) aplicam agrupamento de *traces* para descobrir modelos, Hompes et al. (2017), para detectar casos que desviam do modelo de processo e Koninck et al. (2017), para encontrar variantes do modelo de processo.

A última categoria de estratégias de agrupamento de *traces*, baseada em vetor de características, mais semelhantes às ideias de agrupamento de dados tradicional, é a estratégia de representação de *traces* adotada neste trabalho. Nela, os *traces* são mapeados para um modelo de espaço vetorial baseado na construção da representação de perfis a partir, por exemplo, da frequência de atividades, da frequência de transições entre atividades, dos recursos envolvidos etc. Entre esses vetores de características, várias medidas de distância da área de mineração de dados são reutilizadas para estimar a

⁹ Como algoritmos de agrupamento de propósito geral entende-se aqueles desenvolvidos na área de estudo de agrupamento, sem que sejam especialmente voltados a uma classe de problemas caracterizada para um tipo de aplicação especial. Por exemplo, um algoritmo de agrupamento desenvolvido especialmente para lidar com as características de mineração de processo não é considerado como um algoritmo de propósito geral, neste trabalho.

semelhança entre *traces*. Sobre essa representação, algoritmos de agrupamento de propósito geral são aplicados, como *K-means*.

Com os vetores resultantes do mapeamento de *traces* para um espaço vetorial, métricas de distância (similaridade/dissimilaridade) válidas nesses espaços (por exemplo, Euclidiana e Jaccard) podem ser aplicadas, bem como os algoritmos de agrupamento baseados em distância, como *K-means* ou hierárquicos aglomerativos. Em Lu (2018), os trabalhos de Appice e Malerba (2016), Bose e Aalst (2009b), Greco et al. (2006), Hompes et al. (2017), Song, Gunther e Aalst (2008) e Song, Gunther e Aalst (2008) são apontados como exemplos de estudos que aplicaram estratégias baseadas em vetor de características. Greco et al. (2006) apresentou uma das primeiras abordagens que incorporou o agrupamento de *traces* em algoritmos de descoberta de processo. Para tanto utilizou algoritmos de agrupamento do tipo k-cluster (*K-means*, *K-median* etc) e hierárquicos. Lu (2018) aponta que em Song, Gunther e Aalst (2008) as características dos *traces* em uma dimensão são agrupadas em chamados *perfis de traces* e em um trabalho posterior, (SONG; GUNTHER; AALST, 2008), o efeito da aplicação de técnicas de redução de dimensionalidade é investigado em relação ao desempenho do agrupamento de *traces*; em ambos os trabalhos, são utilizadas técnicas de métricas de distância e agrupamento de dados clássicos, como *K-means*, para encontrar variantes do modelo de processo. Em Appice e Malerba (2016), o objetivo também é encontrar variantes, mas o algoritmo utilizado é hierárquico. Para descobrir modelos, Bose e Aalst (2009a) utiliza também algoritmos de agrupamento hierárquicos. Por fim, Hompes et al. (2017) faz uso de algoritmos do tipo *k-cluster* para detectar casos de desvio do modelo de processo.

Visto que a estratégia de agrupamento de *traces* baseada em vetor de características baseada será a estratégia adotada neste trabalho, as formas de mapeamento de *traces* para o espaço vetorial, são discutidas na seção 2.2.3.

2.2.3 Representação de *traces* no espaço vetorial

Uma maneira de realizar o mapeamento de *traces* de processos para um espaço vetorial é baseado em perfis construídos a partir (i) da ocorrência de atividades, (ii) dos recursos, (iii) das transições entre atividades dos *traces*, ou ainda (iv) das estatísticas descritivas derivadas do comportamento dos *traces* (APPICE; MALERBA, 2016).

Nos três primeiros tipos de perfis, sua base (atividade, recurso ou transição) representa uma característica no vetor de características que mapeia os *traces* para espaço vetorial. Neste trabalho, a construção dos perfis foi realizada utilizando os modelos semânticos baseados em contagem medir o valor de cada característica em cada *trace*: *binário*, *tf* (*term frequency*) e *tfidf* (*term frequency-inverse document frequency*). Essas contagens, do contexto de mineração de textos comumente adaptadas para o contexto de vários tipos de mineração de dados, diferem-se na forma como a ocorrência de cada característica (colunas da matriz dos dados) é valorada para cada dado (linhas da matriz de dados, que, no caso de agrupamento de *traces*, são *traces*): (i) na binária, célula terá valor igual a 1 se houve ocorrência da característica naquele *trace* (independente da quantidade de ocorrências) ou igual a zero para o caso contrário; (ii) na *tf*, o valor corresponde à quantidade de ocorrências; e (iii) na *tfidf*, a quantidade de ocorrências no *trace* é ponderada pela proporção da quantidade de *traces* nos quais a característica ocorre em relação a quantidade total de *traces* no log, de forma que características que ocorrem muitas vezes em todos os *traces* tenham valor tão pequeno na matriz quanto características que ocorrem poucas vezes e em poucos *traces*, visto que em ambos os casos, as características não são relevantes e, portanto, devem ter pouco peso nos modelos aos quais são aplicadas (BAEZA-YATES; RIBEIRO-NETO, 1999).

A figura 7 exemplifica a representação binária para o log de eventos mostrado na figura 5(a), em termos de contagem de atividade em (a) e em termos de contagem de transição em (b). Nessas figuras, cada quadro em tons de azul representa uma matriz de dados binária (ou real, a depender de como as ocorrências são computadas), cada linha representa um *trace*, cada coluna uma característica descritiva do *trace* e cada célula em azul é valorada com a contagem da ocorrência. Neste exemplo, as representações geraram, respectivamente, espaços vetoriais de sete e dez dimensões.

A representação de perfil baseada na ocorrência de *atividades* define cada *trace* com foco nas atividades que nele ocorreram, construindo uma característica no vetor de características por atividade encontrada no log de eventos. Sendo uma característica, atividade pode ser medida considerando os métodos de contagem binário, *tf* ou *tfidf*. Considere o log apresentado na tabela 1, que apresenta um log de eventos genérico simplificado, contendo, para cada evento, as informações de *trace*, atividade, timestamp e recurso (nesse caso, as pessoas que executaram a atividade presente no evento) relacionadas ao evento descrito na linha (evento). Um exemplo dessa representação pode ser acompanhada na tabela 2,

Figura 7 – Exemplo de representação vetorial dos *traces* presentes em um log de eventos (figura 5(a)): o tom escuro de azul representa contagem $\neq 0$ para uma atividade/transição (coluna) no *trace* (linha); e o tom claro, representa contagem $= 0$ para uma atividade/transição (coluna) no *trace* (linha)

Fonte: Thais Rodrigues Neubauer, 2020

que ilustra como é o resultado dessa representação, contendo sete diferentes atividades e considerando a contagem *tf*. Com essa abordagem, portanto, cada *trace* é mapeado para um vetor de sete valores de características, um para cada atividade distinta e o valor de cada característica no *trace* mede o número de vezes que a atividade é ocorreu no *trace*.

Tabela 1 – Exemplo de log de eventos genérico e simplificado, com informações de atividade, timestamp e recurso de cada evento

Trace	Atividade	Timestamp	Recurso
1	A	2010-12-30 11:02	Ana
1	B	2010-12-31 10:06	Beto
1	C	2011-05-01 15:12	Caio
1	D	2011-01-06 11:08	Davi
1	E	2011-01-07 14:24	Ana
2	A	2010-12-30 11:32	Caio
2	C	2010-12-30 12:12	Caio
2	F	2010-12-30 14:16	Ana
2	D	2010-01-05 11:22	Davi
2	G	2010-01-08 12:05	Eva

Fonte: Adaptado de Appice e Malerba (2016)

Tabela 2 – Exemplo de perfil de *trace* por atividade, com método de contagem *tf*

Trace	A	B	C	D	E	F	G
1	1	1	1	1	1	0	0
2	1	1	1	1	0	1	1

Fonte: Adaptado de Appice e Malerba (2016)

Ainda, representação de perfil baseada em *recurso* é definida pela perspectiva organizacional, que foca nos recursos de um *trace*. Semelhante ao que ocorre no perfil por

atividade, no perfil por recurso cada recurso presente no log representa uma característica e seu valor é medido por contagem binária, *tf* ou *tfidf*. O resultado dessa representação tendo como base o log apresentado na tabela 1 e a contagem *tf* é apresentado na tabela 3. Os eventos utilizam cinco recursos Ana, Beto, Caio, Davi e Eva e, portanto, os *traces* são traduzidos em vetores de cinco características. Essas características são medidas pela quantidade de ocorrências de cada uma nos eventos do *trace* em questão (contagem *tf*).

Tabela 3 – Exemplo de perfil de *trace* por recurso, com método de contagem *tf*

Trace	Ana	Beto	Caio	Davi	Eva
1	2	1	1	1	0
2	2	0	2	1	1

Fonte: Adaptado de Appice e Malerba (2016)

A representação de perfil baseada em transição é definido a partir da perspectiva de fluxo de controle, que se concentra na ordenação de atividades dos *traces*. As características, nesse caso, representam as relações de sequência direta entre as atividades do *trace*. Para qualquer combinação de duas atividades, por exemplo (A, B), uma característica levando em conta a transição é definido por (A:B) representando as vezes em que um evento com a atividade denominada A foi diretamente seguido por outro evento com a atividade chamada B no *trace*. O resultado dessa representação tendo como base o log da tabela 1 é apresentado na tabela 4. Nele, oito transições formam as características que representam os *traces* e o valor de cada uma delas em cada *trace* foi medida pela contagem *tf*.

Tabela 4 – Exemplo de perfil de *trace* por transição, com método de contagem *tf*

Trace	A:B	A:C	B:C	C:D	C:F	D:E	D:G	F:D
1	1	0	1	1	0	1	0	0
2	0	1	0	0	1	0	1	1

Fonte: Adaptado de Appice e Malerba (2016)

Por fim, a representação de perfil baseada em desempenho foca nos *timestamps*. Esse perfil possui um conjunto predefinido de características: o tamanho do *trace* (número de eventos no *trace*); a duração de tempo do *trace* (diferença de tempo computada entre o último e o primeiro eventos do *trace*); a diferença mínima, máxima, média e mediana de tempo calculadas entre eventos consecutivos no *trace*. O resultado dessa representação tendo como base o log apresentado na tabela 1 é apresentado na tabela 5, contendo vetores de seis características, uma para cada valor de desempenho predefinido.

Tabela 5 – Exemplo de perfil de *trace* por desempenho

Trace	Tamanho	Duração	Mínimo	Máximo	Média	Mediana
1	5	195.3	20.10	125.1	48.84	23.07
2	5	216.5	0.67	141.1	54.14	2.07

Fonte: Adaptado de Appice e Malerba (2016)

Na área de mineração de texto e recuperação de informação, diferentes abstrações são feitas a partir das contagens de ocorrências palavras e documentos, o que possibilita gerar representações que extraem características importantes dos textos. Usando algumas abstrações dessas áreas, um *trace* e um log de eventos podem ser interpretados como um documento e um corpus, respectivamente; e atividades, recursos e transições podem ser vistos como termos/palavras. Isso significa que as representações usando as métricas conhecidas *tf-idf* (term frequency $\times$ inverse document frequency) (BAEZA-YATES; RIBEIRO-NETO, 1999; SILVA; PERES; BOSCARIOLI, 2016) e vetores semânticos (TURNERY; PANTEL, 2010) podem ser explorados no contexto de mineração de processos e, portanto, podem fornecer informações aprimoradas sobre os *traces* usadas pelos algoritmos de agrupamento.

Em Koninck, Broucke e Weerdt (2018), a ideia de vetores semânticos para processos de negócio foi implementada por meio do desenvolvimento de técnicas de aprendizado de representação. Esse paradigma de representação foi aplicado a atividades, *traces*, logs e modelos, a fim de aprender, baseados em uma arquitetura de rede neural, vetores altamente informativos (vetores semânticos) e de baixa dimensionalidade, frequentemente chamados de *embeddings*. A ideia é que esses vetores possam ser usados para tarefas automatizadas de inferência, como agrupamento de *traces*, comparação de processos, monitoramento de processos preditivos, detecção de anomalias, etc. Em uma avaliação experimental, foi mostrado o poder de tais representações derivadas no contexto de agrupamento de *traces*.

2.3 Agrupamento interativo em mineração de processo

Foi encontrada na literatura o trabalho de Koninck et al. (2017) que propõe a inclusão do conhecimento de especialista em um processo de agrupamento que os autores classificam como semi-supervisionado. Na solução descrita, a ideia central é combinar a definição de perfis realizada previamente por especialista com o agrupamento de *traces* utilizando o log de eventos.

Essa combinação pode ser feita utilizando *consensus clustering* para quantificar com que frequência dois elementos são agrupados em diferentes soluções e, em seguida, construir um particionamento final com base nessa quantificação. Porém, se houver apenas duas soluções para combinar, a do especialista e a solução algorítmica de agrupamento de *traces*, aplicar um *consensus clustering* pode ser difícil por ser principalmente útil para combinar um número maior de soluções. Assim, no caso de um único agrupamento de especialistas esteja disponível, uma estratégia diferente é proposta: a partir desse agrupamento, reagrupar os *traces* agrupados pelo especialista cuja atribuição ao grupo atual diminui a qualidade final do agrupamento em relação ao desempenho em outros objetivos, como a qualidade do modelo de processo subjacente ao grupo de *traces*.

Essa proposta de agrupamento semi-supervisionado foi experimentada em um cenário de estudo do comportamento de usuários durante a leitura de notícias em um aplicativo. A partir de entrevistas com 30 usuários, um especialista elaborou um conjunto de perfis típicos de leitura, com uma explicação textual sobre o tipo de comportamento de cada perfil. Esse conjunto é considerado um agrupamento orientado por especialista, pois é baseado nos relatos dos próprios usuários sobre seu comportamento. Os autores ressaltam que a utilização do comportamento autorrelatado de usuários é como as empresas normalmente fazem segmentação de usuários, especialmente quando não há dados disponíveis sobre como o produto é realmente usado.

Os dados armazenados em logs provenientes da utilização do aplicativo por 209 usuários (2900 sessões de leitura) foram utilizados em conjunto com os perfis definidos pelo especialista em três abordagens: (i) *K-means* usando as variáveis utilizadas pelo especialista para descrever os perfis de leitura; (ii) *Single Nearest Neighbour* com centroides sendo observações representantes definidas para cada grupo a partir da descrição dos perfis de leitura feita pelo especialista, com cada *trace* sendo incluído no grupo do centroide mais próximo a ele em termos de valores normalizados nas variáveis; e (iii) utilizar esses mesmos centroides como “seeds” iniciais na versão “seeded” do *K-means*.

Com a avaliação dos resultados desse experimento, é demonstrada uma compensação benéfica entre desempenho e maior facilidade de validação dos resultados, dado o domínio de conhecimento de especialistas. Os autores apontam ainda que essa proposta é amplamente aplicável em outros contextos.

3 Agrupamento de *traces* interativo

O objetivo geral delineado neste trabalho foi aplicar estratégias de agrupamento interativo encontradas na literatura da área de mineração de dados no contexto de mineração de processos, executando o que foi chamado de agrupamento de *traces* interativo.

O alcance do objetivo foi elaborado em termos de dois tipos de experimentos: (i) com um log de eventos sintéticos e (ii) com um log de eventos reais provenientes de um sistema de gerenciamento de incidentes, apresentados respectivamente nas seções 3.2 e 3.3. Para cada um desses deles, foram definidos cenários para aplicação de agrupamento. Conforme apresentados na 8, para cada cenário, foram executados três experimentos a fim de possibilitar a análise dos efeitos da aplicação do agrupamento de *traces* interativo:

1. Descoberta do processo realizada diretamente a partir do log de eventos. A análise dos resultados é realizada tanto por analistas de processos quanto analistas de negócio.
2. A partir da representação vetorial do log, é executado o agrupamento de *traces* clássico, com o algoritmo *K-means++* e distância euclidiana, para que a partir dos sublogs referentes aos grupos resultantes seja aplicada a descoberta dos processos respectivos a cada grupo. A análise dos resultados desse experimento é realizada por analistas de dados, além dos analistas de processos e de analistas de negócio.
3. Execução de agrupamento de *traces* interativo, utilizando a abordagem de restrições *must-link/cannot-link* e o algoritmo *Cop-Kmeans*, descritos na seção 2.1.3.

Figura 8 – Esquema de execução dos experimentos

Fonte: Thais Rodrigues Neubauer, 2020

A abordagem de agrupamento de *traces* interativo proposta neste trabalho é composta pela abordagem de agrupamento interativo por restrições *must-link/cannot-*

link utilizando o algoritmo *Cop-Kmeans*, descrita na seção 2.1.3, e pelo mapeamento do conhecimento coletado para restrições, descrita na seção 3.1. Essa combinação foi nomeada de Cop-Kmeans com Mapeamento de Conhecimento em Restrições (CMCR).

3.1 Modelagem adotada para a inclusão do especialista em agrupamento de traces

Como apresentado na figura 8, a inclusão do especialista no processo de agrupamento foi modelada em dois principais passos: (i) visualização dos resultados e (ii) mapeamento do conhecimento coletados em termos de restrições. A visualização de cada grupo foi construída com base na frequência dos valores assumidos pelas características de interesse. Características de interesse são aquelas escolhidas para descrever o perfil de traces de cada grupo. Na figura 9, é apresentado um exemplo desse tipo de visualização para um log no qual o conjunto de atividades é $\{A, B, C, D\}$ e os *traces* são separados em dois grupos. Cada grupo caracterizado pelas atividades dos *traces* que o compõe e o tamanho de cada atividade representa a quantidade de *traces* no grupo que passaram por ela. Assim, podemos interpretar, por exemplo, que existem mais *traces* que passam pela atividade “A” no grupo 2 do que no grupo 1.

Figura 9 – Exemplo de visualização de resultados baseada no valor das características

Fonte: Thais Rodrigues Neubauer, 2020

Devido a forma de caracterização dos grupos apresentada aos especialistas, o conhecimento coletado a respeito dos grupos será em relação às frequências dos valores das características, por exemplo uma indicação de que a atividade “A” (característica *atividade* = “A”) deve ocorrer em somente um dos grupos e a atividade “B” dever ocorrer somente no outro. Um mesmo valor de característica poder ocorrer em vários dados, mas cada restrição *must-link/cannot-link* envolve somente um par de dados. Por isso, é necessário mapear o conhecimento coletado em termos dos valores das características para restrições que possam ser aplicadas a pares de dados.

O primeiro passo desse mapeamento é a seleção dos dados que possuem os valores de característica mencionados nas indicações dos especialistas. Se a indicação for de que os valores de características devem ocorrer somente em um dos grupos, são criadas restrições do tipo *must-link* entre todos os dados selecionados. Se a indicação for de que os valores não podem ocorrer em um mesmo grupo, são criadas restrições do tipo *cannot-link* entre os todos dados selecionados.

Considerando o que foi descrito na seção 2.1.3, o que é necessário prover ao algoritmo *Cop-Kmeans* para aplicação do agrupamento com restrições, além dos dados, são ECVs M e N de restrições *must-link* e as de restrições *cannot-link*, respectivamente.

O pseudo-código 2 apresenta o procedimento para criação de $D_n \sim D_n$: geração de um conjunto M de restrições *must-link* a partir de um conjunto $D_n = \{d_1 \dots d_n\}$ de índices dos dados. A criação de restrições *must-link* a partir de um conjunto D_n deve resultar nas restrições $d_{i1} \sim d_{i2}$ e $d_{i2} \sim d_{i1}$ para cada par (d_{i1}, d_{i2}) de D_n . Para obter as restrições já no formato ECV, motivado e descrito na seção 2.1.3, para cada d_i do conjunto D_n (chave na ECV), é atribuído do conjunto $\{d_1 \dots d_n\} - \{d_i\}$ (valor na ECV).

Algoritmo 2 Procedimento para criação da estrutura de restrições *must-link*

$D_n = \{d_1 \dots d_n\}$: conjunto D_n de índices dos n dados.

M : lista de restrições *must-link*.

Para $i \leftarrow 1$ **até** n **faça**:

$M[i] = \{d_1 \dots d_n\} - \{d_i\}$.

Retorne M

Fonte: Thais Rodrigues Neubauer, 2020

O pseudo-código 3 apresenta o procedimento para criação de $D1_{n1} \bowtie D2_{n2}$: geração de um conjunto N de restrições *cannot-link* a partir dos conjuntos $D1_{n1} = \{d1_1 \dots d1_{n1}\}$ e $D2_{n2} = \{d2_1 \dots d2_{n2}\}$ de índices de dados. A criação de restrições *cannot-link* a partir dos conjuntos $D1_{n1}$ e $D2_{n2}$ deve ser realizada de forma que para cada combinação de elementos dos dois conjuntos $(d1_i, d2_j)$, $d1_i \in D1_{n1}$ e $d2_j \in D2_{n2}$, duas restrições *cannot-link* sejam adicionadas a N : $d1_j \bowtie d2_j$ e $d2_j \bowtie d1_i$. Para obtenção das restrições no formato ECV, esse procedimento é realizado de forma que, para cada combinação $(d1_i, d2_j)$, o valor $d2_j$ é adicionado para a chave $d1_i$ e o valor $d1_i$ é adicionado para a chave $d2_j$.

A quantidade de restrições geradas é de grau exponencial em relação à quantidade de dados selecionados e a aplicação do agrupamento iterativo com restrições necessariamente atende a todas as restrições fornecidas, o que pode ser aumentar o custo do algoritmo. Por isso, os experimentos realizados envolvendo essa abordagem foram realizados com

Algoritmo 3 Procedimento para criação da estrutura de restrições *cannot-link*

$D1_{n1} = \{d1_1 \dots d1_{n1}\}$: conjunto $D1_{n1}$ de índices dos $n1$ dados selecionados.

$D2_{n2} = \{d2_1 \dots d2_{n2}\}$: conjunto $D2_{n2}$ de índices dos $n2$ dados selecionados.

N : lista de restrições *cannot-link*.

Para $i \leftarrow 1$ **até** $n1$ **faça**:

Para $j \leftarrow 1$ **até** $n2$ **faça**:

$N[i] = N[i] + \{d2_j\}$

$N[j] = N[j] + \{d1_i\}$

Retorne N

Fonte: Thais Rodrigues Neubauer, 2020

diferentes proporções dos dados selecionados, a fim de entender se o resultado esperado pode ser alcançado sem a utilização de todas elas.

Para cada proporção, a escolha de quais partes dos dados selecionados seriam utilizados na criação das restrições poderia ser uma escolha aleatória entre eles. Contudo, se houver repetição de dados (linhas iguais na representação vetorial), o resultado obtido com agrupamento sem restrições já pode ser sensivelmente alterado em termos de localização dos protótipos, como apresentado nas imagens da figura 10. Em (b) os protótipos dos grupos verde e laranja foram alterados com relação ao que foi obtido em (a) por conta da existência de dados repetidos, que pode ser observada pela coloração mais forte.

Figura 10 – Exemplo de como a repetição de dados pode alterar o resultado de agrupamento

Fonte: Thais Rodrigues Neubauer, 2020

Quando as repetições são envolvidas nas restrições, os resultados podem ser ainda mais alterados. Na figura 11(a), os dados “1”, “2” e “3” foram selecionados para criação de restrições *must-link* entre eles e, com isso, o dado “1” foi alocado ao grupo verde, causando deslocamento dos protótipos dos grupos laranja e verde. Na figura 11(b), com a seleção dos dados “1”, “3” e “4” para criação das restrições, a localização do protótipo do grupo em verde é ainda mais alterada na direção dos dados “1” e “4” incluídos.

No contexto de logs de eventos usados em mineração de processos, um mesmo processo é instanciado diversas vezes e, dessa forma, diversos casos são gerados. Sendo

Figura 11 – Exemplo da diferença de movimentação do protótipo que pode ocorrer se a restrição envolver dados sem repetição ou com repetição

Fonte: Thais Rodrigues Neubauer, 2020

um *trace* uma sequência de atividades, um mesmo *trace* pode representar as atividades de diferentes casos. Desse ponto de vista, se obtivermos um conjunto de *traces* únicos, os diversos casos relacionados a cada um dos *traces* podem ser entendidos como uma repetição do *trace*. Assim sendo, o envolvimento de diferentes casos de um mesmo *trace* na geração de restrições pode alterar os resultados obtidos na tarefa de agrupamento. Do ponto de vista do espaço vetorial, os *traces* podem ser vistos como fatores que alteram a direção dos protótipos resultantes e as repetições (casos) como a força dessa alteração.

Dessa forma, além de avaliar os resultados com parte das restrições, foram realizados experimentos de agrupamento de *traces* interativo a fim de avaliar como os resultados são alterados mediante diferentes proporções de *traces* e de casos. Para combinação dessas proporções, os dados foram selecionados aleatoriamente e a quantidade mínima de elementos do conjunto resultante de cada seleção foi de um, ou seja, não foram selecionados conjuntos de dados vazios. A frequência de repetições depende da quantidade de características escolhidas para descrever os dados na representação vetorial, e.g. uma representação por atividades terá maior probabilidade de repetições do que uma representação por transições.

Na figura 12, é apresentado um exemplo de log na representação vetorial por atividade. Nele, é possível observar dois *traces*, quatro casos para o primeiro *trace* (linhas de fundo cinza) e dois casos para o segundo (linhas de fundo branco). Nesse exemplo, se o valor de característica indicado pelo especialista for atividade = “ d_1 ” e a proporção de *traces* for de 0.5, somente um dos dois *traces* será selecionado. Ainda, supondo que o primeiro *trace* seja selecionado e que a proporção de casos seja de 0.75, então três entre as quatro primeiras linhas serão selecionadas para criação de restrições.

Figura 12 – Exemplo de log mapeado para o espaço vetorial com perfil em atividades para exemplificar a diferença entre dados únicos (*traces*) e repetições (casos)

Case	Atividades							
	a_1	b_1	c_1	d_1	e_1	f_1	g_1	...
1	1	0	1	1	0	1	0	0
2	1	0	1	1	0	1	0	0
3	1	0	1	1	0	1	0	0
4	1	0	1	1	0	1	0	0
5	0	1	0	1	1	0	1	0
6	0	1	0	1	1	0	1	0

Fonte: Thais Rodrigues Neubauer, 2020

3.2 Experimentos com logs de eventos sintéticos

Nesta seção são apresentados os logs de eventos sintéticos utilizados, os três cenários de experimentação elaborados e os resultados obtidos a partir de execuções de agrupamento de *traces* e de agrupamento de *traces* interativo. A intenção da utilização desse tipo log foi a criação de um ambiente de teste independente de contexto, permitindo abstrair questões semânticas relacionadas ao processo subjacente ao log.

3.2.1 Os logs de eventos sintéticos

O log de eventos sintéticos explorado foi disponibilizado pelo *4TU.Centre for Research Data*, no repositório *4TU.ResearchData*¹, mais especificamente no subrepositório *Benchmarking logs to test scalability of process discovery algorithms*², disponibilizado por Will. M. P. Van der Aalst, em 2017 (AALST, 2017). Esse subrepositório consiste em um conjunto de 38 logs de eventos que visam apoiar a avaliação do desempenho de algoritmos de descoberta de processos. Todos os logs foram gerados a partir do mesmo modelo de processo de base, apresentado na figura 13(a) e para gerar processos mais complexos, o modelo é replicado m vezes, usando diferentes métodos de composição. Para cada um desses processos complexos, um número predefinido de n casos foi gerado.

A partir da realização dessas composições, o modelo de processo usado na geração do log passa a representar o comportamento resumidamente ilustrado na figura 13(b). Segundo esse esquema, o mesmo processo base é repetido dentro de uma estrutura do tipo XOR, de forma que em cada *trace* gerado, apenas uma réplica de processo fará parte

¹ Neste repositório, há uma coleção específica chamada *IEEE TF em Process Mining - Event Logs* com logs de eventos sintéticos e log de eventos do mundo real. No conjunto de logs de eventos sintéticos, diferentes tipos de logs foram organizados em 16 subrepositórios (links). (http://data.4tu.nl/repository/collection:event_logs_synthetic)

² (<https://data.4tu.nl/repository/uuid:1cc41f8a-3557-499a-8b34-880c1251bd6e>)

da execução do processo. No log, cada um dos processos base é diferenciado por uma numeração: todas as atividades da n -ésima réplica do processo base são identificadas pelos rótulos $\{an, bn, \dots, gn\}$.

Figura 13 – Modelo de processo usado no exemplo de agrupamento de *traces*. Em (a), o processo base utilizado em todos os logs do subrepositório BM; os locais marcados com x identificam estados nos quais uma atividade excepcional x pode acontecer. Em (b), a esquematização gráfica do processo mais complexo criado usando a composição *de escolha* - Figuras extraídas do arquivo *ReadMe* disponibilizado juntamente com os logs do subrepositório BM (AALST, 2017)

Fonte: Aalst (2017)

No log de eventos explorado, os atributos disponíveis para cada evento são:

- *Case* (caso): identificação do *trace*.
- *Activity* (atividade): identificação de cada atividade e o número da réplica do processo de base relacionado. Exemplo: a_2 - atividade “a” da réplica “2” do processo base).
- *Timestamp*: indicação temporal de quando ocorreu o evento, no formato texto, seguindo o padrão *ano-mes-dia hora:minuto:segundo*.

Duas versões desses logs de eventos sintéticos foram utilizadas, ambas geradas com o método de composição ilustrado na figura 13: a primeira com $m = 100$ réplicas do processo base e $n = 10000$ casos, com ruído (atividade “x”), e a segunda com $m = 5$ réplicas, $n = 493$ casos, sem ruído.

A ferramenta Disco³ foi utilizada para obtenção de um entendimento mais detalhado do processo subjacente ao log estudado. Na figura 14, é apresentado o modelo gerado na ferramenta a partir do log com $m = 100$. Na parte (a) da figura, na qual um recorte do processo completo é apresentado, é demonstrada a alta complexidade do processo completo.

³ A ferramenta Disco oferece uma série de opções e funcionalidades para gerenciar a análise de logs de eventos. Cada tipo de visualização mostra um ponto de vista para análise de processo. Basicamente, os modelos gerados pela ferramenta Disco são um mapa de processo construído com base na sequência e no tempo das atividades presentes nos dados do registro de eventos. (ROZINAT, 2018)

Na parte (b) da mesma figura, é apresentada a visualização simplificada das atividades com o nível de detalhes de caminhos determinado em 0% e o nível de atividades em 2,5%⁴. É possível observar que cada réplica do processo base é independente das outras, pois não há intersecção entre os conjuntos de atividades de cada uma. Nesse contexto, esse parece um ambiente propício para que a tarefa de agrupamento de *traces* consiga identificar os perfis de cada da réplica do processo base, pois os *traces* gerados por uma delas devem ser mais similares entre si do que os *traces* gerados passando por outras réplicas. Ainda, na parte (c) da mesma figura, a visão mais simplificada possível é apresentada (nível de atividades definido como 0%), com o nível de detalhes de caminho determinado para 100%.

Figura 14 – Modelos de processos gerados com a ferramenta DISCO, na qual atividades são representadas por caixas, o fluxo do processo é representado por setas e as setas tracejadas indicam pontos de início e fim do fluxo do processo. Em (a), é apresentado um recorte da visualização do modelo de processo do log com $m = 100$; em (b), o modelo de processo simplificado do mesmo log, com visualização de 2,5% das atividades e 0% dos caminhos; e em (c), o modelo de processo simplificado do mesmo log, com visualização de 0% das atividades e 100% dos caminhos

Fonte: Thais Rodrigues Neubauer, 2020

Para cada um dos logs, foi realizado o mapeamento dos *traces* para vetores de características baseados em perfis por atividade e de por transição, considerando os

⁴ Um nível de detalhe em 0% significa que apenas as atividades (ou caminhos) de máxima frequência serão mostrados na visualização. Em um nível de detalhe de 100% todas as atividades e caminhos são mostrados na visualização.

modelos de contagem com representação binária e *tfidf*⁵, resultando, portanto, em quatro representações para cada log, apresentado no quadro 1.

Quadro 1 – Variações de representação experimentadas para o log sintético

Quesito	Variações
Perfis	{por atividade, por transição}
Representação no modelo de contagem	{binária, <i>tfidf</i> }
Total de representações	6

Fonte: Thais Rodrigues Neubauer, 2020

3.2.2 Log de eventos com $m = 100$ réplicas do processo base

Para esse log de eventos sintéticos com $m = 100$ réplicas do processo base e $n = 10000$ casos, dois cenários de experimentos foram elaborados com o objetivo de verificar as capacidades da abordagem CMCR em comparação com o agrupamento de *traces* clássico. Esses experimentos, identificados como *cenário 1* e *cenário 2*, foram elucidados a partir de um experimento preliminar, identificado como *cenário 0*, executado com o objetivo de observar as capacidades do agrupamento de *traces* clássico.

Cenário 0: exploração do agrupamento clássico

O objetivo geral neste cenário de experimento foi explorar o resultado do agrupamento de *traces* clássico a fim de verificar as capacidades do algoritmo *K-means++* no contexto desse log de eventos sintéticos. Para tanto, neste cenário, foi utilizado o log de eventos sintéticos com $m = 100$ réplicas do processo base e $n = 10000$ casos e foi executado um único experimento: a aplicação de um agrupamento de *traces* clássico, com quantidade de grupos definida em $k = 100$. O resultado esperado era que todos os *traces* que passam por uma mesma réplica do processo base fossem alocados ao mesmo grupo e que somente *traces* de uma mesma réplica fossem alocados a um mesmo grupo. Os resultados foram analisados por meio da caracterização por nuvem de atividades dos grupos obtidos, como apresentado na figura 15.

Com a análise desses resultados, foi possível observar que:

⁵ Para esse log de eventos sintéticos, as representações por contagem binária e *tf* resultam no mesmo vetor, pois cada atividade acontece, no máximo uma vez em cada caso. Por isso, a contagem *tf* não será analisada nos experimentos com esse log.

Figura 15 – Exemplos das nuvens de atividades geradas para três dos grupos resultantes da aplicação do *k-means++*, com $k = 100$, distância euclidiana e contagem binária no log de eventos sintéticos mapeado para o perfil por atividade no log de eventos sintéticos

Fonte: Thais Rodrigues Neubauer, 2020

- Para a maior parte das réplicas, todas suas atividades foram alocadas a um único grupo, como no caso da réplica 47 no “Grupo 1” apresentado na figura 15(a).
- Para quatro das réplicas, o algoritmo dividiu suas atividades em dois grupos: um com *traces* que passaram pelo subconjunto de atividades $Sub_A = \{a, d, c, f\}$ da réplica e outro com *traces* que passaram pela subconjunto $Sub_B = \{b, d, e, g\}$ da réplica. Cada *trace* contém somente um dos subconjuntos por conta do *XOR-split* do processo base. Esse caso é exemplificado para a réplica 30 no “Grupo 87” e no “Grupo 4” apresentados, respectivamente, na figura 15 (b) e (c).
- Por conta dessa divisão das atividades de uma mesma réplica em dois grupos para algumas das réplicas, um dos grupos (“Grupo 99”) englobou as atividades de cinco réplicas (10, 31, 37, 48, 51).

A partir dessa análise, foram elaborados os cenários 1 e 2 para verificar a capacidade dos algoritmos de agrupamento em separar os *traces* com base no *XOR-split* do processo base, ou seja, separar os *traces* que passam por Sub_A dos que passam por Sub_B .

Cenário 1: separação do *XOR-split*

O objetivo geral dos experimentos deste cenário é avaliar a capacidade das abordagens de agrupamento em separar *traces* que passam por cada um dos subconjuntos de atividades decorrentes do *XOR-split* do processo base. Para tanto, a partir do log de eventos sintéticos com $m = 100$ réplicas e $n = 10000$ casos, foram realizados os experimentos com agrupamento clássico e com a abordagem CMCR e $k = 2$ grupos. Para atender à expectativa, em cada experimento, um dos grupos resultantes deveria conter somente

traces que passaram pelo subconjunto de atividades $Sub_A = \{a, d, c, f\}$ e o outro, somente *traces* que passaram pelo subconjunto $Sub_B = \{b, d, e, g\}$.

Para aplicar a abordagem CMCR, o mapeamento descrito na seção 3.1 foi realizado com a execução dos seguintes passos:

1. Seleção do conjunto de índices de *traces* que passam por cada subconjunto de atividades: A para os *traces* que passam por $Sub_A = \{a, d, c, f\}$ e B para os que passam por $Sub_B = \{b, d, e, g\}$.
2. Seleção, a partir dos conjuntos A e B , dos conjuntos A_{select} e B_{select} , respectivamente, envolvidos na criação das restrições para cada proporção de *traces* e casos explorada.
3. Criação de restrições $A_{select} \sim A_{select}$, $B_{select} \sim B_{select}$, $A_{select} \bowtie B_{select}$.

Ainda, foi realizado um experimento de descoberta de modelo de processo a partir do log completo. Com o resultado desse experimento, foi possível comparar a qualidade de modelo de processo com os resultados dos experimentos com agrupamento.

A avaliação dos resultados das execuções de CMCR sob o ponto de vista de mineração de dados, considerando o índice Silhouette, é apresentada na figura 16. Cada ponto nos gráficos representa o valor do Silhouette do agrupamento resultantes nas diferentes execuções. Cada execução foi realizada com uma proporção de *traces*, indicada no eixo x , e uma proporção de casos, indicada por sua cor. A linha azul foi incluída para facilitar a comparação dos valores de cada execução da abordagem CMCR com o resultado obtido no experimento com agrupamento clássico.

Figura 16 – Índice Silhouette para os agrupamentos do log sintético no cenário 1

Fonte: Thais Rodrigues Neubauer, 2020

As visualizações revelam que melhores índices de qualidade, inclusive melhor que o experimento com agrupamento clássico, foram alcançados com a perfil por transição e representação binária, mesmo para os casos com baixa proporção de *traces* e casos envolvidos nas restrições. Para as outras combinações de perfil e representação, foram alcançados índices similares (ligeiramente melhores para alguns casos de perfil por atividade e representação binária, com 100% de *traces* e casos envolvidos nas restrições, e ligeiramente piores para os demais). Este comportamento mostra que do ponto de vista do índice Silhouette, a organização dos grupos sofre uma perturbação quando da aplicação das restrições, pois o espaço vetorial e a função de similaridade continuam os mesmos e os centróides são reposicionados por conta das realocações dos dados aos grupos.

Para analisar os resultados obtidos em termos de atendimento à expectativa delineada no cenário, foram contabilizados os *acertos*. Suponha uma execução na qual tenham sido obtidas em cada grupo as quantidades de casos de cada subconjunto de atividades apresentadas na tabela 6.

Tabela 6 – Exemplo de distribuição dos casos nos grupos

	Grupo 1	Grupo 2
Casos que passam pelo Sub_A	4900	100
Casos que passam pelo Sub_B	2000	3000

Nessa situação, se fossem aplicadas nesse resultado as restrições geradas com todos os *traces* e casos dos subconjuntos, diversas realocações de dados seriam necessárias para não violar restrições. Por exemplo, os 2000 casos que passam pelo subconjunto de atividades Sub_B no primeiro grupo serem realocados para o segundo grupo e os 100 casos que passam pelo subconjunto de atividades Sub_A no segundo grupo serem realocados para o primeiro. Foram contabilizados como acertos a combinação para a qual a menor quantidade de realocações possíveis precisariam ser realizadas para não violar restrições e, conseqüentemente, atender à expectativa delineada neste cenário. Assim, opta-se pela primeira das duas possibilidades de contagem de acertos: (i) $4900 + 3000 = 7900$, considerando os casos do subconjunto Sub_A do primeiro grupo e os casos do subconjunto Sub_B do segundo grupo; (ii) $2000 + 100 = 2100$, considerando os casos do subconjunto Sub_A do primeiro grupo e os 100 casos do subconjunto Sub_B do segundo grupo. Ainda, foi realizada uma ponderação em relação a quantidade de acertos totais que seriam alcançados quando da execução com proporção total de *traces* e casos, que, neste caso, corresponde a

10000. Dessa forma, os 7900 acertos do exemplo correspondem a 0.79 do total de dados envolvidos na expectativa delineada neste cenário.

A figura 17 mostra os resultados das execuções deste cenário em termos de quantidade de acertos para cada uma das combinações de proporção de *traces* e casos explorada.

Figura 17 – Resultados em termos de acertos dos agrupamentos do log sintético no cenário 1. Em (a), as linhas contínuas apresentam a quantidade total de acertos e as pontilhadas, as respectivas quantidades de dados envolvidos nas restrições utilizadas no experimento.

(a) Acertos e quantidade de dados envolvidos

(b) Aumento de acertos aprendidos por restrição adicionada

Fonte: Thais Rodrigues Neubauer, 2020

Nessas visualizações, a ideia principal é comparar as formas de representação a fim de identificar quais as que mais colaboram para alcançar taxas maiores de acerto com menores quantidades de restrições, isto é, quais mais auxiliam o processo de aprendizado do algoritmo⁶. Para identificar essas formas de representação que mais colaboram para o

⁶ a partir da inserção de um dado nas restrições, ele será obrigatoriamente alocado ao grupo que atende à expectativa delineada e, portanto, certamente maior será a taxa total de acerto, mas não necessariamente o aumento nos acertos foi proveniente do aprendizado do algoritmo.

aprendizado em relação ao atendimento da expectativa, complementarmente à quantidade total de acertos, as linhas tracejadas apresentam as respectivas quantidades de dados envolvidos nas restrições utilizadas na execução.

Além disso, na figura 17(b) é apresentado o ganho em termos de acertos aprendidos por dado adicionado às restrições em relação ao experimento imediatamente anterior (que potencialmente⁷ tem menor quantidade de restrições por conta de envolvimento de menos casos⁸). Sendo e uma execução, n o identificador da execução na lista ordenada de execuções, α a quantidade absoluta de acertos e β a quantidade de dados envolvidos nas restrições, para estimar a quantidade de acertos aprendidos ω foi realizado o seguinte cálculo: $\omega_{en} = (\alpha_{en} - \beta_{en})$. Ainda, para estimar a quantidade θ de acertos aprendidos por dado adicionado às restrições foi realizado o seguinte cálculo: $\theta_{en} = (\omega_{en} - \omega_{en-1}) \div (\beta_{en} - \beta_{en-1})$.

Analizando esse cenário, é possível observar que a representação por perfil de atividade leva a maiores taxas de acerto desde as menores proporções de *traces* e casos. Porém, observando a quantidade de restrições, percebemos que para o perfil por transição, a quantidade de restrições é menor. Observando a figura 17(b), podemos notar que os perfis e representações não parecem influenciar no aprendizado e os aumentos observados em acertos são resultantes somente da aplicação de uma maior quantidade de restrições.

Ainda, para analisar a diferença nos resultados alcançados por conta de decisões tomadas ao longo da execução do algoritmo, como por exemplo a ordem de passagem dos dados para atribuição aos grupos, algumas execuções foram repetidas por 10 vezes e o resultado é apresentado na figura 18 por meio da média e desvio padrão dos acertos nas repetições.

Na figura 19, é apresentada a análise dos resultados obtidos nos experimentos em termos das medidas de qualidade do modelo de processo, definidas na seção 2.2. Nela, a linha azul representa o resultado obtido com o agrupamento clássico (*K-means++*) e a linha cinza, o obtido a partir do log completo. Os valores alcançados pelas execuções da abordagem CMCR não apresentaram diferenças significativas entre si, com exceção os seguintes pontos levantados quando da comparação com o resultado alcançado pelo

⁷ Não necessariamente uma menor proporção de casos significa menor quantidade de restrições, pois se a quantidade relativa de casos na representação em questão for pequena, um aumento de proporção de casos pode não levar ao aumento da quantidade absoluta de dados envolvidos nas restrições.

⁸ Quando a proporção de casos é a menor (0.01), a quantidade de acertos anterior assumida é a do experimento com proporção de *traces* imediatamente menor, com 100% de casos

Figura 18 – Variação da quantidade de acertos dos agrupamentos do log sintético no cenário 1. A área ao redor de cada linha representa a variação na quantidade de acertos observada na execução das 10 repetições

experimento de descoberta com o log completo (linhas cinzas) e com o agrupamento clássico:

- *Fitness*: com perfis por atividade, houve melhora da medida para todos os resultados de CMCr em comparação tanto com os resultados do experimento com o log completo (linha cinza) quanto com os resultados do experimento com o agrupamento clássico (linha azul). Porém, com perfis por transição, os valores só foram iguais ou melhores aos obtidos com o agrupamento clássico e representação tfidf, apesar de nos casos nos quais é pior, ainda ser superior ao resultado do experimento com log completo.
- *Precisão*: podemos observar um padrão de comportamento para perfis por atividade e representação binária, pois todas as execuções resultaram em valores melhores do que o agrupamento clássico. Ainda, na maior parte das execuções, os valores obtidos foram melhores também do que o resultados com o log completo. Com perfis por transição e representação tfidf, todas as execuções resultaram em valores mais baixos do que os obtidos no experimento com agrupamento clássico, mas bastante semelhantes e até um pouco superiores em alguns casos, do que o com o log completo. Para duas outras combinações de perfis e representação, o comportamento teve variações significativas a depender da quantidade de *traces* e casos aplicados e os resultados do agrupamento clássico e do log completo são bastante similares.
- *Simplicidade*: os valores para o agrupamento clássico e para o log completo são bastante semelhantes e todos os resultados do experimento com CMCr tiveram valores iguais ou melhores do que os de agrupamento clássico e log completo.

- Generalização: os valores para o agrupamento clássico e para o log completo são bastante semelhantes e todos os resultados do experimento com CMCR tiveram valores mais baixos.

Figura 19 – Medidas de qualidade de modelo de processos para os resultados do cenário 1

Fonte: Thais Rodrigues Neubauer, 2020

Cenário 2: separação do *XOR-split* e *traces* com ruído

Assim como para os experimentos do cenário 1, para este cenário, o objetivo geral dos experimentos envolveu a verificação das capacidades dos agrupamentos em separar os *traces* que passam por cada subconjunto de atividades decorrentes do *XOR-split* do processo base. Contudo, além dessa separação, também faz parte do objetivo deste cenário a separação dos *traces* ruidosos (com a presença da atividade “x”) em um grupo diferente dos demais. Foi utilizado o log de eventos sintético com $m = 100$ réplicas do processo base e $n = 10000$ casos e foram realizados os experimentos com algoritmo K-means++ e com a abordagem CMCR. A quantidade de grupos foi definida em $k = 3$ e, para atender à expectativa, os *traces* de cada subconjunto de atividades devem ser alocados a um grupo distinto e o terceiro grupo deve ser composto somente por todos os *traces* com ruído.

O mapeamento para aplicação da abordagem CMCR também se deu de forma semelhante ao que foi realizado para o cenário 1 e é descrito pelos seguintes passos:

1. Seleção do conjunto de índices de *traces* que passam por cada subconjunto de atividades e não tem ruído: conjunto A de índices dos *traces* que passam por Sub_A e não passam por nenhuma atividade “x” e conjunto B de índices dos *traces* que passam por Sub_B e não passam por nenhuma atividade “x”.
2. Seleção do conjunto de índices de *traces* ruidosos: conjunto X de índices dos *traces* que passam pela atividade “x”;
3. Seleção, a partir de A , B e X , dos conjuntos de índices A_{select} , B_{select} e X_{select} , respectivamente, envolvidos na criação das restrições para cada proporção de *traces* e *casos* explorada.
4. Criação de restrições $A_{select} \sim A_{select}$, $B_{select} \sim B_{select}$, $X_{select} \sim X_{select}$, $A_{select} \bowtie B_{select}$, $A_{select} \bowtie X_{select}$, $B_{select} \bowtie X_{select}$.

O experimento de descoberta de modelo de processo a partir do log completa foi realizado neste cenário a fim de possibilitar a comparação da qualidade de modelo de processo com os resultados dos experimentos com agrupamentos.

A maior parte dos resultados foram bastante semelhantes aos valores encontrados para o cenário 1 e são apresentados na íntegra no apêndice B. Os resultados que não são semelhantes aos obtidos no cenário 1 são apresentados e discutidos a seguir.

Na figura 20, são apresentados os valores alcançados para o índice Silhouette. Todas as execuções de CMCR alcançaram valores mais baixos do que os resultados obtidos com o agrupamento clássico, corroborando com a questão da perturbação que a aplicação de restrições pode causar, comentada na avaliação do índice Silhouette no cenário 1.

As métricas de qualidade de modelo processos *fitness* e precisão são apresentadas, respectivamente, nos gráficos (a) e (b) da figura 21. É possível perceber que o perfil por transição com representação tfidf é que tem valores mais baixos de *fitness* do que o experimento com agrupamento clássico. Além disso, a precisão tem sempre valores mais baixos do que os resultados alcançados no experimento com agrupamento clássico, mas é sempre semelhante ou superior ao alcançado com o log completo.

Figura 20 – Índice Silhouette para os agrupamentos do log sintético no cenário 2

Fonte: Thais Rodrigues Neubauer, 2020

Figura 21 – Medidas de qualidade de modelo processos para os resultados de agrupamento do log sintético no cenário 2

Fonte: Thais Rodrigues Neubauer, 2020

3.2.3 Log de eventos com $m = 5$ réplicas do processo base

Nos cenários 1 e 2, apresentados na seção 3.2.2, com o log de eventos com $m = 100$ réplicas do processo base, quando as restrições são aplicadas a todos os dados dos conjuntos de índices selecionados (100% de *traces* e 100% dos casos de cada trace), o resultado é totalmente determinado e o problema pode até mesmo ser modelado como uma tarefa de classificação se as restrições forem mapeadas para rótulos. Essa determinação do resultado é decorrente do fato de que somente um resultado de atribuições dos dados aos grupos atende a todas as restrições, visto que elas envolvem todos os dados do conjunto. O cenário 3, descrito a seguir, foi elaborado a fim de analisar as capacidades da abordagem CMCR

se o resultado de agrupamento com a abordagem CMCR não é totalmente determinado quando todos os *traces* e *casos* são selecionados para geração das restrições.

Cenário 3: separação de réplicas

O objetivo geral neste cenário, utilizando o log de eventos sintéticos com $m = 5$ réplicas e $n = 493$ casos, foi analisar as capacidades das abordagens de agrupamento clássico e CMCR em separar os *traces* em $k = 3$ grupos de forma que: (i) os que passaram pelas réplicas 1 e 2 sejam atribuídos ao mesmo grupo; e (ii) os que passaram pelas réplicas 3 e 4 sejam atribuídos a grupos distintos. Assim, para atender à expectativa, um dos grupos tem que conter todos os *traces* que passaram pelas réplicas 1 e 2 e nenhum dos grupos podem conter *traces* que passaram pela réplica 3 e pela réplica 4.

Para isso, o mapeamento para restrições foi realizado pelos passos:

1. Seleção do conjunto de índices de *traces* que passam pelas réplicas envolvidas na expectativa: $REP1$ para os índices dos *traces* que passam pela réplica 1, $REP2$ para os índices dos *traces* que passam pela réplica 2, $REP3$ para os índices dos *traces* que passam pela réplica 3 e $REP4$ para os índices dos *traces* que passam pela réplica 4.
2. Seleção, a partir dos conjuntos $REP1$, $REP2$, $REP3$ e $REP4$, dos conjuntos $REP1_{select}$, $REP2_{select}$, $REP3_{select}$ e $REP4_{select}$, respectivamente, envolvidos na criação das restrições para cada proporção de *traces* e casos explorada.
3. Criação de restrições $REP1_{select} \sim REP2_{select}$ e $REP3_{select} \bowtie REP4_{select}$.

Ainda, foi realizado o experimento de descoberta de modelo de processo a partir do log completo. Com esse resultado, foi possível comparar a qualidade desse modelo com os modelos descobertos a partir dos grupos obtidos nos experimentos com agrupamentos. As análises para estes experimentos foram analisados seguindo as mesmas estratégias de visualização e raciocínios das apresentadas nos cenários 1 e 2.

Os índices Silhouettes apresentados na figura 22 revelam que do ponto de vista de mineração de dados, os resultados obtidos com o agrupamento clássico atingiram um índice de separação melhor dos que todos os resultados obtidos com a abordagem CMCR para todas as execuções com perfis por transição. Essa observação corrobora, novamente, com o que foi discutido sobre a perturbação causada pela aplicação das restrições no cenário 1.

Figura 22 – Índice Silhouette dos agrupamentos do log sintético no cenário 3

Fonte: Thais Rodrigues Neubauer, 2020

Na figura 23, são apresentadas as visualizações dos acertos, ou seja, de quantos dos dados aos quais as restrições se aplicariam estão obdecendo-as, e sua análise revela uma variação muito maior nos resultados quando comparado aos resultados apresentados nos cenários 1 e 2. Ainda assim, a estimativa do aumento de acertos aprendidos por restrição adicionada, como pode ser observado na figura 23 (b), também não mostrou relação clara entre os acertos aprendidos e a adição de dados nas restrições. Isso reforça que o objetivo do especialista só é garantidamente alcançado quando todos os dados envolvidos nesse objetivo são utilizados aplicações às restrições. Porém, é possível observar um indício de que há maior estabilidade na quantidade de acertos por restrição adicionada com os perfis por transição do que com os perfis por atividade, que apresenta até mesmo uma indicação de aprendizado quando da utilização da representação binária, mas apresenta o efeito contrário para alguns casos com a representação tfidf.

A variação da quantidade de acertos entre as repetições das execuções com mesma combinação de perfil, representação e quantidade de *traces* e de casos, apresentada na figura 24, mostra que quanto menor a quantidade de casos, maior é a variação. Assim, as indicação de aprendizado comentadas na análise da figura 23 precisam ser melhor verificadas, preferencialmente isolando esse tipo de fator de aleatoriedade.

As visualizações das métricas de qualidade de modelo de processos, *fitness*, precisão, simplicidade e generalização, definidas na seção 2.2, são apresentadas na figura 25. Os valores alcançados pelas execuções com diferentes proporções de *traces* e casos do experimento com a abordagem CMCR não apresentaram diferenças significativas entre si. Comparando

Figura 23 – Resultados em termos de acertos nos agrupamentos com log sintético no cenário 3

(a) Acertos e quantidade de dados envolvidos

(b) Aumento de acertos aprendidos por restrição adicionada

Fonte: Thaís Rodrigues Neubauer, 2020

Figura 24 – Variação da quantidade de acertos dos agrupamentos do log sintético no cenário 3, em 10 execuções

esses resultados com os alcançados com log completo (linhas cinzas) e com o algoritmo clássico, os seguintes pontos são destacados:

- *Fitness*: os valores obtidos em todas as execuções do experimento com CMCR foram superiores aos obtidos com o log completo. Quando comparados com agrupamento clássico, os perfis por atividade resultaram em valores semelhantes enquanto os perfis por transição resultaram em valores um pouco menores;
- *Precisão*: os valores obtidos em todas as execuções do experimento com CMCR também foram superiores aos obtidos com o log completo. Porém, quando comparados com agrupamento clássico, os perfis por atividade resultaram em valores maiores enquanto os perfis por transição resultaram em valores menores;
- *Simplicidade*: o resultado de todas as execuções do experimento com CMCR foram superiores aos valores obtidos com os resultados com agrupamento clássico e bastante semelhantes aos obtidos com o log completo;
- *Generalização*: os valores para o agrupamento clássico e para o log completo também são próximos a todos os resultados do experimento com a abordagem CMCR.

3.2.4 Considerações finais sobre os experimentos com logs de eventos sintéticos

Comparando os resultados obtidos nos experimentos de cada um dos cenários do ponto de vista de mineração de dados, os valores obtidos para o índice Silhouette revelam que, em geral, há piora da qualidade quando da aplicação de restrições, ou seja, comparando os valores obtidos na abordagem iterativa (CMCR) com os obtidos na abordagem clássica (*K-means++*). Associa-se essa piora à perturbação supracitada causada na localização dos centroides, visto que o espaço vetorial e a função de similaridade não são alteradas e os centroides são reposicionados. Contudo, vale ressaltar que, considerando o intervalo possível de valor de Silhouette ($[-1, 1]$), as piores observadas não são significativas, da ordem de 0,01 em média.

Em relação ao alcance da expectativa de especialista delineada em cada cenário, avaliado em termos de “acertos” e “acertos aprendidos”, não foi observado combinação alguma de representação, espaço menor do que 1.0 e força menor do que 1.0 que levasse consistentemente ao alcance do resultado esperado. A partir disso, podemos afirmar que

Figura 25 – Medidas de qualidade de modelo de processos para os resultados de agrupamento do log sintético no cenário 3

Fonte: Thais Rodrigues Neubauer, 2020.

a utilização de parte dos dados envolvidos em uma restrição não garante o alcance da expectativa delineada com a restrição.

Quanto a análise dos resultados do ponto de vista de mineração de processos, é possível perceber que, há diferenciações significativas entre eles a depender da natureza da expectativa inserida no agrupamento por meio das restrições. A fim de estruturar a análise desses resultados de medidas de mineração de processos, foram estabelecidos os *baselines* e as comparações apresentadas na tabela 7, na qual são também explicitadas as motivações para as respectivas comparações.

Tabela 7 – Comparações dos valores obtidos nas métricas de mineração de processos

<i>Baseline</i>	<i>Resultado</i>	<i>Motivação da comparação</i>
Log completo	<i>K-means++</i>	Avaliar se o agrupamento clássico apresenta avaliações melhores do que as obtidas com o log completo
<i>K-means++</i>	CMCR	Avaliar se há variação entre os valores obtidos com os resultados dessas abordagens
Log completo	CMCR	Avaliar se a abordagem proposta alcança melhores valores do que os obtidos com o log completo

Fonte: Thais Rodrigues Neubauer, 2020

A figura 26 apresenta as comparações feitas conforme explicitado na tabela 7, sendo o agrupamento clássico a aplicação do K-means++ e o agrupamento interativo a aplicação da abordagem CMCR. Observando os valores apresentados, é possível notar que:

- Quando comparados os valores obtidos a partir dos resultados do agrupamento clássico com o *baseline* do log completo, só houve alteração significativa nos valores obtidos para a métrica de precisão nos cenários 2 e 3.
- Quando comparados os valores obtidos a partir dos resultados da aplicação da abordagem CMCR proposta tanto com o *baseline* do agrupamento clássico quanto com o *baseline* do log completo, é possível notar que há piora significativa na métrica de generalização em todos os cenários, principalmente no cenário 3.

Figura 26 – Comparação dos resultados obtidos nos cenários de experimentação com o log sintético. Valores menores do que o *baseline* (piora da métrica) são sinalizados em tons de vermelho e os maiores (melhora da métrica), em tons de verde.

Log completo - Agrupamento clássico

representação	Cenário 1				Cenário 2				Cenário 3			
	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização
atividade binary	0.0657	-0.0471	0.0162	0.0252	-0.0861	0.0742	-0.0590	-0.0229	-0.0061	0.0001	-0.0334	-0.0004
atividade tfidf	-0.0617	-0.0563	-0.0489	0.0205	0.0093	0.0880	-0.0218	0.0258	0.0029	0.0025	-0.0334	-0.0007
transição binary	0.1157	0.0554	-0.0135	0.0125	-0.0305	0.3701	-0.1073	-0.0350	0.0699	0.3512	-0.1046	-0.0060
transição tfidf	0.1152	0.0540	-0.0142	0.0119	0.1038	0.3508	-0.0695	-0.0587	0.0700	0.3518	-0.1046	-0.0067

Agrupamento clássico - Agrupamento interativo

representação	Cenário 1				Cenário 2				Cenário 3			
	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização
atividade binary	0.0049	0.0721	-0.0043	-0.1572	0.1775	0.0086	0.0641	-0.2391	0.0283	0.1678	0.0476	-0.7108
atividade tfidf	0.1331	0.0598	0.0643	-0.1016	0.0808	-0.0488	0.0338	-0.2144	0.0153	0.1508	0.0489	-0.7107
transição binary	-0.0412	0.0162	0.0181	-0.2156	0.1228	-0.2816	0.1114	-0.2222	-0.0384	-0.1443	0.1147	-0.7149
transição tfidf	-0.0448	-0.0398	0.0270	-0.1182	-0.0129	-0.3102	0.0805	-0.1216	-0.0449	-0.1624	0.1165	-0.7104

Log completo - Agrupamento interativo

representação	Cenário 1				Cenário 2				Cenário 3			
	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização	fitness	precisão	simplicidade	generalização
atividade binary	0.0050	0.0731	-0.0043	-0.1593	0.1798	0.0087	0.0649	-0.2422	0.0287	0.1706	0.0484	-0.7225
atividade tfidf	0.1348	0.0606	0.0651	-0.1029	0.0818	-0.0494	0.0342	-0.2172	0.0153	0.1508	0.0489	-0.7223
transição binary	-0.0417	0.0164	0.0183	-0.2184	0.1244	-0.2853	0.1129	-0.2250	-0.0384	-0.1443	0.1147	-0.7257
transição tfidf	-0.0454	-0.0403	0.0273	-0.1197	-0.0131	-0.3142	0.0815	-0.1231	-0.0449	-0.1624	0.1165	-0.7212

Escala de cores	Maior piora	-1.0
	Maior melhora	1.0

Fonte: Thais Rodrigues Neubauer, 2020.

3.3 Experimentos com log de eventos de um processo de gerenciamento de incidentes

Esta seção apresenta os experimentos realizados com um log de eventos proveniente de um processo de negócio do mundo real. Em Amaral, Fantinato e Peres (2018a), estudo

que utilizou esse mesmo log, são apresentados métodos automáticos de seleção de atributos para aprimoramento da tarefa de predição de tempo de resolução de incidente. Os métodos apresentados indicaram dois conjuntos diferentes de atributos para determinar (prever) o tempo necessário para resolução de um incidente: o primeiro algoritmo indicou o conjunto de atributos $atrib1 = \{caller_id, assigned_to\}$; o segundo, o conjunto $atrib2 = \{incident_state, location\}$. Os resultados de predição utilizando esses conjuntos foram comparados entre si e também comparados aos resultados de predição obtidos utilizando um conjunto de atributos considerado como indicação de especialista, já que está baseado no padrão ITIL – *Information Technology Infrastructure Library*. O conjunto foi tratado pelos autores como sendo a escolha mais provável de um especialista na área de gerenciamento de incidentes e é formado por três atributos: $atrib_espec = \{incident_state, category, priority\}$.

Como indicado por Folino, Guarascio e Pontieri (2014), a aplicação de agrupamento como etapa de pré-processamento pode auxiliar na obtenção de preditores de séries temporais mais precisos. A ideia é que, para cada grupo resultante, haja um preditor distinto que será mais preciso do que o preditor construído com base em todos os dados. Por isso, pensando na tarefa de predição de tempo de resolução de incidentes já aprimorada em Amaral, Fantinato e Peres (2018a), os experimentos desta seção tem como objetivo geral avaliar se agrupamento de *traces* interativo pode ser mais efetivo do que o agrupamento clássico na obtenção de subconjuntos de incidentes com menores variações de tempo de resolução (duração) em cada grupo. Assim, a expectativa é de que os subconjuntos de incidentes de cada grupo tenham duração semelhante. A estratégia para implementação de agrupamento interativo escolhida neste experimento foi a mesma da explorada com o log de eventos sintéticos na seção 3.2, a abordagem CMCR.

O mapeamento do log para o espaço vetorial foi realizado utilizando os três conjuntos de atributos supracitados ($atrib1$, $atrib2$ e $atrib_espec$) e três modelos semânticos baseados em contagem (*binário*, *tf* e *tfidf*). Além disso, como apresentado na subseção 3.3.2, foram exploradas duas maneiras de representação vetorial do log em relação a como tratar os diferentes atributos existentes de forma que eles constituíssem as “atividades” que definem o comportamento dos casos.

Para avaliar os resultados, foram medidas as qualidades dos agrupamentos obtidos por meio do índice Silhouette e de estatísticas descritivas, como média e desvio padrão, do tempo para resolução dos incidentes alocados em cada grupo.

3.3.1 O log de eventos - incidentes

O log real explorado foi utilizado por Amaral, Fantinato e Peres (2018a) e relata eventos de um processo de gerenciamento de incidentes extraído de uma instância da plataforma *ServiceNow*TM utilizada por uma empresa de TI. O log é composto por 24.918 incidentes, extraídos de março de 2016 a fevereiro de 2017, e contém informações anonimizadas por questões de privacidade. A descrição dos atributos consta no documento apresentado por Amaral, Fantinato e Peres (2018b), apresentado na íntegra no anexo A.

Na figura 27, é apresentado um fragmento do formato original do log. Para cada incidente, existe um atributo que define o seu estado dentro do processo básico de gerenciamento de incidentes (atributo *incident_state*) e mais uma série de atributos que contextualizam (descrevem) o incidente (atributos *active*, *reassignment_count*, etc). Para obter o tempo de resolução (duração) de cada um dos incidentes presentes no log, foi realizado o cálculo da diferença entre os atributos *closed_at* e *opened_at*.

Figura 27 – Fragmento do formato original do log de dados reais

	number	incident_state	active	reassignment_count	...	close_code	resolved_by	resolved_at	closed_at
0	INC0000045	New	True	0	...	code 5a	Resolved by 149	29/2/2016 11:29	2016-03-05 12:00:00
1	INC0000045	Resolved	True	0	...	code 5a	Resolved by 149	29/2/2016 11:29	2016-03-05 12:00:00
2	INC0000045	Resolved	True	0	...	code 5a	Resolved by 149	29/2/2016 11:29	2016-03-05 12:00:00

Fonte: Thais Rodrigues Neubauer, 2020

3.3.2 Definição de atividades e escolha da representação vetorial

O log utilizado possui várias características descritivas dos eventos e o primeiro passo do mapeamento vetorial foi definir como essas características podem compor uma “atividade” no sentido aplicado no contexto de mineração de processos. A primeira abordagem para solucionar essa questão foi considerar o valor de cada um dos atributos descritivos do incidente concatenado ao seu rótulo identificador como uma atividade. Essa abordagem será referenciada pelo termo *por atributo individual*. Na segunda abordagem, referenciada por *por combinação de atributos*, além dos valores de cada atributo ser concatenado a seu nome, as concatenações resultantes de cada atributo são também concatenadas e esse segundo resultado de concatenação forma uma atividade. No quadro 2 as duas formas de mapeamento dos atributos são exemplificadas.

Quadro 2 – Exemplo das formas de mapeamento de atributo para atividade no log real de incidentes: (a) exemplo do formato original do log; (b) exemplo da lista de atividades geradas pela abordagem de mapeamento de atributo para atividade denominada *por atributo individual*; (c) exemplo da lista de atividades geradas pela abordagem *por combinação de atributos*

Trace	Atrib.1	Atrib.2	Lista de atividades 1	Lista de atividades 2
Trace 1	a	b	Atrib.1 - a	Atrib.1 - a - Atrib.2 - b
Trace 2	a	c	Atrib.2 - b	Atrib.1 - a - Atrib.2 - c
Trace 3	a	b	Atrib.2 - c	

(a)

(b)

(c)

Fonte: Thais Rodrigues Neubauer, 2020

A fim de avaliar como as diferentes representações podem influenciar os resultados obtidos, tendo em vista o objetivo de agrupar incidentes com tempo de resolução similares, foram realizadas vários testes de agrupamento clássico, com a aplicação do algoritmo *K-means++*, com distância euclidiana e com diferentes quantidades de grupo (valor do parâmetro k), de 3, 5 e 10. O quadro 3 apresenta todos os quesitos e suas variações analisadas nesta etapa do experimento.

Quadro 3 – Variações de cada quesito para o experimento do log real de incidentes

Quesito	Variações
Conjunto de atributos	{alg1, alg2, espec}
Composição de atividade	{Por atributo individual, Por combinação de atributos}
Modelo de contagem	{binária, tf, tfidf}
Valores de k	{3,5,10}

Fonte: Thais Rodrigues Neubauer, 2020

Os resultados de agrupamento alcançado pela combinação das variações dos quesitos apresentados no quadro 3 foram analisados em relação às durações dos incidentes de cada grupo e ao índice Silhouette.

A figura 28 apresenta os resultados do ponto de vista da duração dos incidentes de cada grupo. A fim de orientar a interpretação desse tipo de visualização, os gráficos contém duas linhas em cinza, uma na vertical e uma na horizontal, dividindo-os em quadrantes. O desejável nessas visualizações é que os valores nos eixos x e y sejam os menores possíveis, o que nos leva ao quadrante mais abaixo e à esquerda, em direção à origem do sistema de coordenadas sobre o qual o gráfico é apresentado. Analisando a imagem em (a), vários resultados constam nesse quadrante, tanto das representações de composição de atividade por atributo individual quanto por combinação. Em (b) os

mesmos resultados são apresentados de forma a ressaltar sua distribuição pelos quadrantes em relação aos conjuntos de atributos, revelando que somente os conjuntos “alg1” e “espec” se aproximam do quadrante desejado.

Na figura 29, são apresentados os valores obtidos para o índice Silhouette nos mesmos testes. Com a análise do gráfico apresentado em (a), também não parece haver evidência de padrões de melhora ou piora para algum dos conjuntos de atributos, para alguma composição de atividade ou para alguma das formas de contagem.

Figura 28 – Resultados obtidos com o algoritmo *K-means++* e diferentes representações vetoriais do log de incidentes em termos de duração dos incidentes

(a) Resultados para todas as representações testadas

(b) Visualização dos resultados ressaltando diferenças entre os conjuntos de atributos testados

Fonte: Thais Rodrigues Neubauer, 2020

Com a finalidade de guiar a escolha da representação e também conhecer o perfil de parte dos especialistas envolvidos nos experimentos com o log de incidentes, foi enviado o questionário do anexo C a pessoas que atuam na indústria com a instância da plataforma da qual foi extraído o log do processo de gerenciamento de incidentes explorado nesta seção. Os resultados obtidos com os seis especialistas participantes são apresentados na figura 30.

O questionário foi dividido em quatro partes, as três primeiras sendo tarefas e a última com questões sobre perfil profissional. Depois de uma contextualização sobre o objetivo geral deste trabalho, sobre o sistema de gerenciamento de incidentes em questão e sobre a tarefa de predizer o tempo de duração dos incidentes, na primeira tarefa, foi

Figura 29 – Valores para o índice Silhouette obtidos com o algoritmo *K-means++* e diferentes representações vetoriais do log de incidentes

Fonte: Thais Rodrigues Neubauer, 2020

Figura 30 – Visualizações das respostas dos especialistas ao primeiro questionário

Fonte: Thais Rodrigues Neubauer, 2020

solicitado que o especialista indicasse atributos (pelo menos um e no máximo cinco) que considerasse relevantes para determinar a duração de incidentes. Como indicado na figura 30(a), os atributos mais indicados foram “categoria” e “prioridade”, ambos propostos por cinco dos seis especialistas; juntamente com o atributo “estado”, indicado por metade dos especialistas. Esses atributos formam o conjunto de atributos “espec”, o que já pode ser visto como um indício de que esse é o conjunto de atributos que seria escolhido pelos especialistas. Além de “estado”, os atributos “subcategoria”, “grupo de associação” e “tempo transcorrido” também foram indicados por metade dos especialistas.

Na segunda tarefa, assim como na tarefa anterior, os especialistas deveriam escolher os atributos mais relevantes em relação à determinação da duração de incidentes, mas essa a escolha foi feita a partir da lista apresentada a eles de atributos disponíveis no log. Os atributos escolhidos são apresentados na figura 30(b). Novamente, “categoria” foi escolhido por cinco especialistas e “prioridade” e “estado” também foram parte dos mais escolhidos (três e dois especialistas, respectivamente).

Por fim, na terceira tarefa, os três conjuntos de atributos selecionados (quadro 3) foram apresentados e os especialistas indicaram se os consideravam adequados ou não para determinação da duração de incidentes. Como pode ser observado na figura 30(c), todos os especialistas indicaram o conjunto “espec” como adequando, enquanto somente um deles indicou os outros conjuntos também como adequados.

Quanto às questões da quarta e última parte desse questionário, o objetivo principal foi identificar a experiência profissional dos especialistas em relação ao tema abordado para apoiar a confiança nas respostas obtidas nas tarefas propostas. Os resultados em relação a isso são apresentados nas figuras 30(d) e 30(e). Sobre essa parte, vale ressaltar que: (i) somente um dos especialistas apontou que trabalha há menos de um ano no contexto de gerenciamento de processos de incidentes; o restante trabalha por períodos superiores a esse; (ii) para todos os outros assuntos relevantes no contexto desta pesquisa que foram questionados, gerenciamento de processos, plataforma ServiceNow e Framework ITIL, todos os especialistas participantes tem mais de um ano de experiência; (iii) um dos especialistas classifica como básico seu conhecimento a respeito de gerenciamento de processos e framework ITIL, outro classificou seu conhecimento sobre gerenciamento de processos como avançado e para todos os outros assuntos e todos os outros especialistas, a classificação do conhecimento foi como intermediário.

A análise dessas respostas quanto à experiência profissional juntamente com as respostas às tarefas motivou a decisão de selecionar o conjunto de atributos “espec”. Na figura 31 são apresentados somente os resultados das representações que utilizam esse conjunto. Observando somente esses resultados na visualização da duração dos incidentes, figura 31(a), foi possível notar superioridade nos resultados com $k = 5$ e $k = 10$ em relação aos obtidos com $k = 3$ pela quantidade de dados no quadrante objetivo ou próximos a eles. Tendo em vista os experimentos relacionadas à agrupamento de *traces* interativo planejados para dar continuidade a esses, que demandaram a análise dos grupos obtidos pelos especialistas, foi escolhida a quantidade de $k = 5$ pela simplicidade de análise que uma menor quantidade de grupos pode trazer. Já com a visualização dos valores do índice Silhouette das representações com o conjunto “espec” e $k = 5$, apresentada na parte (b) da mesma figura, é possível perceber que a representação com maior Silhouette, *individual-esp-tfidf*, quando identificada na parte (a) da figura, é também um dos resultados próximos ao quadrante desejado (baixa média de variação da duração dos incidentes nos grupos e baixo desvio da média dessa variação). O resultado da representação *combined-esp-binary* apesar de estar no quadrante desejado, tem um valor de índice Silhouette muito mais baixo e, por isso, foi dada prioridade àquela, *individual-esp-tfidf*.

3.3.3 Análise dos resultados de agrupamento de *traces* interativo

Após a escolha da representação a ser utilizada nesses testes conforme apresentado na subseção 3.3.2, foi planejada a segunda interação com os especialistas, parte da execução dos experimentos de agrupamento de *traces* interativo. A ideia central dessa etapa foi apresentar os resultados obtidos até então com agrupamento clássico e solicitar aos especialistas propostas de melhoria desses resultados em relação à obtenção de grupos com incidentes de tempo de resolução mais semelhantes em cada grupo. O questionário aplicado para esse fim consta na íntegra no anexo D e foi aplicado a dois especialistas acadêmicos e dois especialistas da indústria, que tem experiência no contexto do log explorado e de gerenciamento de processos organizacionais.

A partir do levantamento e agrupamento das sugestões obtidas nas tarefas 1, 3 e 4 desse questionário (disponíveis no anexo D), foram escolhidas uma referente a cada um dos atributos utilizados, apresentadas na tabela 8.

Figura 31 – Seleção dos resultados obtidos com o algoritmo *K-means++* e representações vetoriais do log de incidentes com o conjunto de atributos “espec”

(a) Visualização da duração dos incidentes.

(b) Visualização do índice Silhouette para resultados de $k = 5$.

Fonte: Thais Rodrigues Neubauer, 2020

Outra questão de interesse neste trabalho pertinente ao contexto da aplicação de agrupamento interativo é avaliar como a abordagem interativa pode ser diferente de abordagens semi-supervisionadas, nas quais há inserção de conhecimento humano, mas *a priori*. Para esse fim, foi também desenvolvida uma atividade de controle para avaliar se as informações apresentadas dos resultados obtidos até o momento de fato podem alterar as sugestões para agrupamento fornecidas pelos especialistas. Para tanto, três especialistas - dois acadêmicos com experiência no contexto de gerenciamento de processos organizacionais e um atuante na indústria com a instância da plataforma da qual foi extraído o log - realizaram o agrupamento manual dos valores dos atributos selecionados. Vale ressaltar que para o atributo “category”, somente os 15 valores mais comuns foram apresentados. Essa atividade de controle consta na íntegra no anexo E.

Com a análise dos agrupamentos realizados e suas justificativas, as seguintes heurísticas foram identificadas em ambos os questionários como formas de separação mais adequada dos incidentes em grupos:

- separar níveis de prioridade em grupos diferentes;

Tabela 8 – Restrições levantadas junto aos especialistas e selecionadas para aplicação

incident_state	Execuções com objetivo de manter em um mesmo grupo todos os incidentes que passaram pelos estados “Awaiting ...”. A justificativa para tal é que esses incidentes podem ter maior duração por conta do período gasto com uma espera. A aplicação no processo de agrupamento foi realizada com a criação de restrições do tipo <i>must-link</i> foram criadas entre os incidentes que passaram por esses quatro estados (<i>Awaiting Evidence</i> , <i>Awaiting Problem</i> , <i>Awaiting User Info</i> e <i>Awaiting Vendor</i>), de forma a garantir que esses incidentes sejam alocados ao mesmo grupo, mas sem impedir que outros incidentes incomuns (semelhantes a esses em relação à duração do incidente) também resultem nesse grupo.
priority	Execuções com objetivo de dividir os incidentes com prioridades mais altas (valores 1 e 2), dos com prioridade média (valor 3) dos com prioridade baixa (valor 4). Sua aplicação foi realizada por meio da criação de restrições do tipo <i>cannot-link</i> entre os incidentes com esses diferentes níveis de prioridade, tendo em mente que, em geral, o nível de prioridade deveria alterar a velocidade com que um incidente é resolvido.
category	Execuções com objetivo de colocar em um mesmo grupo incidentes que parecem ser de fácil resolução e que poderiam até mesmo, por exemplo, ser resolvido remotamente: “Password and access”, “Remote access” e “Office pack”. Para incluir essa ideia no processo de agrupamento, restrições do tipo <i>must-link</i> foram criadas entre os incidentes categorizados em alguma delas, para, semelhantemente à estratégia referida nas restrições para o atributo incident_state, garantir que esses incidentes sejam alocados ao mesmo grupo, sem impedir que outros incidentes semelhantes a esses em relação à duração do incidente também resultem nesse grupo.

Fonte: Thais Rodrigues Neubauer, 2020

- manter categorias similares (em relação à área afetada ou ao grupo responsável por resolver problemas daquela natureza).

Contudo, a preocupação com tratamento dos casos de incidentes que passam por estados do tipo “Awaiting ...” só foi trazido no primeiro questionário e, apesar de não ser totalmente conclusivo e mais experimentos nesse sentido sendo necessários, essa observação já indica uma possível diferenciação do tipo de análise trazida pelos especialistas quando informações sobre um agrupamento anterior são apresentadas.

Quanto a quantidade de grupos, dois especialistas apontam que menos grupos já seriam suficientes para dividir os incidentes de forma adequada e o outro acredita que mais grupos seriam necessários.

A partir das respostas obtidas no questionário e posteriormente selecionadas, apresentadas na tabela 8, foram criadas restrições. Os resultados apresentados na figura 32 foram alcançados com a aplicação dessas restrições. Semelhantemente às visualizações apresentadas nas figuras 28 e 31(a) da seção 3.3.2, o desejado é que os resultados estejam

localizados no quadrante mais abaixo e a esquerda das linhas em cinza que tem como origem o resultado com o agrupamento clássico. Apesar de nenhum dos resultados ter alcançado essa posição, alguns dos experimentos com restrições no atributo “incident_state” ou no atributo “priority” alcançaram resultados nos quais houve declínio na variação da duração dos incidentes em alguns grupos, acarretando na diminuição da média da variação nos grupos - resultados esses localizados, portanto, no quadrante mais abaixo e a direita do resultado alcançado com o agrupamento clássico.

Ainda, a fim de verificar como as restrições aplicadas alterariam os resultados alcançados por um preditor de tempo de resolução de incidentes mais robusto do que somente a média em cada grupo, os sublogs dos grupos obtidos para resultados selecionados foram executados pelo preditor utilizado em Fernandes (2019). Como apresentado na visualização da figura 32(b), que pode ser interpretada da mesma forma que a figura 32(a), pois o desejado é que o erro RMSPE médio dos grupos seja pequeno, assim como o desvio do erro entre os grupos, vários dos resultados dos experimentos com CRCM são melhores do que o alcançado com agrupamento clássico. Na mesma figura, no canto superior esquerdo, consta também o erro médio do preditor utilizando o log completo ao invés dos sublogs referentes aos grupos, apontando que todas as execuções de agrupamento de *traces* selecionadas levam a erros menores e, portanto, provando também a importância de agrupamento de *traces* para a tarefa de predição de tempo de resolução de incidentes.

Em relação à análise do índice Silhouette para os experimentos com a abordagem CMCR, os valores variaram de 0.24 a 0.68, sem que fosse observado padrão da variação em relação às proporções de *trace* e casos.

As análises das medidas de qualidade de modelo de processos, definidas na seção 2.2, são apresentadas na figura 33. Comparando essas medidas entre experimentos com CMCR, com o agrupamento clássico e com o resultado alcançado pelo log completo (linhas cinzas):

- *Fitness*: todos os valores obtidos com CMCR foram semelhantes ou superiores ao resultado com agrupamento clássico, que, por sua vez, foi superior ao valor obtido com o log completo.
- Generalização: todos os resultados foram semelhantes ao resultado obtido com o log completo e superior ao obtido com agrupamento clássico.
- Precisão: o resultado com o log completo foi um pouco superior ao com agrupamento clássico. Em relação às execuções com a abordagem CMCR, os valores obtidos

Figura 32 – Resultados obtidos com a abordagem CMCR no log de incidentes

(a) Visualização da duração dos incidentes.

(b) Visualização da média do erro do preditor nos experimentos selecionados.

Fonte: Thais Rodrigues Neubauer, 2020

oscilaram entre 0.2 e 0.6 sem que fosse indentificado um padrão nas oscilações. Para o atributo “incident_state” e proporção de casos = 1.0, os resultados obtidos variaram entre semelhantes ao resultado com agrupamento clássico e semelhantes entre o resultado com o log completo.

- Simplicidade: todos os experimentos com CMCR foram inferiores ao resultado alcançado com agrupamento clássico, mas foram semelhante ou pouco superiores ao alcançados com o log completo.

Na segunda tarefa do questionário aplicado, os especialistas foram questionados a respeito da quantidade de grupos apresentada: dois deles apontaram que acreditam que menos grupos seriam suficientes para obter uma boa separação dos incidentes em relação a duração enquanto a outra metade apontou que o ideal seria aumentar a quantidade.

Lembrando que um dos objetivos deste trabalho é avaliar também o custo que os especialistas dispenderam em cada sessão de participação do processo de agrupamento, o

Figura 33 – Resultados obtidos com o algoritmo *Cop-Kmeans* no log de incidentes

Fonte: Thais Rodrigues Neubauer, 2020

horário de início e término do questionário foi solicitado a cada um. O tempo dispendido para responder ao questionário foi de 43 a 78 min, com média de 58 min, o que não foi considerado um envolvimento muito longo e dispendioso se a quantidade de interações for baixa. Para o caso de uma maior quantidade de interações, um primeiro ponto a ser considerado é que possivelmente o tempo da primeira interação seja um pouco maior do que as demais, pois ela envolve uma ambientação e um aprendizado por parte do especialista que provavelmente é bastante diminuído quanto maior a quantidade de interações. Ainda assim, um outro ponto relevante para aumentar a quantidade de interações é o investimento na simplificação das tarefas propostas.

3.3.4 Considerações finais sobre os experimentos com o log de eventos reais

Os experimentos conduzidos com o log de eventos reais revelaram que a aplicação de restrições com a abordagem CMCR pode levar a melhoria na qualidade dos resultados obtidos em relação à tarefa de predição de tempo de resolução de incidentes, como demonstrado na figura 32(b). Em relação às análises de qualidade do ponto de vista de mineração de dados e de mineração de processos, as análises levam a conclusões bastante semelhantes às alcançadas com os experimentos com os logs de eventos sintéticos.

4 Conclusão

Neste capítulo, são apresentadas as conclusões gerais deste trabalho, seguidas pelas contribuições, limitações impostas e pelas oportunidades de trabalhos futuros.

A proposta deste trabalho foi a aplicação de agrupamento interativo em mineração de processos – essa junção de áreas foi nomeada como agrupamento de *traces* interativo. O objetivo principal foi verificar se estratégias já existentes de inclusão de conhecimento de especialistas no processo de agrupamento encontradas na literatura de mineração de dados se aplicam ao contexto de mineração de processos. Além disso, foi analisada como essa inserção pode afetar a qualidade dos agrupamentos obtidos, tanto do ponto de vista de mineração de dados quanto do ponto de vista de mineração de processos. Para alcançar esse objetivo, a abordagem de agrupamento interativo, identificada como CMCR, foi aplicada nos experimentos deste trabalho, projetados em dois conjuntos: um com um log de eventos sintéticos e um com um log de eventos reais.

O log de eventos sintéticos, apresentados na seção 3.2, foi utilizado para iniciar a avaliação da aplicação da abordagem CMCR em um ambiente controlado, que permitiu a abstração de questões semânticas do log. Foram utilizadas duas variações desse log e foram elaborados três cenários com diferentes metas de separação de *traces* para simular a expectativa de especialistas.

A partir da análise por comparação do índice Silhouette entre os resultados obtidos pelo agrupamento clássico e pela abordagem CMCR, foi possível confirmar a hipótese levantada na seção 1.2 de que o atendimento às expectativas de especialistas (aplicação das restrições no caso da abordagem CMCR) para obter resultados mais aderentes ao contexto do processo de negócio, não causa degradação significativa na qualidade dos grupos do ponto de vista de mineração de dados.

Já sob a ótica de mineração de processos, conforme apresentado principalmente na figura comparativa 26:

- Para as métricas de fitness e simplicidade, os resultados alcançados a partir dos resultados da aplicação da abordagem CMCR não foram significativamente diferentes dos alcançados a partir dos resultados da aplicação do agrupamento clássico nem dos alcançados a partir do log completo.

- Para a métrica de precisão no cenário 2 e com perfil por transição, foi possível observar que os valores obtidos a partir dos resultados da aplicação da abordagem CMCR foram significativamente menores quando comparados aos obtidos a partir dos resultados do agrupamento clássico e do log completo.
- Para a métrica de generalização, nos cenários 2 e 3, com ênfase no último, os valores obtidos a partir dos resultados da aplicação da abordagem CMCR também foram significativamente menores do que os obtidos a partir dos resultados do agrupamento clássico e do log completo.

Quanto à influência das representações de *traces* nos resultados, apesar de ter sido possível notar valores de Silhouette um pouco melhores, por exemplo, com a representação binária e perfil por transição no cenário 1, as diferenças observadas entre diferentes representações não foram significativas ou conclusivas sob os pontos de vista analisados.

Ainda, em relação à quantidade de dados incluídos nas restrições aplicadas, também foi possível observar que, independentemente da quantidade de dados envolvidos, não há perda no aprendizado do algoritmo com a aplicação das restrições, ou seja, a mesma quantidade de dados que já estavam de acordo com a expectativa dos especialistas antes da aplicação das restrições permaneceu a mesma depois da aplicação das restrições. Contudo, foi possível observar que a utilização de parte dos dados (porcentagens tanto de *traces* quanto de casos) em uma restrição não garante o alcance da expectativa delineada.

Os experimentos apresentados na seção 3.3 utilizando um log de incidentes foram realizados com a finalidade de avaliar essas questões e analisar a aplicabilidade do agrupamento interativo em um contexto real de mineração de processos. Nesse cenário, a expectativa dos especialistas era que os incidentes fossem separados em grupos de forma que fosse facilitada a tarefa de predição de tempo de resolução dos incidentes em cada grupo. Por isso, além de avaliar os resultados obtidos em termos de métrica de mineração de dados e de mineração de processos como feito nos cenários com logs sintéticos, foi avaliado também o quão similar era a duração dos incidentes de cada grupo por meio da média de variação da duração dos incidentes nos grupos e o desvio dessa média entre os grupos. Além disso, para alguns resultados selecionados a partir da análise dessa similaridade de duração intra-grupos, foi aplicado um preditor de tempo de resolução em cada grupo e calculada a média do erro de predição dos grupos.

As conclusões obtidas a partir da análise dos resultados obtidos com esse log de incidentes sob o ponto de vista de mineração de dados e de mineração de processos foram os mesmos dos obtidos nos experimentos com os logs sintéticos. Em relação à tarefa de predição de tempo de resolução de incidentes:

- A partir da visualização da média de variação da duração dos incidentes nos grupos e o desvio dessa média entre os grupos, foi possível notar que em algumas execuções houve indícios de melhora na média de variação, mas não em termos de desvio dessa média.
- A partir da análise do erro médio do preditor de tempo de resolução de incidente nos grupos, foi possível notar que, em algumas execuções, houve melhora tanto no erro (menores valores de erro) quanto no desvio do erro entre os grupos, significando melhora em todos os grupos dessas execuções.

Tanto do ponto de vista de mineração de dados quanto de mineração de processos e da facilitação da tarefa da predição de tempo de resolução dos incidentes, não foram identificados padrões de melhora ou piora dos resultados a depender do atributo sob o qual as restrições são aplicadas ou da proporção de *traces* e casos.

Por fim, o conjunto de experimentos envolvendo esse log de incidentes e especialistas externos a esta pesquisa, possibilitou a medição do esforço do especialista em termos de tempo despendido para participação no processo de agrupamento. Analisando as participações dos especialistas nas fases de análise de grupos obtidos previamente e sugestão de heurísticas para melhorar a separação dos incidentes, a média de tempo de participação por especialista foi de uma hora. Esse esforço pode ser considerado baixo se levado em conta o interesse do especialista na melhora dos resultados obtidos e se a quantidade sessões de interações não for muito alta.

4.1 Contribuições

A principal contribuição deste trabalho foi a experimentação do agrupamento interativo no contexto de dados sintéticos e reais da área de mineração de processos.

Outra contribuição foram as diferentes formas de avaliação da inclusão do conhecimento de especialista na tarefa de agrupamento de *traces*.

A avaliação da possibilidade de atingir o objetivo das restrições usando parte dos dados que seriam nelas envolvidos, com o estudo de força e espaço explicado na seção 3.1, é considerada mais uma contribuição deste trabalho.

A última contribuição percebida foi a elucidação da necessidade de avaliação da diferença entre os resultados obtidos com a abordagem interativa e com abordagens semi-supervisionadas. Naquela, a supervisão é incluída no processo de separação dos dados após a apresentação de um resultado preliminar, enquanto que na abordagem semi-supervisionada, a inclusão é realizada antes do processo ser iniciado, fazendo parte dos dados de entrada do problema. Essa contribuição foi esboçada pela aplicação da atividade de controle, explanada na seção 3.3.3, por meio da qual foi iniciada a avaliação da diferença que a apresentação de um resultado preliminar de fato pode fazer na supervisão coletada.

4.2 Limitações

A primeira limitação identificada neste estudo é a execução da maior parte dos testes por somente uma vez, possibilitando que os resultados sejam influenciados significativamente por fatores de aleatoriedade.

Uma outra questão é a quantidade de especialistas participantes dos experimentos, pois para aumentar a confiabilidade das conclusões, os questionários deveriam ser aplicados a mais especialistas e com maior diversidade de nível de conhecimento sobre o contexto.

Existe ainda a limitação de tempo de participação que cada especialista disponibiliza à tarefa de supervisão, relacionado a possibilidades pessoais e/ou engajamento. Ainda sobre a participação de especialistas, é importante ressaltar que a aferição do esforço do especialista é uma tarefa subjetiva e relativa para cada pessoa e, portanto, o modo utilizado para tanto neste trabalho pode não refletir completamente a ideia de esforço que cada pessoa pode tomar para si.

Por fim, as formas de avaliação da intervenção na qualidade dos grupos obtidos como resultado da tarefa de agrupamento podem não refletir aspectos subjetivos e importantes como o alinhamento desse resultado com a expectativa humana.

4.3 *Trabalhos futuros*

Os primeiros pontos levantados para trabalhos futuros são iniciativas que minimizariam duas das limitações apresentadas: repetição das execuções dos experimentos realizados por mais vezes para minimizar fatores de aleatoriedade de inicialização de processos de agrupamento; e aplicação dos questionários a mais especialistas.

Além desses, os seguintes pontos foram levantados como pontos relevantes para continuação da pesquisa realizada neste trabalho:

- Selecionar os dados a serem envolvidos nas restrições com base em medidas de interessabilidade, utilizando, por exemplo, estratégias de *active learning* para isolar o fator de aleatoriedade nas restrições.
- Executar o preditor de tempo de resolução de incidentes para todos os resultados obtidos com a aplicação da abordagem CMCR no log de eventos reais.

Referências¹

- AALST, A. K. A. M. W. M. P. van der; WEIJTERS, A. Process equivalence: Comparing two process models based on observed behavior. In: *Business Process Management*. Berlin, Germany: Springer, 2006. (Lecture Notes in Computer Science, v. 4102), p. 129–144. Citado na página 39.
- AALST, W. M. P. van der. *Process mining – discovery, conformance and enhancement of business processes*. 1st. ed. Berlin, Germany: Springer, 2011. Citado 5 vezes nas páginas 22, 34, 36, 37 e 38.
- AALST, W. M. P. van der. *Process Mining: Data Science in Action*. 2. ed. Berlin, Germany: Springer, 2016. Citado 3 vezes nas páginas 15, 34 e 35.
- AALST, W. M. P. van der. Benchmarking logs to test scalability of process discovery algorithms. Logs dataset and related documentation published by Eindhoven University of Technology (The Netherlands), at 4tu Center for Research Data. 2017. Citado 2 vezes nas páginas 54 e 55.
- AALST, W. M. P. van der; ADRIANSYAH, A.; DONGEN van B. F. Replaying history on process models for conformance checking and performance analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, v. 2, n. 2, p. 182–192, 2012. Citado 3 vezes nas páginas 36, 37 e 38.
- AALST, W. M. P. van der; SCHONENBERG, M.; SONGA, M. Time prediction based on process mining. *Information Systems*, Elsevier B.V., v. 36, n. 2, p. 450–475, 2011. Citado na página 23.
- ACHLIOPTAS, D.; MCSHERRY, F. On spectral learning of mixtures of distributions. In: *Proceedings of the 18th Annual Conference on Learning Theory*. Berlin, Germany: Springer-Verlag, 2005. (COLT’05), p. 458–469. Citado 2 vezes nas páginas 17 e 26.
- AGRAWAL, R.; BAYARDO, R.; SRIKANT, R. Athena: Mining-based interactive management of text databases. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 1777, p. 365–379, 2000. Citado na página 29.
- AMARAL, C. A. L.; FANTINATO, M.; PERES, S. M. Attribute selection with filter and wrapper: An application on an incident management process. In: *Proceedings of the 2018 Federal Conference on Computer Science and Information Systems*. Washington, DC, USA: IEEE, 2018. (Annals of Computer Science and Information Systems, v. 15), p. 679–682. Citado 4 vezes nas páginas 22, 72, 73 e 74.
- AMARAL, C. A. L.; FANTINATO, M.; PERES, S. M. *Attributes description: enriched event log of an incident management process*. 2018. (http://each.uspnet.usp.br/sarajane/wp-content/uploads/2018/09/Incidents_event_log_description.pdf). Citado na página 74.
- APPICE, A.; MALERBA, D. A co-training strategy for multiple view clustering in process mining. *IEEE Transactions on Services Computing*, v. 9, p. 832–845, 2016. Citado 7 vezes nas páginas 16, 17, 41, 43, 45, 46 e 47.

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

ARMSTRONG, J.; COLLOPY, F. Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, v. 8, p. 69–80, 06 1992. Citado na página 23.

AWASTHI, P.; BALCAN, M.; VOEVODSKI, K. Local algorithms for interactive clustering. *Journal of Machine Learning Research*, v. 18, p. 1–35, 2017. Citado 4 vezes nas páginas 17, 26, 30 e 31.

BABAKI, B. *COP-Kmeans version 1.5*. 2017. Disponível em: <https://doi.org/10.5281/zenodo.831850>. Citado na página 33.

BAEZA-YATES, R. A.; RIBEIRO-NETO, B. *Modern information retrieval*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., 1999. Citado 2 vezes nas páginas 44 e 47.

BELKIN, M.; SINHA, K. Polynomial learning of distribution families. In: *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE Computer Society, 2010. (FOCS '10), p. 103–112. ISBN 978-0-7695-4244-7. Citado 2 vezes nas páginas 17 e 26.

BOSE, R. P. J. C.; AALST, W. M. P. van der. Context aware trace clustering: Towards improving process mining results. In: *Proceedings of the SIAM International Conference on Data Mining*. Sparks, Nevada, USA: SIAM, 2009. p. 401—412. Citado 3 vezes nas páginas 16, 42 e 43.

BOSE, R. P. J. C.; AALST, W. M. P. van der. Trace clustering based on conserved patterns: Towards achieving better process models. In: *Business Process Management Workshops*. Germany: Springer, 2009. p. 170—181. Citado 2 vezes nas páginas 42 e 43.

BRUBAKER, S. C.; VEMPALA, S. Isotropic pca and affine-invariant clustering. In: *Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE Computer Society, 2008. (FOCS '08), p. 551–560. Citado 2 vezes nas páginas 17 e 26.

CADEZ, I. V. et al. Model-based clustering and visualization of navigation patterns on a web site. *Data Mining Knowledge Discovery*, v. 7, n. 4, 2003. Citado na página 42.

CAO, N. et al. Dicon: Interactive visual analysis of multidimensional clusters. *IEEE Transactions on Visualization and Computer Graphics*, v. 17, n. 12, p. 2581–2590, 2011. Citado na página 29.

CAVALLO, M.; DEMIRALP, Ç. Clustrophile 2: Guided visual clustering analysis. *IEEE Transactions on Visualization and Computer Graphics*, v. 25, p. 267–276, 2018. Citado na página 29.

CHANG, S. et al. Appgrouper: Knowledge-based interactive clustering tool for app search results. In: *Proceedings of the 21st International Conference on Intelligent User Interfaces*. USA: ACM, 2016. (IUI '16), p. 348–358. Citado 2 vezes nas páginas 27 e 29.

CHAUDHURI, K.; DASGUPTA, S. Rates of convergence for the cluster tree. In: LAFFERTY, J. D. et al. (Ed.). *Advances in Neural Information Processing Systems 23*. USA: Curran Associates, 2010. p. 343–351. Citado 4 vezes nas páginas 17, 26, 30 e 31.

- CODEN, A. et al. A method to accelerate human in the loop clustering. In: *Proceedings of the 17th SIAM International Conference on Data Mining*. Philadelphia, Pennsylvania, United States: Society for Industrial and Applied Mathematics, 2017. p. 237–245. Citado 3 vezes nas páginas 17, 18 e 26.
- CORREA, G. et al. Interactive textual feature selection for consensus clustering. *Pattern Recognition Letters*, v. 52, p. 25–31, 2015. Citado 5 vezes nas páginas 17, 18, 26, 29 e 31.
- DEZA, M. M.; DEZA, E. *Encyclopedia of Distances*. 1. ed. Berlin, Germany: Springer, 2009. Citado na página 25.
- DODIG-CRANKOVIC, G. Scientific Methods in Computer Science. In: *Conference for the Promotion of Research in IT at New Universities and at University Colleges in Sweden*. Skövde: [s.n.], 2002. Citado na página 21.
- DUMAS, M. et al. *Fundamentals of Business Process Management, Second Edition*. [S.l.]: Springer, 2018. Citado 3 vezes nas páginas 15, 33 e 34.
- DYBA, T.; DINGSOYR, T.; HANSEN, G. K. Applying systematic reviews to diverse study types: An experience report. In: *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)*. [S.l.: s.n.], 2007. p. 225–234. Citado na página 99.
- ESTER, M. et al. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. [S.l.]: AAAI Press, 1996. p. 226–231. Citado na página 25.
- FERNANDES, A. G. L. *Tratamento do impacto de casos non-fitting em predição de tempo de resolução usando mineração de processos com múltiplos atributos*. São Paulo: Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, 2019. Citado 2 vezes nas páginas 23 e 82.
- FERREIRA, D. R. et al. Approaching process mining with sequence clustering: Experiments and findings. In: *Business Process Management, 5th International Conference*. Berlin, Germany: Springer, 2007. p. 360–374. Citado na página 42.
- FOLINO, F.; GUARASCIO, M.; PONTIERI, L. A framework for the discovery of predictive fix-time models. In: *Proceedings of the 16th International Conference on Enterprise Information Systems*. Portugal: SCITEPRESS - Science and Technology Publications, 2014. v. 1, p. 99–108. Citado 2 vezes nas páginas 23 e 73.
- GEORGESCU, R. et al. Scalable human-in-the-loop decision support. In: *Proceedings of the IEEE Aerospace Conference*. USA: IEEE, 2015. v. 2015-June. Citado na página 29.
- GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de Pesquisa*. Porto Alegre, RS, Brasil: UFRGS Editora, 2009. Citado na página 21.
- GIL, A. C. *Métodos e técnicas de pesquisa social*. 6a. ed. São Paulo, SP, Brasil: Editora Atlas, 2008. Citado na página 21.
- GRECO, G. et al. Discovering expressive process models by clustering log traces. *IEEE Transactions on Knowledge and Data Engineering*, v. 18, n. 8, p. 1010–1027, 2006. Citado 2 vezes nas páginas 16 e 43.

GRÜNWALD, P. D. *The Minimum Description Length Principle*. MA: The MIT Press, 2007. Citado na página 38.

HALKIDI, M.; BATISTAKIS, Y.; VAZIRGIANNIS, M. On clustering validation techniques. *Journal of Intelligent Information Systems*, v. 17, n. 2-3, p. 107–145, 2001. Citado na página 28.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. 3rd. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011. Citado 2 vezes nas páginas 15 e 24.

HOMPES, B. et al. Detecting changes in process behavior using comparative case clustering. *Lecture Notes in Business Information Processing*, v. 244, p. 54–75, 2017. Citado 3 vezes nas páginas 16, 42 e 43.

HOSSAIN, H.; KHAN, M.; ROY, N. Active learning enabled activity recognition. *Pervasive and Mobile Computing*, v. 38, p. 312–330, 2017. Citado na página 29.

HU, Y.; MILIOS, E. E.; BLUSTEIN, J. Interactive document clustering with feature supervision through reweighting. *Intelligent Data Analysis*, v. 18, p. 561–581, 01 2014. Citado 3 vezes nas páginas 17, 26 e 27.

JANG, S.; ELMQVIST, N.; RAMANI, K. Gestureanalyzer: Visual analytics for pattern analysis of mid-air hand gestures. In: *Proceedings of the 2nd ACM Symposium on Spatial User Interaction*. New York, NY, USA: ACM, 2014. p. 30–39. Citado na página 29.

KALAI, A. T.; MOITRA, A.; VALIANT, G. Efficiently learning mixtures of two gaussians. In: *Proceedings of the Forty-second ACM Symposium on Theory of Computing*. New York, NY, USA: ACM, 2010. (STOC '10), p. 553–562. Citado 2 vezes nas páginas 17 e 26.

KHODABANDEH, M. et al. Discovering human interactions in videos with limited data labeling. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. New Jersey, USA: IEEE, 2015. v. 2015-October, p. 9–18. Citado na página 29.

KONINCK, P.; BROUCKE, S.; WEERDT, J. act2vec, trace2vec, log2vec, and model2vec: Representation learning for business processes. In: *Business Process Management*. Berlin, Germany: Springer, 2018. (Lecture Notes in Computer Science, vol 11080), p. 305–321. Citado na página 47.

KONINCK, P. et al. An approach for incorporating expert knowledge in trace clustering. In: *Advanced Information Systems Engineering - 29th International Conference*. Germany: CAiSE, 2017. Citado 3 vezes nas páginas 16, 42 e 47.

LEI, Y. et al. Interactive k -means clustering method based on user behavior for different analysis target in medicine. *Computational and Mathematical Methods in Medicine*, v. 2017, 2017. Citado 4 vezes nas páginas 17, 18, 26 e 29.

LEONI, M. de; AALST, W. M. P. van der; DEES, M. A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. *Information Systems*, v. 56, p. 235–257, 2015. Citado 3 vezes nas páginas 15, 16 e 41.

LU, X. *Using behavioral context in process mining: exploration, preprocessing and analysis of event data*. Tese (PhD dissertation) — Eindhoven University of Technology, 2018. Citado 6 vezes nas páginas 16, 36, 39, 41, 42 e 43.

LU, X.; FAHLAND, D.; AALST, W. M. Conformance checking based on partially ordered event data. In: *Business Process Management Workshops*. Berlin, Germany: Springer, 2014. (Lecture Notes in Business Information Processing, v. 202), p. 75–88. Citado 2 vezes nas páginas 17 e 40.

LUXBURG, U. von. A tutorial on spectral clustering. *Statistics and Computing*, v. 17, p. 395–416, 2007. Citado na página 30.

MAITA, A. R. C. *Um estudo da aplicação de técnicas de inteligência computacional e de aprendizado em máquina de mineração de processos de negócio*. São Paulo: Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, 2016. Citado na página 36.

MAITA, A. R. C. et al. A systematic mapping study of process mining. *Enterprise Information Systems*, v. 12, p. 1–45, 2017. Citado 2 vezes nas páginas 15 e 22.

MARUSTER, L. et al. Process mining: Discovering direct successors in process logs. In: *Discovery Science*. [S.l.]: Springer, 2002. (Lecture Notes in Computer Science, v. 2534), p. 364–373. Citado na página 15.

MAUDER, M.; BOBKOVA, Y.; NTOUTSI, E. Gmmbuilder-user-driven discovery of clustering structure for bioarchaeology. *Lecture Notes in Computer Science*, v. 9853 LNCS, p. 8–11, 2016. Citado na página 29.

MEINCHEIM, A. et al. Combining process mining with trace clustering: Manufacturing shop floor process - an applied case. In: *IEEE 29th International Conference on Tools with Artificial Intelligence*. Washington, DC, USA: IEEE, 2017. p. 498–505. Citado na página 16.

MOITRA, A.; VALIANT, G. Settling the polynomial learnability of mixtures of gaussians. In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE, 2010. p. 93–102. Citado 2 vezes nas páginas 17 e 26.

MUNOZ-GAMA, J. *Conformance checking and diagnosis in process mining*. Tese (PhD dissertation) — Universitat Politècnica de Catalunya, 2014. Citado na página 39.

RINALDO, A. Generalized density clustering. *Annals of Statistics*, p. 2678–2722, 2010. Citado 2 vezes nas páginas 17 e 26.

ROUSSEEUW, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, v. 20, n. 1, p. 53–65, 1986. Citado na página 29.

ROZINAT, D. A. *Disco User's Guide*. 2018. (<https://fluxicon.com/disco/files/Disco-User-Guide.pdf>). Citado na página 55.

RUTAYISIRE, T. et al. A modified cop-kmeans algorithm based on sequenced cannot-link set. In: *Rough Sets and Knowledge Technology*. Berlin, Heidelberg: Springer, 2011. p. 217–225. Citado na página 33.

- SILVA, L. A. da; PERES, S. M.; BOSCARIOLI, C. *Introdução à Mineração de Dados: Com Aplicações em R*. Brasil: Elsevier, 2016. 277 p. Citado 2 vezes nas páginas 15 e 47.
- SONG, M.; GUNTHER, C. W.; AALST, W. M. P. van der. Trace clustering in process mining. In: *Business Process Management Workshops*. Berlin, Germany: Springer, 2008. p. 109–120. Citado 3 vezes nas páginas 16, 41 e 43.
- TURNEY, P. D.; PANTEL, P. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, v. 37, n. 1, p. 141–188, 2010. Citado na página 47.
- VIKRAM, S.; DASGUPTA, S. Interactive bayesian hierarchical clustering. In: *Proceedings of the 33rd International Conference on Machine Learning*. USA: JMLR.org, 2016. v. 5, p. 3033–3042. Citado 3 vezes nas páginas 17, 18 e 26.
- WAGSTAFF, K. et al. Constrained k-means clustering with background knowledge. In: *Proceedings of the Eighteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001. (ICML 2001), p. 577–584. Citado 2 vezes nas páginas 32 e 33.
- WANG, M. et al. Intelligent photo clustering with user interaction and distance metric learning. *Pattern Recognition Letters*, v. 33, n. 4, p. 462–470, 2012. Citado na página 29.
- WEERDT, J. et al. Active trace clustering for improved process discovery. *IEEE Transactions on Knowledge and Data Engineering*, v. 25, n. 12, p. 2708–2720, 2013. Citado 2 vezes nas páginas 16 e 42.
- WESKE, M. *Business Process Management: Concepts, Languages, Architectures*. 2. ed. Berlin, Germany: Springer, 2007. Citado 3 vezes nas páginas 15, 33 e 34.
- YANG, S. et al. A data-driven process recommender framework. In: *23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2017. (KDD '17), p. 2111–2120. ISBN 978-1-4503-4887-4. Citado na página 16.

Apêndice A – Protocolo para revisão sistemática: agrupamento interativo

Os passos desse protocolo são: *(A)* seleção das bases de dados, *(B)* elaboração da estratégia de pesquisa, *(C)* definição dos critérios de seleção, *(D)* definição do métodos de extração de dados e síntese de resultados e *(E)* definição dos critérios de qualidade.

- A. **Bases de dados:** dado o interesse em agrupamento interativo do ponto de vista de Ciência da Computação, várias são as editoras que publicam trabalhos com esse foco, incluindo Springer, ACM, IEEE, Elsevier, Taylor & Francis, Inderscience, Emerald, World Scientific e Wiley. Os trabalhos publicados por essas e outras editoras são indexados pela *Scopus*¹, conhecida como a maior base de dados de literatura revisada por pares. Por isso, Scopus foi escolhida para explorar as bases de dados dessas editoras e de outras relacionadas. Devido a grande quantidade de bancos de dados indexados pelo Scopus, essa revisão também abrange editoras de universidades e outras associações científicas. Pesquisas realizadas diretamente nas bibliotecas IEEE, ACM ou Springer, por exemplo, seriam redundantes e, portanto, desnecessárias.
- B. **Estratégia de pesquisa:** foi criada uma string de busca compreendendo palavras-chave relacionadas a agrupamento interativo. Por meio de análise exploratória, descobrimos que os pesquisadores usaram termos diferentes para se referir a agrupamento interativo. Assim, a cadeia de pesquisa compreende disjunções de termos sinônimos de agrupamento interativo obtidos na literatura especializada. Além disso, a sequência da string de busca possui uma segunda parte projetada para filtrar apenas documentos especificamente relacionados à mineração de dados ou áreas relacionadas. Uma representação canônica da string de busca é apresentada na tabela 9. Para limitar de forma controlada a amplitude dos resultados trazidos pela primeira parte da cadeia de pesquisa, usamos o operador de proximidade W/n , que define a distância entre as palavras, mas não a ordem. Por exemplo, *user W/3 cluster* define que *user* pode ser encontrado a três palavras de *cluster* em qualquer ordem.

¹ <https://www.elsevier.com/solutions/scopus/how-scopus-works/content>

Tabela 9 – Representação canônica da string de busca da revisão sistemática

(((((interactive* OR “human in the loop*” OR “human computation” OR “active learn*”) OR ((human* OR user* OR client* OR customer* OR expert*) W/3 (collaborat* OR cooperat* OR supervis* OR knowledge OR participat* OR interact*))) W/3 (cluster* OR unsupervis*))) **AND** (cluster* OR mining OR “machine learning” OR “data science” OR “pattern recogni*” OR intelligen*))

Fonte: Thais Rodrigues Neubauer, 2020

C. Estratégia de seleção de estudos primários: devido ao amplo espectro da string de busca, os resultados precisam ser verificados a fim de identificar falso-positivos. As tabelas 10 e 11 mostram, respectivamente, os Critérios de Inclusão (CI) e os Critérios de Exclusão (CE) definidos para essa verificação. Para um estudo retornado pela busca ser selecionado como um estudo primário da revisão, ele deve cumprir todos os critérios de inclusão e nenhum dos critérios de exclusão.

Tabela 10 – Critérios de Inclusão (CI)

CI-1	Agrupamento interativo estar entre os objetivos de pesquisa.
CI-2	O estudo aborda a aplicação de técnicas computacionais de agrupamento com intervenção humana e discute os resultados dessa aplicação.
CI-3	A intervenção do usuário ocorre durante a criação dos grupos, levando a resultados necessariamente alcançados por conta dessa intervenção.

Fonte: Thais Rodrigues Neubauer, 2020

Tabela 11 – Critérios de Exclusão (CE)

CE-1	O estudo foi publicado antes de 2007.
CE-2	O estudo não está totalmente disponível em língua inglesa.
CE-3	O estudo não está primariamente relacionado a Ciência da Computação, Sistemas de Informação, Engenharia or áreas fortemente relacionadas a essas.
CE-4	O estudo não é um trabalho científico que passou pr revisão por pares.
CE-5	O estudo não é primário.
EC-6	O estudo é uma versão preliminar de um estudo posterior e de maior relevância.

Fonte: Thais Rodrigues Neubauer, 2020

D. Extração de dados e síntese de resultados: os trabalhos primários selecionados foram lidos na íntegra. A leitura foi guiada pelas questões de pesquisa levantadas e os trechos dos trabalhos que ajudavam a responder às questões foram selecionados. O conjunto de trechos referentes a cada questão apoiou a elaboração das respostas a elas e a reunião dos pontos fortes e fracos encontrados para agrupamento interativo.

E. Critérios de qualidade: o conjunto de critérios de qualidade (CQ) determinado é apresentado na tabela 12. Eles avaliaram os estudos primários selecionados em termos da qualidade da pesquisa em si e do relato Dyba, Dingsoyr e Hanssen (2007) e foram aplicados considerando que cada critério podia ser: *não atendido*, *atendido parcialmente* e *atendido totalmente*. Os critérios de qualidade não foram utilizados como forma adicional de exclusão, pois é argumentado em Dyba, Dingsoyr e Hanssen (2007) que não há consenso sobre limitar a quantidade estudos primários em uma revisão de literatura levando em conta a qualidade dos estudos.

Tabela 12 – Critérios de qualidade (CQ), baseados em Dyba, Dingsoyr e Hanssen (2007)

Qualidade do relato dos resultados de pesquisa
CQ-01 O estudo apresenta sistematicamente motivação, objetivos, método de pesquisa, resultados, análises, limitações e comparação com outros trabalhos.
Qualidade de pesquisa
Rigor: a abordagem é completa e adequada para resolver o problema?
CQ-02 Tomadas de decisão e escolhas feitas no estudo (por exemplo, escolha de algoritmos de agrupamento, escolha de como incluir supervisão de especialistas etc.) são explícitas e justificadas.
CQ-03 Os resultados alcançados são avaliados usando métricas de avaliação da qualidade (ou seja, índices de validação de grupos ou índices de qualidade de aprendizado supervisionado).
CQ-04 Os resultados alcançados são comparados com outras abordagens de agrupamento.
CQ-05 A validade e a significância dos resultados obtidos são atestadas estatisticamente.
Credibilidade: o método é reproduzível e verificável e os resultados são generalizáveis?
CQ-06 Os experimentos da abordagem proposta são conduzidos com mais de um conjunto de dados para evitar a dependência do contexto de um conjunto de dados específico.
CQ-07 O estudo é reproduzível, ou seja, as etapas e os recursos utilizados são descritos em detalhes e acessíveis quando relevantes (por exemplo, código, dados brutos, dados resumidos etc.).
Relevância: os resultados são úteis para a comunidade científica e a indústria de software?
CQ-08 As motivações para a aplicação de agrupamento interativo são explícitas, ou seja, a aplicação do agrupamento com a inclusão da supervisão de especialistas é justificada.
CQ-09 A qualidade dos resultados do agrupamento é discutida considerando a inclusão da supervisão de especialistas, ou seja, os resultados do agrupamento interativo e do agrupamento clássico são comparados.
CQ-10 O estudo discute se o custo-benefício da inclusão de supervisão por especialistas justifica tal inclusão.

Fonte: Thais Rodrigues Neubauer, 2020

Apêndice B – Resultados complementares

Neste apêndice, são apresentados os resultados obtidos com as execuções do experimento do cenário 2, apresentado na subseção 3.2.2. A elaboração desse experimento foi detalhada e seus resultados foram analisados na referida subseção, mas devido a semelhança a outros resultados comentados na mesma, somente alguns deles foram incluídos.

Na figura 34, são apresentados os acertos dos resultados obtidos, calculados da mesma forma como explicado para o cenário 1. Em (a) são apresentados os acertos e a quantidade de dados envolvidos nas restrições em cada execução e em (b), o aumento de acertos aprendidos por restrição adicionada.

Figura 34 – Resultado de agrupamentos do log sintético do cenário 2

(a) Acertos e quantidade de dados envolvidos

(b) Aumento de acertos aprendidos por restrição adicionada

Fonte: Thais Rodrigues Neubauer, 2020

Na figura 35, são apresentadas as variações na quantidade de acertos referentes a repetição de algumas das execuções por 10 vezes.

Figura 35 – Variação de resultado dos agrupamentos log sintético do cenário 2, em 10 repetições das execuções indicadas

Por fim, na figura 36, são apresentados os valores obtidos nas execuções deste cenário para as métricas de qualidade de modelo processos.

Figura 36 – Medidas de qualidade de modelo de processo para os resultados do log sintético do cenário 2

Fonte: Thais Rodrigues Neubauer, 2020

Apêndice C – Questionário - interação 1

Gerenciamento e predição de tempo de resolução de incidentes

Este questionário faz parte de um procedimento referente a uma pesquisa sobre mineração de processos. Ao respondê-lo, você estará nos ajudando no desenvolvimento da pesquisa.

O questionário é composto por quatro partes.

Em cada uma das três primeiras partes, você receberá uma tarefa e deverá fazer algumas escolhas para resolvê-la. Gostaríamos que você se sentisse envolvido pelas tarefas de forma a resolvê-las da melhor maneira possível. Então, escolha um momento em que você esteja motivado a participar desta pesquisa. Acreditamos que você não usará mais do que 20 minutos para resolver as três tarefas.

Na quarta parte, serão feitas dez perguntas sobre você e sobre sua experiência profissional. Algumas não são obrigatórias, mas, se você responder a todas elas, você nos ajudará ainda mais.

Agradecemos a sua colaboração e nos colocamos à sua disposição para conversar mais sobre mineração de processos, caso você tenha interesse.

Respeitosamente,
Thais Rodrigues Neubauer e Sarajane Marques Peres.

* Required

Tarefa 1: Projetando um sistema de gerenciamento de incidentes e um sistema preditor de tempo de resolução de incidentes

Considere que você precisa projetar um sistema de gerenciamento de incidentes, no qual estarão armazenadas informações que podem também ajudar a projetar um sistema preditor de tempo de resolução desses incidentes. Um preditor de tempo de resolução é um sistema capaz de, em qualquer momento, estimar quanto tempo falta para que um incidente seja resolvido.

Ao projetar os sistemas, considere que:

- o gerenciador de incidentes está sendo projetado para uma empresa que oferece suporte a serviços baseados em estruturas de hardware e software, por exemplo, uma empresa que oferece serviços de telecomunicações;
- o processo de gerenciamento de incidentes seguido por esta empresa está suportado pelo framework ITIL;
- o preditor de tempo de resolução poderá ser acionado em qualquer momento do ciclo de vida de um incidente;
- o ciclo de vida de um incidente passa por estados como: novo, ativo, aguardando informação, resolvido, fechado etc.

Os sistemas que você está projetando deverão contar com um banco de dados com informações sobre cada incidente, sobre o seu processo de gerenciamento e sobre o contexto no qual os incidentes ocorrem, e também deverá contar com um sistema de auditoria que armazenará (em um log) todas as operações efetuadas durante o ciclo de vida de todos os incidentes, desde as suas criações até os seus fechamentos. Todas as informações necessárias para esses sistemas serão organizadas na forma de atributos que descrevem características dos incidentes e das operações realizadas sobre os incidentes, durante todo o seu ciclo de vida, considerando as diferentes ações pertinentes ao gerenciamento deles.

Sabendo que você também será o responsável por projetar o preditor de tempo de resolução, indique os cinco atributos que não podem faltar no banco de dados ou no log de auditoria para que a predição de resolução dos incidentes tenha grandes chances de ser eficaz.

Como "atributos", entenda, por exemplo as colunas de uma tabela em um banco de dados. Se o contexto de gerenciamento de clientes estivesse sendo usado, atributos poderiam ser CPF, nome, endereço, data de nascimento etc.

Como "predição eficaz", considere que:

- seu sistema deve ser capaz de fazer predições acuradas, ou seja, dizer quanto tempo falta para o tratamento do incidente ser finalizado com uma margem de erro pequena, digamos, na ordem de poucos minutos de erro para incidentes cuja resolução consome algumas horas;
- seu sistema deve ser capaz de fazer predições acuradas o mais cedo possível, ou seja, o quanto antes uma predição acurada puder ser feita considerando o ciclo de vida do incidente, mais eficaz será o sistema.

Por favor, ordene os atributos escolhidos por importância, do mais importante para o menos importante.

1. Primeiro atributo (o mais importante): *

2. Segundo atributo:

3. Terceiro atributo:

4. Quarto atributo:

5. Quinto atributo:

Tarefa 2: Usando um sistema de gerenciamento de incidentes existente para projetar um sistema preditor de tempo de resolução de incidentes

Agora, considere que já existe um sistema de gerenciamento de incidentes e que nele são armazenados 29 atributos conforme listados abaixo:

- [1] identificador do cliente -- identificador do usuário afetado pelo incidente
- [2] localização -- identificador do local afetado pelo incidente
- [3] identificador do reclamante -- identificador do usuário que relatou o incidente para o atendente
- [4] identificador do atendente -- identificador do usuário que registrou o incidente na plataforma de gerenciamento
- [5] identificador do responsável -- identificador de usuário indicado como responsável pelo incidente
- [6] grupo de associação -- identificador do grupo de suporte que ficou responsável pelo tratamento do incidente
- [7] tipo de contato -- forma de contato usada para relatar o incidente
- [8] data-hora de abertura -- data e hora de abertura do incidente na plataforma de gerenciamento
- [9] categoria -- descrição de primeiro nível do serviço afetado
- [10] subcategoria -- descrição de segundo nível do serviço afetado - relacionado ao primeiro nível (categoria)
- [11] identificador de item -- identifica o item afetado pelo incidente
- [12] identificador do vendedor -- identifica o vendedor relacionado ao gerenciamento do incidente
- [13] identificador do problema -- identifica o problema associado ao incidente
- [14] requisição de mudança -- identifica a requisição de mudança associada com o incidente
- [15] sintoma -- descrição sobre a percepção do usuário sobre a disponibilidade do serviço
- [16] impacto -- descrição do impacto causado pelo incidente. Os valores possíveis são: alto, médio e baixo
- [17] urgência -- descrição da urgência solicitada pelo usuário para a resolução do incidente. Os valores possíveis são: alta, média e baixa
- [18] prioridade -- valor de prioridade calculada pelo sistema com base nos atributos impacto e urgência. Os valores possíveis são: crítica, alta, média e baixa
- [19] confirmação de prioridade -- indicação sobre a realização de checagem dupla no registro da prioridade do incidente
- [20] estado do incidente -- atributo que pode assumir valores referentes a oito estados pertinentes ao ciclo de vida de um incidente no processo de gerenciamento. Os estados possíveis são: novo, resolvido, fechado, ativo, aguardando informação do usuário, aguardando informação sobre o problema, aguardando informação do vendedor, e aguardando informação sobre a evidência
- [21] contagem de reassociações -- número de vezes que o incidente mudou de grupo de analistas de suporte responsáveis pelo seu tratamento
- [22] contagem de reaberturas -- número de vezes que a resolução do incidente foi rejeitada pelo usuário que é afetado pelo incidente
- [23] contagem de alterações -- número de vezes que o incidente passou por uma alteração
- [24] usuário de alteração -- identificador do usuário que realizou uma determinada alteração para o incidente
- [25] conhecimento a priori -- indicação se alguma documentação existente no sistema foi usada como suporte para resolver o incidente
- [26] notify -- indicação sobre notificações geradas para um incidente (se foram geradas ou não, e se sim, quais são elas)
- [27] controle de SLA -- atributo que indica se o SLA (Service Level Agreement) para resolução do incidente foi excedido ou não
- [28] data de criação -- horário e data em que um incidente foi criado no sistema de gerenciamento
- [29] data de alteração -- horário e data em que um incidente sofreu uma alteração no sistema de gerenciamento

Obs.: Esses atributos são baseados naqueles usados pela plataforma ServiceNow. Se você tiver experiência com contextos nos quais esta plataforma é usada, considere sua experiência para resolver esta tarefa.

Quais dos atributos acima você escolheria como base para projetar um preditor de tempo de resolução de incidentes?

Escolha até cinco atributos e os ordene do mais importante para o menos importante. Use a barra de rolagem para acessar as opções que não aparecem na visão inicial do quadro abaixo. A barra de rolagem talvez esteja oculta até que você posicione o mouse na parte inferior externa do quadro.

6. *

Mark only one oval per row.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Atributo 1 - mais importante	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Atributo 2	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Atributo 3	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Atributo 4	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Atributo 5 - menos importante	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

7. Você acrescentaria atributos à lista de 29 atributos apresentada acima? Se sim, informe quais seriam esses atributos.

Tarefa 3: Avaliando escolhas de atributos para projetar um sistema preditor de tempo de resolução de incidentes

Nessa última tarefa, você deve avaliar algumas escolhas que já embasaram projetos preditores de tempo de resolução de incidentes. As escolhas estão organizadas em três conjuntos diferentes. Cada conjunto deu origem a um sistema preditor, ou seja, com base nas informações que esses atributos fornecem, o sistema desenvolve o raciocínio que permite estimar quanto tempo ainda falta para o incidente ser resolvido. Os conjuntos de atributos são listados abaixo.

- Conjunto 1: identificador do cliente, identificador do responsável.

- Conjunto 2: estado do incidente, localização.

Com base nos três conjuntos de atributos apresentados, responda as três perguntas abaixo:

8. Você considera o Conjunto 1 uma boa escolha de atributos para suportar a predição de tempo de resolução de incidentes? *

Mark only one oval.

- ☐ Sim
☐ Não

9. Justifique a sua resposta.

10. Você considera o Conjunto 2 uma boa escolha de atributos para suportar a predição de tempo de resolução de incidentes? *

Mark only one oval.

- ☐ Sim
☐ Não

11. Justifique a sua resposta.

12. Você considera o Conjunto 3 uma boa escolha de atributos para suportar a predição de tempo de resolução de incidentes? *

Mark only one oval.

- ☐ Sim
☐ Não

13. Justifique a sua resposta.

Conte-nos mais sobre você...

As informações solicitadas aqui tem o objetivo de enriquecer as respostas que você já nos deu. Além de informações gerais, nós solicitamos o seu nome para o caso de identificarmos a necessidade de voltar a lhe pedir alguma informação. Entretanto, seu nome e seu cargo não são respostas obrigatórias. Se você não se sente confortável em fornecê-las, não há problemas.

14. Qual seu nome completo?

15. Qual seu cargo atual?

16. Há quanto tempo você trabalha com gerenciamento de processos? *

Mark only one oval.

- ☐ Eu não trabalho com gerenciamento de processos
☐ Menos de um ano
☐ De um a três anos
☐ Mais de três anos

17. Como você classifica seu nível de conhecimento a respeito de gerenciamento de processos? *

Mark only one oval.

- ☐ Eu não tenho conhecimento sobre gerenciamento de processos
☐ Conhecimento básico
☐ Conhecimento intermediário
☐ Conhecimento avançado

18. Há quanto tempo você trabalha com gerenciamento de processos de incidentes? **Mark only one oval.*

- ☐ Eu não trabalho com gerenciamento de processos de incidentes
- ☐ Menos de um ano
- ☐ De um a três anos
- ☐ Mais de três anos

19. Como você classifica seu nível de conhecimento a respeito de gerenciamento de processos de incidentes? **Mark only one oval.*

- ☐ Eu não tenho conhecimento sobre gerenciamento de processos de incidentes
- ☐ Conhecimento básico
- ☐ Conhecimento intermediário
- ☐ Conhecimento avançado

20. Há quanto tempo você trabalha com o framework ITIL? **Mark only one oval.*

- ☐ Eu não trabalho com o framework ITIL
- ☐ Menos de um ano
- ☐ De um a três anos
- ☐ Mais de três anos

21. Como você classifica seu nível de conhecimento a respeito do framework ITIL? **Mark only one oval.*

- ☐ Eu não tenho conhecimento sobre o framework ITIL
- ☐ Conhecimento básico
- ☐ Conhecimento intermediário
- ☐ Conhecimento avançado

22. Há quanto tempo você trabalha com a plataforma ServiceNow? **Mark only one oval.*

- ☐ Eu não trabalho com a plataforma ServiceNow
- ☐ Menos de um ano
- ☐ De um a três anos
- ☐ Mais de três anos

23. Como você classifica seu conhecimento sobre o suporte a processos de gerenciamento de incidentes oferecido pela plataforma ServiceNow? **Mark only one oval.*

- ☐ Não tenho conhecimento sobre a plataforma ServiceNow
- ☐ Conhecimento básico
- ☐ Conhecimento intermediário
- ☐ Conhecimento avançado

Obrigadas por sua colaboração!**24. Use este espaço para sugestões, comentários ou informações adicionais.**

Caso você queira saber mais sobre esta pesquisa, entre em contato conosco:

- Thaís Rodrigues Neubauer: thais.neubauer@usp.br
- Sarajane Marques Peres: sarajane@usp.br

E não esqueça de submeter a sua resposta!! :-)

Apêndice D – Questionário - interação 2

Predição de tempo de resolução de incidentes

Este questionário faz parte de uma pesquisa sobre mineração de processos. Ao respondê-lo, você estará nos ajudando no desenvolvimento dessa pesquisa. Agradecemos a sua colaboração e nos colocamos à sua disposição para conversar mais sobre mineração de processos, caso você tenha interesse.

Respeitosamente,
Thais Rodrigues Neubauer, Profa. Dra. Sarajane Marques Peres e Prof. Dr. Marcelo Fantinato.

*Obrigatório

1. Endereço de e-mail *

2. Qual seu nome completo? *

Contexto

Considere um “sistema de gerenciamento de incidentes” que armazena informações que podem ser usadas para ajudar a projetar um “sistema preditor de tempo de resolução de incidentes”. Um sistema preditor de tempo de resolução de incidentes é capaz de, em qualquer momento, estimar quanto tempo falta para que um incidente seja resolvido.

Considere ainda que:

- 1) o sistema de gerenciamento de incidentes é usado por uma empresa que oferece suporte a serviços baseados em hardware e software, por exemplo, uma empresa que oferece serviços de telecomunicações;
- 2) o processo de gerenciamento de incidentes seguido por essa empresa é baseado no framework ITIL;
- 3) o ciclo de vida de um incidente passa por estados como: novo, ativo, aguardando informação, resolvido, fechado etc;
- 4) o sistema preditor de tempo de resolução de incidentes poderá ser acionado em qualquer momento do ciclo de vida de um incidente.

Nosso objetivo:

Criar um “sistema preditor de tempo de resolução de incidentes” que realiza a predição com base no conhecimento derivado do histórico de gerenciamento dos incidentes já tratados na empresa.

Nosso problema:

Usar todo o histórico de incidentes tratados na empresa propiciará um sistema preditor com baixa precisão, já que existe uma grande variabilidade nas características dos incidentes e de seu processo de resolução.

Nossa solução:

Usar um algoritmo que agrupa incidentes similares com base no ciclo de vida dos incidentes e em algumas informações dos incidentes. Essas informações não incluem o tempo de resolução dos incidentes, pois essa informação não estaria ainda disponível em um ambiente real de produção e é justamente o que queremos prever. Temos a suposição que podemos construir “um preditor de tempo de resolução por grupo de incidentes”, aumentando a precisão da predição, visto que cada preditor trabalhará em um grupo de incidentes similares, com processo de resolução também similares.

Sua ajuda:

O nosso algoritmo produz grupos de incidentes similares que talvez possam ser melhorados por meio da análise de um especialista. Por isso, pedimos que você analise os grupos que obtivemos com o algoritmo e nos indique possíveis melhorias e quais são elas.

Para orientar sua análise, elaboramos quatro tarefas para você realizar. Gostaríamos que você se sentisse envolvido pelas tarefas de forma a resolvê-las da melhor maneira possível. Então, escolha um momento em que você esteja motivado a participar desta pesquisa. Acreditamos que a resolução de todas as tarefas levará cerca de 40 minutos.

Para lhe auxiliar, em algumas tarefas, um exemplo da resolução esperada será apresentado usando um contexto ilustrativo.

Antes de começarmos...

3. Para que possamos estimar o tempo que você levou para realizar as tarefas propostas, informe abaixo o horário em que você está iniciando as tarefas: *

Exemplo: 08h30

Tarefa 1: Melhorando a caracterização dos grupos

Contexto ilustrativo

Suponha que o nosso algoritmo criou os seguintes grupos de pratos servidos em um restaurante, com base nas informações de quem os prepara e equipamentos usados no preparo. E suponha que o objetivo final dessa análise é reorganizar os preços praticados no restaurante.

Grupo 1	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha	Fogão
Cozinheiro	Forno
Auxiliar 1	Processador
Auxiliar 2	Geladeira

Grupo 2	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha	Fogão
Churrasqueiro	Forno
Auxiliar 1	Congelador
Auxiliar 2	Maçarico

Grupo 3	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Cozinheiro	Forno
Auxiliar 1	Maçarico
Auxiliar 2	Congelador
	Batedeira

Grupo 4	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Cozinheiro	Forno
Churrasqueiro	Churrasqueira
Auxiliar 1	Processador
	Congelador

Agora, imagine que, ao mostrarmos esses quatro grupos de pratos para o chefe de cozinha do restaurante, tenhamos recebido as seguintes observações dele:

- Observação 1: todos os pratos que envolvem a mim (chefe de cozinha) têm custo maior do que os pratos que não precisam do meu envolvimento. Então, todos os pratos que precisam do meu trabalho como chefe de cozinha devem ter o preço bem maior que dos demais.

- Observação 2: os pratos que envolvem o churrasqueiro e precisam da churrasqueira possuem um valor agregado diferenciado dos pratos que são congelados (ou seja, que envolvem o congelador). Geralmente, os pratos congelados são bem mais baratos.

Repare que as observações do chefe de cozinha envolveram a mesma característica de mais de um grupo ou diferentes características de um mesmo grupo, como destacado abaixo:

Grupo 1	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha	Fogão
Cozinheiro	Forno
Auxiliar 1	Processador
Auxiliar 2	Geladeira

Grupo 2	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha	Fogão
Churrasqueiro	Forno
Auxiliar 1	Congelador
Auxiliar 2	Maçarico

Grupo 3	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Cozinheiro	Forno
Auxiliar 1	Maçarico
Auxiliar 2	Congelador
	Batedeira

Grupo 4	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Cozinheiro	Forno
Churrasqueiro	Churrasqueira
Auxiliar 1	Processador
	Congelador

As observações do chefe de cozinha nos dizem que precisamos “ensinar” o algoritmo a fazer duas coisas para melhorar a qualidade dos grupos:

- 1) garantir que todos os pratos preparados pelo chefe de cozinha fiquem em um mesmo grupo;
- 2) garantir que pratos que envolvam churrasqueira e pratos que envolvam congelador sejam colocados em grupos diferentes.

Agora, transfira essa estratégia de análise para o caso da construção de um sistema de predição de tempo de resolução de incidentes e clique em “NEXT” para resolver a tarefa.

Tarefa 1: Melhorando a caracterização dos grupos

O algoritmo criou cinco grupos de incidentes usando informação sobre: estado, prioridade e categoria. Os grupos estão apresentados no quadro abaixo. Para interpretar o quadro, considere que:

- Os “Estados” que os incidentes assumiram durante seu ciclo de vida são: new, active, resolved, closed, awaiting evidence, awaiting problem, awaiting user info e awaiting vendor. No quadro, a ordem de apresentação desses estados não necessariamente corresponde à ordem de ocorrência deles no ciclo de vida dos incidentes. Um estado é apresentado em um grupo se pelo menos um incidente no grupo passou por esse estado, ou seja, nem todos os incidentes do grupo passaram por todos os estados apresentados no grupo. Os estados são apontados uma única vez em cada grupo, mesmo que um incidente tenha passado por um estado mais de uma vez.
- A “Prioridade” que os incidentes receberam quando foram cadastrados no sistema de gerenciamento de incidentes são: 1 - Critical, 2 - High, 3 - Moderate e 4 – Low. No quadro, a informação sobre prioridade é apresentada nessa mesma ordem.
- A “Categoria” do objeto sobre o qual o incidente ocorreu. Existem 36 categorias. Nós apresentamos no quadro abaixo apenas as dez categorias mais frequentes nos incidentes de cada grupo. A informação sobre categoria é apresentada em ordem alfabética.

Todas as outras informações sobre os incidentes foram desprezadas pelo algoritmo, tais como: sub-categoria, localidade, sintoma, impacto e urgência.

Os grupos que o algoritmo criou estão mostrados abaixo:

Grupo 1		
Incident_state	Priority	Category
New	1 - Critical	Desktop
Active	2 - High	Internet
Resolved	3 - Moderate	Notebook
Closed	4 - Low	Passwords and access
Awaiting Evidence		Printer
Awaiting Problem		SAP ECC
Awaiting User Info		SAP ECC EN
Awaiting Vendor		Server
		Software - industry
		Telephony

Grupo 2		
Incident_state	Priority	Category
New	1 - Critical	Others
Active	2 - High	Passwords and access
Resolved	3 - Moderate	Remote access
Closed	4 - Low	SAP
Awaiting Evidence		SAP CRM
Awaiting Problem		SAP ECC
Awaiting User Info		SAP ECC EN
Awaiting Vendor		SAP GRC
		Software
		Software - industry

Grupo 3		
Incident_state	Priority	Category
New	1 - Critical	Peripheral
Active	2 - High	Radio frequency antenna
Resolved	3 - Moderate	Software - industry
Closed	4 - Low	SAP
Awaiting Evidence		Telephony
Awaiting Problem		Passwords and access
Awaiting User Info		Internet
Awaiting Vendor		Remote access
		Infrastructure
		Network

Grupo 4		
Incident_state	Priority	Category
New	1 - Critical	Infrastructure
Active	2 - High	Internet
Resolved	3 - Moderate	Network
Closed	4 - Low	Passwords and access
		Peripheral
Awaiting Problem		Radio frequency antenna
Awaiting User Info		Remote access
Awaiting Vendor		SAP
		Software - industry
		Telephony

Grupo 5		
Incident_state	Priority	Category
New	1 - Critical	Desktop
Active	2 - High	Internet
Resolved	3 - Moderate	Office pack
Closed	4 - Low	Passwords and access
		Remote access
Awaiting Problem		SAP
Awaiting User Info		SAP CRM
Awaiting Vendor		SAP ECC EN
		Software
		Software - industry

Sua tarefa:

Considerando as informações que descrevem os incidentes dentro desses grupos, faça como o “chefe de cozinha” fez com suas observações e forneça pelo menos três observações para organizar melhor os incidentes e proporcionar uma base mais precisa para criar os nossos preditores.

Lembrando que para melhorar nossos preditores, os grupos deveriam ser compostos por incidentes com perfis mais similares em termos de tempo de resolução.

Você pode apresentar mais que três observações se achar relevante. Ordene suas observações por importância: a mais importante é a primeira.

4. Observação 1: *

5. Observação 2: *

6. Observação 3: *

7. Outras observações:

Tarefa 2: Melhorando a organização dos incidentes nos grupos

Considere aqueles mesmos cinco grupos de incidentes, porém agora com informações mais detalhadas apenas sobre os “estados” pelos quais os incidentes passaram durante seu ciclo de vida.

Para ajudar a interpretar esses grupos, observe as seguintes características no gráfico abaixo para o “Grupo 4”:

- nenhum incidente no Grupo 4 passou pelo estado “Awaiting Evidence”;
- todos os incidentes no Grupo 4 passaram pelos estados “Resolved” e “Closed”;
- muitos incidentes (porém não todos) no Grupo 4 passaram pelos estado “New” e “Active” - (embora não seja esperado que existam incidentes que não passem, por exemplo, pelo estado “New”, é admissível que o sistema que registra as informações sobre os incidentes não seja totalmente fiel ao que se espera);
- um pouco de incidentes no Grupo 4 passou pelo estado “Awaiting User Info”;
- muito poucos incidentes no Grupo 4 passaram pelos estados “Awaiting Problem” e “Awaiting Vendor”.

Grupo 1

Grupo 2

Grupo 3

Grupo 4

Grupo 5

Para uma visualização melhorada desses gráficos, acesse os links abaixo:

- Grupo 1 - https://temporesolucaoincidentes.netlify.com/grupo%201/incident_state.html
- Grupo 2 - https://temporesolucaoincidentes.netlify.com/grupo%202/incident_state.html
- Grupo 3 - https://temporesolucaoincidentes.netlify.com/grupo%203/incident_state.html
- Grupo 4 - https://temporesolucaoincidentes.netlify.com/grupo%204/incident_state.html
- Grupo 5 - https://temporesolucaoincidentes.netlify.com/grupo%205/incident_state.html

Lembrando que grupos deveriam ser compostos por incidentes com perfis similares em termos de tempo de resolução.

8. Você acha que, em relação ao número de grupos:

Marcar apenas uma oval.

☐

é necessário mais que cinco grupos para separar os incidentes de forma mais adequada em relação à similaridade de tempo de resolução;

☐

cinco grupos é muito; o melhor seria ter menos grupos, pois de fato os incidentes são bastante similares;

☐

cinco grupos é o número ideal para acomodar os incidentes.

9. Explique suas motivações para a escolha acima.

10. Avalie as observações abaixo em relação ao quão relevante elas podem ser para melhorar os grupos de incidentes, considerando a similaridade de tempo de resolução. Atribua “0” para observações irrelevantes, e gradue aquelas que você acha relevante (“1” para relevância baixa; “2” para relevância média; “3” para relevância alta). *

Marcar apenas uma oval por linha.

	0	1	2	3
(A) Todos os incidentes que passaram por qualquer um dos estados “Awaiting ...” deveriam ficar juntos em um único grupo	☐	☐	☐	☐
(B) Incidentes que não passaram pelo estado “Active” e não passaram por nenhum estado “Awaiting ...” deveriam ficar juntos em um único grupo e separados dos demais incidentes.	☐	☐	☐	☐
(C) Incidentes que passaram pelo estado “Awaiting User Info” deveriam ficar em um grupo isolado dos demais incidentes.	☐	☐	☐	☐

11. Explique suas motivações para a escolha acima. Neste espaço, você também pode acrescentar outras regras que achar necessárias para melhorar a distribuição dos incidentes nos grupos. *

Tarefa 3: Analisando a qualidade de grupos individualmente

Contexto ilustrativo

Voltando para o nosso exemplo ilustrativo de pratos servidos em um restaurante. Temos abaixo os gráficos que informam como um dos quatro grupos criados pelo algoritmo é descrito em termos das características dos pratos (ou seja, os profissionais e equipamentos envolvidos em seu preparo). Um prato pode ou não incluir mais de um profissional e/ou equipamento em seu preparo.

No grupo apresentado abaixo, nenhum prato exigiu o envolvimento do chefe de cozinha no seu preparo e nenhum prato envolveu o uso de fogão, batadeira, maçarico e geladeira. Ainda é possível observar, por exemplo, que 60% dos pratos incluídos neste grupo exigiram o trabalho do churrasqueiro e 12% deles envolveu o uso de processador.

Mostramos esses gráficos ao chefe de cozinha e perguntamos a ele quais das características que estão nesse grupo (que foi criado pelo algoritmo) ele achava que funcionariam melhor se estivessem separadas (aparecendo em grupos diferentes e não no mesmo grupo). Lembrando que nesse caso o objetivo principal era agrupar pratos com similaridade em termos de preços.

Ele nos deu algumas sugestões para melhorar a composição do grupo:

1) Os pratos que envolvem tanto o “Auxiliar 1” quanto a “Churrasqueira” são mais complexos de serem preparados e, portanto, devem ter preço mais alto do que os que não envolvem esses recursos no preparo. Seria razoável então que esses pratos ficassem separados dos demais que estão no grupo.

2) O “Auxiliar 2” é um profissional alocado para preparar pratos de baixa complexidade e baixo custo. Então, pratos que envolvem o “Auxiliar 1” ou a “Churrasqueira” não deveriam ficar no mesmo grupo dos pratos que envolvem o “Auxiliar 2”. Ou seja, o algoritmo não deveria permitir que em um mesmo grupo ocorresse a combinação do conjunto “Auxiliar 1”+“Churrasqueira” com o “Auxiliar 2”.

Então, na sequência, nós mostramos ao chefe de cozinha a variação de preços dos pratos incluídos no grupo (veja na figura abaixo que os preços estão variando entre R\$15,00 e R\$98,00) e ressaltamos que a faixa de preços que o restaurante pode praticar é de R\$10,00 a R\$120,00. Pedimos que ele analisasse novamente a situação para verificar se ele poderia nos dar mais algumas sugestões.

O chefe de cozinha analisou as características dos grupos e reafirmou que os pratos que envolve o “Auxiliar 1” ou a “Churrasqueira” são de mais alto custo do que os demais pratos e, portanto, deveriam ser alocados em um grupo particular (exclusivamente formado por tais pratos).

Agora, transfira essa estratégia de análise para o caso da construção de um sistema de predição de tempo de resolução de incidentes e clique em “NEXT” para resolver a tarefa.

Tarefa 3: Analisando a qualidade de grupos individualmente

Voltamos para o caso do “sistema de gerenciador de incidentes”. Analise as informações sobre o Grupo 1 de incidentes, descrito em termos de frequência por “categoria”, “estado” e “prioridade”.

Grupo 1

Grupo 1 - category

Grupo 1 - incident_state

Grupo 1 - priority

Para uma visualização melhorada desses gráficos, acesse os links abaixo:

- Category - <https://temporesolucaoincidentes.netlify.com/grupo%201/category.html>

- State - https://temporesolucaoincidentes.netlify.com/grupo%201/incident_state.html
 - Priority - <https://temporesolucaoincidentes.netlify.com/grupo%201/priority.html>
-

Quais das características que estão nesse grupo (que foi criado pelo algoritmo), você acha que funcionariam melhor se estivessem separadas (aparecendo em grupos diferentes e não no mesmo grupo)? Lembrando sempre que o objetivo principal é agrupar incidentes que possuam similaridade em termos de tempo de resolução de incidente.

Você pode montar sugestões parecidas com esses formatos:

- 1) prioridade "X" + estado "Y" deveriam ficar sempre separados da categoria "Z".
- 2) estado "A" deveria ficar sempre separado da categoria "B".

Uma outra forma de ler essas sugestões é:

- 1) prioridade "X" + estado "Y" deveriam estar em um grupo no qual a categoria "Z" não está presente.
- 2) estado "A" deveria estar em um grupo no qual não aparece a categoria "B".

12. Sugestão 1 (a mais importante): *

13. Sugestão 2:

14. Outras sugestões (uma por linha, enumeradas):

Agora, considerando a informação sobre o tempo de resolução de incidentes. Observe no gráfico abaixo que os tempos de resolução dos incidentes agrupados no Grupo 1 variaram de 0 dias a 341 dias (de uma faixa total de 0 a 341 presente no conjunto total de incidentes):

Tempo de resolução de incidente (em dias)

Para uma visualização melhorada desses gráficos, acesse o link abaixo:

- <https://temporesolucaoincidentes.netlify.com/grupo%201/tempo%20de%20resolucao>

Lembre-se que os incidentes foram agrupados considerando apenas “prioridade”, “estados” e “categoria”, esperando que os incidentes fossem indiretamente agrupados com “tempo de resolução” potencialmente similares.

15. Ao observar os tempos de resolução dos incidentes deste grupo, você identifica problemas na alocação desses incidentes no mesmo grupo?

Marcar apenas uma oval.

- ☐ Não, eu não consigo ver nenhum problema com a forma como esse grupo foi criado.
- ☐ Sim, eu acho que a variabilidade de tempo de resolução de incidentes para este grupo está muito grande e vou apontar abaixo algumas sugestões para diminuir essa variabilidade, considerando as informações que o algoritmo que faz os grupos é capaz de usar: “estado”, “prioridade” e “categoria”.

16. Sugestão 1 (a mais importante):

17. Sugestão 2:

18. Outras sugestões (uma em cada linha, enumeradas):

Tarefa 3: Analisando a qualidade de grupos individualmente

Voltamos para o caso do “sistema de gerenciador de incidentes”. Analise as informações sobre o Grupo 2 de incidentes, descrito em termos de frequência por “categoria”, “estado” e “prioridade”.

Grupo 2

Para uma visualização melhorada desses gráficos, acesse os links abaixo:

- Category - <https://temporesolucaoincidentes.netlify.com/grupo%202/category.html>
- State - https://temporesolucaoincidentes.netlify.com/grupo%202/incident_state.html
- Priority - <https://temporesolucaoincidentes.netlify.com/grupo%202/priority.html>

Quais das características que estão nesse grupo (que foi criado pelo algoritmo), você acha que funcionariam melhor se estivessem separadas (aparecendo em grupos diferentes e não no mesmo grupo)? Lembrando sempre que o objetivo principal é agrupar incidentes que possuam similaridade em termos de tempo de resolução de incidente.

Você pode montar sugestões parecidas com esses formatos:

- 1) prioridade “X” + estado “Y” deveriam ficar sempre separados da categoria “Z”.
- 2) estado “A” deveria ficar sempre separado da categoria “B”.

Uma outra forma de ler essas sugestões é:

- 1) prioridade “X” + estado “Y” deveriam estar em um grupo no qual a categoria “Z” não está presente.
- 2) estado “A” deveria estar em um grupo no qual não aparece a categoria “B”.

19. Sugestão 1 (a mais importante): *

20. Sugestão 2:

21. Outras sugestões (uma por linha, enumeradas):

Agora, considerando a informação sobre o tempo de resolução de incidentes. Observe no gráfico abaixo que os tempos de resolução dos incidentes agrupados no Grupo 2 variaram de 0 dias a 330 dias (de uma faixa total de 0 a 341 presente no conjunto total de incidentes):

Tempo de resolução de incidente (em dias)

Para uma visualização melhorada desses gráficos, acesse o link abaixo:

- <https://temporesolucaoincidentes.netlify.com/grupo%202/tempo%20de%20resolucao>

Lembre-se que os incidentes foram agrupados considerando apenas “prioridade”, “estados” e “categoria”, esperando que os incidentes fossem indiretamente agrupados com “tempo de resolução” potencialmente similares.

22. Ao observar os tempos de resolução dos incidentes deste grupo, você identifica problemas na alocação desses incidentes no mesmo grupo?

Marcar apenas uma oval.

- ☐ Não, eu não consigo ver nenhum problema com a forma como esse grupo foi criado.
- ☐ Sim, eu acho que a variabilidade de tempo de resolução de incidentes para este grupo está muito grande e vou apontar abaixo algumas sugestões para diminuir essa variabilidade, considerando as informações que o algoritmo que faz os grupos é capaz de usar: “estado”, “prioridade” e “categoria”.

23. Sugestão 1 (a mais importante):

24. Sugestão 2:

25. Outras sugestões (uma em cada linha, enumeradas):

Tarefa 4: Analisando o agrupamento geral

Agora você pode ver as informações dos cinco grupos juntos.

Grupo 1 - category

Grupo 1 - incident_state

Grupo 1 - priority

Tempo de resolução de incidente (em dias)

Grupo 2 - category

Grupo 2 - incident_state

Grupo 2 - priority

Tempo de resolução de incidente (em dias)

Grupo 3 - category

Grupo 3 - incident_state

Grupo 3 - priority

Tempo de resolução de incidente (em dias)

Grupo 4 - category

Grupo 4 - incident_state

Grupo 4 - priority

Tempo de resolução de incidente (em dias)

Grupo 5 - category

Grupo 5 - incident_state

Grupo 5 - priority

Tempo de resolução de incidente (em dias)

26. Considerando essa visualização geral dos incidentes em cada grupo e das variações dos tempos de resolução de incidentes para cada grupo, você acrescentaria ou alteraria alguma observação ou sugestão sobre a forma como os incidentes foram agrupados, pensando na predição de tempo de resolução de incidentes? *

Quase finalizando

27. Para que possamos estimar o tempo que você levou para realizar as tarefas propostas, informe abaixo o horário em que você está finalizando as tarefas: *

Exemplo: 08h30

Agradecemos a sua colaboração!

28. Use este espaço para sugestões, comentários ou informações adicionais.

Caso você queira saber mais sobre esta pesquisa, entre em contato conosco:

- Thais Rodrigues Neubauer: thais.neubauer@usp.br

E não esqueça de enviar a sua resposta!! :))

☐ Envie para mim uma cópia das minhas respostas.

Powered by

Apêndice E – Questionário - interação 2 - grupo de controle

Predição de tempo de resolução de incidentes

Este questionário faz parte de uma pesquisa sobre mineração de processos. Ao respondê-lo, você estará nos ajudando no desenvolvimento dessa pesquisa. Agradecemos a sua colaboração e nos colocamos à sua disposição para conversar mais sobre mineração de processos, caso você tenha interesse.

Respeitosamente,
Thais Rodrigues Neubauer, Profa. Dra. Sarajane Marques Peres e Prof. Dr. Marcelo Fantinato.

1. Email address *

2. Qual seu nome completo?

Contexto

Considere um "sistema de gerenciamento de incidentes" que armazena informações que podem ser usadas para ajudar a projetar um "sistema preditor de tempo de resolução de incidentes". Um sistema preditor de tempo de resolução de incidentes é capaz de, em qualquer momento, estimar quanto tempo falta para que um incidente seja resolvido.

Considere ainda que:

- 1) o sistema de gerenciamento de incidentes é usado por uma empresa que oferece suporte a serviços baseados em hardware e software, por exemplo, uma empresa que oferece serviços de telecomunicações;
- 2) o processo de gerenciamento de incidentes seguido por essa empresa é baseado no framework ITIL;
- 3) o ciclo de vida de um incidente passa por estados como: novo, ativo, aguardando informação, resolvido, fechado etc;
- 4) o sistema preditor de tempo de resolução de incidentes poderá ser acionado em qualquer momento do ciclo de vida de um incidente.

Nosso objetivo:

Criar um "sistema preditor de tempo de resolução de incidentes" que realiza a predição com base no conhecimento derivado do histórico de gerenciamento dos incidentes já tratados na empresa.

Nosso problema:

Usar todo o histórico de incidentes tratados na empresa propiciará um sistema preditor com baixa precisão, já que existe uma grande variabilidade nas características dos incidentes e de seu processo de resolução.

Nossa solução:

Usar um algoritmo que agrupa incidentes similares com base no ciclo de vida dos incidentes e em algumas informações dos incidentes. Essas informações não incluem o tempo de resolução dos incidentes, pois essa informação não estaria ainda disponível em um ambiente real de produção e é justamente o que queremos prever. Temos a suposição que podemos construir "um preditor de tempo de resolução por grupo de incidentes", aumentando a precisão da predição, visto que cada preditor trabalhará em um grupo de incidentes similares, com processo de resolução também similares.

Sua ajuda:

Antes de executar nosso algoritmo, gostaríamos de saber se, analisando os valores dos atributos que descrevem os incidentes, você nos daria algumas sugestões sobre atributos que caracterizam incidentes que deveriam ficar obrigatoriamente em um mesmo grupo e atributos que caracterizam incidentes que deveriam obrigatoriamente ser alocados em grupos diferentes.

Gostaríamos que você se sentisse envolvido pela tarefa proposta de forma a resolvê-la da melhor maneira possível. Então, escolha um momento em que você esteja motivado a participar desta pesquisa. Acreditamos que a resolução da tarefa levará cerca de 20 minutos.

Para lhe auxiliar, um exemplo da resolução esperada será apresentado usando um exemplo referente a um problema em outro domínio de aplicação.

Antes de começarmos...

3. Para que possamos estimar o tempo que você levou para realizar as tarefas propostas, informe abaixo o horário em que você está iniciando a tarefa:

Example: 8:30 AM

Exemplo

Imagine que sua tarefa seja agrupar os pratos servidos em um restaurante para possibilitar a reorganização de seus preços e que os atributos descritivos disponíveis dos pratos sejam: informação de

quem os prepara e equipamentos usados no preparo.

As listas dos possíveis valores de cada atributo descritivo são apresentadas abaixo:

- Profissionais envolvidos no preparo: Auxiliar 1, Auxiliar 2, Churrasqueiro, Cozinheiro e Chefe de cozinha, ordenados da menor para a maior experiência em gastronomia.
- Equipamentos necessários no preparo: Congelador, Geladeira, Fogão, Forno, Batedeira, Processador, Maçarico e Churrasqueira, ordenados do mais simples de ser utilizado ao mais complexo.

Suponha que, a partir dessas informações e conhecendo o contexto do restaurante em questão, os seguintes grupos de pratos tenham sido propostos:

Grupo 1	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha	Forno Batedeira Maçarico Geladeira

Grupo 2	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Chefe de cozinha Churrasqueiro Cozinheiro Auxiliar 2	Forno Batedeira Maçarico Churrasqueira

Grupo 3	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Auxiliar 1 Auxiliar 2	Congelador Fogão Forno Geladeira

Grupo 4	
Profissionais envolvidos no preparo	Equipamentos necessários no preparo
Auxiliar 1 Auxiliar 2	Congelador Forno

Lembrando sempre que o objetivo é agrupar pratos com preços semelhantes no mesmo grupo e manter pratos com preços muito diferentes em grupos separados. Suponha também que as principais observações sobre o contexto que justificam esse agrupamento e que devem ser obedecidas por qualquer mecanismo de agrupamento desses pratos sejam:

1. Todos os pratos preparados exclusivamente pelo "Chefe de cozinha" devem ficar separados (Grupo 1), pois possuem um valor agregado bem superior que os demais pratos do cardápio.
2. Os pratos que envolvem muitos profissionais ("Churrasqueiro", "Cozinheiro", "Chefe de cozinha" e "Auxiliar 2") devem ficar em um grupo separado (Grupo 2) dos grupos que envolvem somente o "Auxiliar 1" e o "Auxiliar 2" (Grupos 3 e 4) devido ao custo dos profissionais envolvidos.
3. Os pratos que necessitam somente do "Congelador" e do "Forno" para serem preparados devem ficar separados (Grupo 4) pois envolvem menos custos do que os demais pratos.

Agora, transfira essa estratégia de análise para o caso da construção de um sistema de predição de tempo de resolução de incidentes e clique em "NEXT" para resolver a tarefa.

Tarefa: Agrupando incidentes

Suponha que você tenha disponível os seguintes atributos descritivos de incidentes a serem agrupados: estado, prioridade e categoria.

A lista dos valores de cada um deles são:

- Os "Estados" que os incidentes assumiram durante seu ciclo de vida são: new, active, resolved, closed, awaiting evidence, awaiting problem, awaiting user info e awaiting vendor. A ordem de apresentação desses estados não necessariamente corresponde à ordem de ocorrência deles no ciclo de vida dos incidentes.
- A "Prioridade" que os incidentes receberam quando foram cadastrados no sistema de gerenciamento de incidentes são: 1 - Critical, 2 - High, 3 - Moderate e 4 - Low.
- A "Categoria" do objeto sobre o qual o incidente ocorreu. Existem 36 categorias. Na lista a seguir, são apresentadas, em ordem alfabética, apenas as 15 categorias mais frequentes nos incidentes: Desktop, Internet, Network, Notebook, Office pack, Passwords and access, Peripheral, Printer, Remote access, SAP, SAP ECC, SAP ECC EN, Software, Software - industry, Telephony.

Utilizando esses valores e considerando o contexto do sistema de gerenciamento de incidentes, acesse o link abaixo para alocar os incidentes em cinco grupos. Todos os valores devem aparecer em ao menos um grupo e todos os grupos devem ser caracterizados por pelo menos um valor de cada atributo descritivo.

https://docs.google.com/spreadsheets/d/157dx1svHo_-CVdOapeZqD4vMHdbKj2-Br9yj1lptjMQ/edit?usp=sharing

Após ter criado os grupos no link acima, clique em "NEXT" para continuar a tarefa.

Tarefa: Descrevendo os grupos de incidentes

Lembre-se que o objetivo é agrupar incidentes semelhantes em relação ao seu tempo de resolução.

Destaque abaixo as regras que você usou para formar os grupos e que você acredita que obrigatoriamente devem ser obedecidas por qualquer mecanismo de agrupamento desses incidentes de forma que a organização dos incidentes nos grupos seja apropriada.

Por exemplo:

1. Incidentes com prioridade X e categoria Y devem ficar obrigatoriamente no mesmo grupo.
2. Incidentes que passam pelo estado Z devem ficar separados dos incidentes que passam pelo estado W.

4. Regras a serem obedecidas no agrupamento (uma por linha, enumeradas):

5. Você acha que, em relação ao número de grupos:

Mark only one oval.

- ☐ é necessário mais que cinco grupos para separar os incidentes de forma mais adequada em relação à similaridade de tempo de resolução;
- ☐ cinco grupos é muito; o melhor seria ter menos grupos, pois de fato os incidentes são bastante similares;
- ☐ cinco grupos é o número ideal para acomodar os incidentes.

6. Explique suas motivações para escolha acima.

Quase finalizando...

7. Para que possamos estimar o tempo que você levou para realizar as tarefas propostas, informe abaixo o horário em que você está finalizando a tarefa:

Example: 8:30 AM

Agradecemos a sua colaboração!

8. Use este espaço para sugestões, comentários ou informações adicionais.

Caso você queira saber mais sobre esta pesquisa, entre em contato conosco:

- Thais Rodrigues Neubauer: thais.neubauer@usp.br

E não esqueça de enviar a sua resposta!! :)

Planilha para realização manual do agrupamento:

AGRUPAMENTO DE INCIDENTES

- Instruções de preenchimento:
- 1. Copie os valores dos atributos descritivos (tabela "Valores") para a coluna correspondente no grupo no qual você deseja que aquele valor seja alocado;
 - 2. Um mesmo valor pode constar em mais de um grupo;
 - 3. Todos os valores devem estar em ao menos um grupo (nenhuma coluna "Cont" pode permanecer com valor "0");
 - 4. Cada grupo deve conter ao menos um valor para cada atributo descritivo;
 - 5. Se necessário for, adicione linhas nos grupos.

Grupo 1		
Categoria	Estado	Prioridade

Grupo 2		
Categoria	Estado	Prioridade

Grupo 3		
Categoria	Estado	Prioridade

Grupo 4		
Categoria	Estado	Prioridade

Grupo 5		
Categoria	Estado	Prioridade

Valores					
Categorias	Cont	Estados	Qtd.	Prioridades	Cont
Desktop	0	New	0	1 - Critical	0
Internet	0	Active	0	2 - High	0
Network	0	Resolved	0	3 - Moderate	0
Notebook	0	Closed	0	4 - Low	0
Office pack	0	Awaiting Evidence	0		
Passwords and access	0	Awaiting Problem	0		
Peripheral	0	Awaiting User Info	0		
Printer	0	Awaiting Vendor	0		
Remote access	0				
SAP	0				
SAP ECC	0				
SAP ECC EN	0				
Software	0				
Software - industry	0				
Telephony	0				

Cont.: É um contador de uso do valor. Trata-se de um controle para ajudá-lo a verificar quais valores você já usou e quais ainda precisa colocar no preenchimento dos grupos.

Anexo A – Atributos log de dados reais

Attributes description: enriched event log of an incident management process

Claudio A. L. Amaral¹, Marcelo Fantinato¹, and Sarajane M. Peres^{1,*}

¹University of São Paulo, São Paulo, 038280-000, Brazil

*sarajane@usp.br

Abstract

In this document, we present the descriptions of the attributes which compose an enriched event log of an incident management process extracted from an instance of the *ServiceNowTM* platform used by an IT company. The log is made available in two versions: (a) [incident_event_log_01.csv] the log as it was used in the experiments for time completion prediction as described in our paper¹ and (b) [incident_event_log_02.csv] an improved log with event labels standardization. This log is composed by 24,918 cases (incidents) and 141,712 events and it was extracted from Mar-2016 to Feb-2017. Information was anonymized for privacy.

Citation Request

If you publish some work based on this event log, please make the appropriate citations to our work as best suited the context of your work. If your experiments are not related to ours, just cite the paper¹ as the first one to use this event log. If you have downloaded this event log from any academic repository, please refer to the corresponding citation policy.

Attributes description

- Control Attributes:
 - *number*: incident identifier with the same number as total cases;
 - *incident state*: attribute with eight levels controlling incident management process transitions from opening until closing the case;
 - *active*: boolean attribute indicating if record is active or closed/canceled;
 - *reassignment_count*: number of times incident has changed group or support analysts;
 - *reopen_count*: number of times incident resolution was rejected by caller;
 - *sys_mod_count*: number of incident updates until that moment;
 - *made_sla*: boolean attribute to incident exceeded target SLA or not;
- Identification and Classification Attributes:
 - *caller_id*: user identifier affected;
 - *opened_by*: user identifier that reported the incident;
 - *opened_at*: incident opening date and time;
 - *sys_created_by*: user identifier that registered the incident;
 - *sys_created_at*: incident creation date and time;
 - *sys_updated_by*: user identifier that made update and generated current log record;
 - *sys_updated_at*: log update date and time;
 - *contact_type*: categorical field with values indicating how incident was reported;
 - *location*: location identifier of place being affected;
 - *category*: description of the first level of service being affected;
 - *subcategory*: description of the second level of service being affected – related to first level;

- *u_symptom*: description about user perception of service availability;
 - *cmdb_ci*: (confirmation item) identifier (not mandatory) referencing homonyms relation and used to report item being affected;
 - *impact*: description of the impact caused by incident. Values are: 1–High; 2–Medium; 3–Low;
 - *urgency*: description to the urgency asked by user for incident resolution. Values are same as *impact*;
 - *priority*: priority calculated by system based on *Impact* and *urgency*;
- Support, Diagnosis and Other Attributes:
 - *assignment_group*: identifier referencing the relation *Group* (database relational model in *ServiceNowTM*) describing support group in charge of incident;
 - *assigned_to*: user identifier in charge of incident;
 - *knowledge*: boolean attribute indicating whether a knowledge base document was used to resolve incident;
 - *u_priority_confirmation*: boolean attribute indicating whether *priority* field was double checked;
 - *notify*: categorical attribute indicating whether notifications was generated for this incident;
 - *problem_id*: identifier referencing homonyms relation describing problem identifier associated with this incident;
 - *rfc*: (chance request) identifier referencing homonyms relation describing change request identifier associated with incident;
 - *vendor*: identifier referencing homonyms relation describing vendor in charge of incident;
 - *caused_by*: relation with RFC code responsible by the incident;
 - *close_code*: resolution code of the incident;
 - *resolved_by*: user identifier who resolved the incident;
 - *resolved_at*: incident resolution date and time;
 - *closed_at*: incident close date and time;

The attributes *sys_created_by*, *sys_created_at*, *sys_updated_by* and *sys_updated_at* are audit log attributes extracted from the audit system offered by *ServiceNowTM*. In fact, the last two attributes were derived from such an audit system. All other attributes were extracted from a relational database related to the incident management system. The attributed *closed_at* is used to determine the dependent variable in the time completion prediction task. The attribute *resolved_at* is highly correlated with *closed_at*. In this event log, some rows may have the same values (they are equal) since not all attributes involved in the real-world process are present in the log. Attributes used to record textual information were not placed in this log.

References

1. do Amaral, C. A. L., Fantinato, M. & Peres, S. M. Attribute selection with filter and wrapper: An application on incident management process. In Ganzha, M., Maciaszek, L. & Paprzycki, M. (eds.) *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems*, vol. 15 of *Annals of Computer Science and Information Systems*, 679–682, DOI: <http://dx.doi.org/10.15439/2018F126> (IEEE, 2018).