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

CLAUDIO APARECIDO LIRA DO AMARAL

Seleção de atributos para mineração de processos na gestão de incidentes

São Paulo
2018

CLAUDIO APARECIDO LIRA DO AMARAL

Seleção de atributos para mineração de processos na gestão de incidentes

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 20 de março de 2018. 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

2018

CATALOGAÇÃO-NA-PUBLICAÇÃO
(Universidade de São Paulo. Escola de Artes, Ciências e Humanidades. Biblioteca)
CRB-8 4625

Amaral, Claudio Aparecido Lira do
Seleção de atributos para mineração de processos na gestão de incidentes / Claudio Aparecido Lira do Amaral ; orientadora, Sarajane Marques Peres ; coorientador, Marcelo Fantinato. – 2018.
136 f. : il.

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. Tìtulo.

CDD 22.ed.– 006.312

Dissertação de autoria de Claudio Aparecido Lira do Amaral, sob o título **"Seleção de atributos para mineração de processos na gestão de incidentes"**, apresentado à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo, como parte dos requisitos 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, aprovado em 20 de março de 2018 pela comissão examinadora constituída pelos doutores:

Profa. Dra. Sarajane Marques Peres
Presidente
Instituição: EACH - USP

Profa. Dra. Kelly Rosa Braghetto
Instituição: IME - USP

Prof. Dr. Clodoaldo Aparecido de Moraes Lima
Instituição: EACH - USP

Profa. Dra. Lucinéia Heloísa Thom
Instituição: UFRGS

Resumo

AMARAL, Claudio Aparecido Lira do. **"Seleção de atributos para mineração de processos na gestão de incidentes"**. 2018. 136 f. Dissertação (Mestrado em Ciências) – Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo, 2018.

O processo de tratamento de incidentes é o mais adotado pelas empresas, porém, ainda carece de técnicas que possam gerar estimativas assertivas para o tempo de conclusão. Este trabalho atua no estudo de um processo real, por meio de um procedimento de mineração de processos, capaz de descobrir o modelo do processo sob a forma de um sistema de transição anotado e propõe meios automatizados de escolha dos atributos que o descrevam adequadamente, de modo a gerar estimativas realistas sobre o tempo necessário para sua conclusão. A estratégia resultante da aplicação de técnicas de seleção de atributos - filtro e invólucro - é capaz de propiciar a geração de sistemas de transição anotados mais precisos e com algum grau de generalização. A solução apresentada neste trabalho representa uma melhoria na mineração de processos, no contexto específico da criação de sistemas de transição anotados e no seu uso como um gerador de estatísticas para o processo nele modelado.

Palavras-chaves: Mineração de processos. Incidente. ITIL. Atributos. Filtro. Invólucro.

Abstract

AMARAL, Claudio Aparecido Lira do. **"Attribute selection for process mining on incident management process"**. 2018. 136 p. Dissertation (Master of Science) – School of Arts, Sciences and Humanities, University of São Paulo, São Paulo, 2018.

The incident management process is the most widely adopted by companies. However, still lacks techniques that can generate precise estimates for the completion time. This work performs a study in a real incident management process, by means of process mining, able to find out the real process model in the form of annotated transition system and propose automated means for selecting attributes that describe it accordingly, in order to generate realistic estimates of the time to conclusion. The resulting strategy of application feature selection techniques - filter and wrapper - is able to provide generation of more accurate annotated transition systems with some degree of generalization. The solution presented in this paper represents an improvement in process mining on the specific context of creation annotated transition system and its use as a statistics generator for the whole modeled process.

Keywords: Process mining. Incident. ITIL. Attribute. Filter. Wrapper.

Lista de figuras

Lista de algoritmos

Lista de tabelas

Lista de abreviaturas e siglas

AG	Algoritmo genético
Atr.	Atributo
CCTA	Central Computar and Telecommunications Agency
DP	Desvio-padrão
Exec.	Execução
Hor.	Horizonte
ITIL	Information Technology Infrastructure Library
ITSM	Information Technology Service Management
MAE	Mean absolute error - erro médio absoluto
MAPE	Mean absolute percentage error - erro percentual médio absoluto
Máx.	Máximo
Med.	Mediana
MSE	Mean squared error - erro quadrático médio
NF	Non-fitting - de índice de não reprodutibilidade do sistema
OGC	Office for Government Commerce
PCF	Process Classification Framework
POMDP	Partial Observable Markov Decision Process
ProM	Process Mining Framefork
Quart.	Quartil
RMSE	Root Mean Square Error - raíz do erro quadrático médio
RMSPE	Root Mean Square Percentage Error - raíz do erro percentual quadrático médio

SAW	Simple Additive Weighting
SLA	Service Level Agreement - acordo de nível de serviço
STA	Sistema de transição anotado
TI	Tecnologia da Informação
XES	eXtensible Event Stream

Lista de símbolos

Log_E	Log de eventos enriquecido
S_A	Estratégia para seleção de atributos
L_A	Lista de atributos
M_T	Sistema de transições de estado anotado
E_T	Estimativa de tempo de execução
L_C	Lista completa de atributos
^{TM}	Marca registrada em inglês
f	Função
A	Conjunto de elementos A
A^*	Conjunto de sequencias finitas
B	Conjunto de elementos B
σ	Sequência de eventos
$\oplus$	Operador de concatenação sequência
$hd^k(\sigma)$	Operador de seleção k primeiros elementos sequência
$tl^k(\sigma)$	Operador de seleção k últimos elementos sequência
$\uparrow$	Projeção uma sequência em outra
∂	Conversão de sequência em outra representação
$B(A)$	Conjunto dos multiconjuntos sobre domínio A
$\mathcal{X}$	Multiconjunto
$\mathcal{X}(a)$	Número de vezes que a está incluído no multiconjunto
AN	Conjunto de atributos
$\#_{an}$	Referencia de atributo an

$\#_{an}(c)$	Referencia de atributo an para caso c
$\perp$	Vazio
ε	Universo dos casos
E*	Espaço de eventos
c	Identificador de um caso
e	Identificador de um evento
$\#_{traco}(c)$	Traço de um caso c
$\hat{c}$	Referencia rápida ao traço de um caso
$\emptyset$	Conjunto vazio
E	Conjunto de eventos
$\mathbb{T}$	Domínio do tempo
$prop_T(e)$	Função data e hora do evento e
$\overline{e}$	Simplificação da função data e hora para o evento e
TS	Sistema de transições
S	Espaço de estados
T	Conjunto de transições
l^{estado}	Função representação estado
C	Conjunto de todos os traços possíveis
L	Traço
R	Conjunto das representações possíveis
$l_1^{estado}(\sigma, k)$	Função representação estado abstração sequência
$l_2^{estado}(\sigma, k)$	Função representação estado abstração multiconjunto
$l_3^{estado}(\sigma, k)$	Função representação estado abstração conjunto

Q	Conjunto das representações possíveis para o evento
S^{inicio}	Conjunto dos estados iniciais
S^{fim}	Conjunto dos estados finais
h	Horizonte
M	Conjunto de valores medidos
l^{medicao}	Função de medição
$max_{\tau}(\sigma)$	Função máximo valor do operador atributo tempo na sequência
$min_{\tau}(\sigma)$	Função máximo valor do operador atributo tempo na sequência
$A(s)$	Função associação conjunto medição ao estado s
$\overline{b}$	Média amostral do multiconjunto
$predicao_{media}(b)$	Função de predição feita com o cálculo da média
η^2	Estatística de correlação eta ao quadrado
η	Estatística de correlação eta
$\overline{r}$	Média dos tempos restantes de um multiconjunto
med_r	Mediana dos tempos restantes de um multiconjunto
$\#$	Indicador de numeração
v	Quantidade de sublogs
j	Quantidade de expansões
$Fitness(i)$	Valor de avaliação para o elemento **i**.
a	Nome da representação utilizada
fn	Nome da função para cálculo

Sumário

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

1 Introdução

A melhoria da eficiência e eficácia em áreas operacionais são metas almejadas em todas as organizações. Os cenários são complexos e por vezes objetivos antagônicos precisam ser atingidos. Alguns exemplos são: a otimização de recursos para redução de custos diretos e indiretos contra a utilização de tecnologias inovadoras que exigem investimentos e possuem custos elevados; outro exemplo é a maximização de lucros versus a melhoria da satisfação dos clientes. Nesse contexto, a utilização de ferramentas de padronização, análise de dados e processos surge como forma de suportar as decisões e tornar tais cenários mais previsíveis e fáceis de gerenciar.

Em alguns setores, como o de prestação de serviços na área de operações de processos de tecnologia, a busca pela previsibilidade e otimização de recursos - humanos e equipamentos - tem difundido a utilização de diversos "modelos de boas práticas", conhecidos como "frameworks". O mais prevalente e amplamente utilizado é o *Information Technology Infrastructure Library* (ITIL) (INTERNATIONAL, 2013).

O "framework ITIL" apresenta uma proposta de organização da aplicação de recursos da tecnologia de informação a partir da divisão de processos de acordo com a sua finalidade. Em sua versão 2011, também conhecida como versão três (v3), abrange um total de vinte e seis (26) processos. Destaca-se, como mais utilizado, o processo de tratamento de incidentes (MARRONE et al., 2014), o qual versa sobre as ações necessárias para corrigir falhas ou degradações. Esse processo tem a característica de gerar resultados operacionais tangíveis a curto prazo (geralmente medido em meses); e sobressai-se a contribuição, nesse caso, na identificação de prioridades, redução do tempo de atendimento, melhoria na forma de previsão de capacidade e utilização dos recursos, entre outras formas de otimização do trabalho.

Nesse cenário de possibilidades, recomendação de estruturação e formalização do processo de tratamento de incidentes, levando à utilização de modelos de atuação sugeridos e à necessidade do estabelecimento de premissas de cenários com tempos alvo para a resolução, há um dificultador, que é a complexidade em realizar eficientemente estimativas precisas acerca do tempo necessário para concluir a execução de uma instância do processo.

Esta lacuna está relacionada às características intrínsecas do próprio incidente [1] [2], à forma de organização do trabalho adotada por pessoas e equipes durante sua atuação na resolução dos incidentes e ao elevado número de atributos utilizados para fazer uma descrição completa dos incidentes. A atuação profissional na área de gerenciamento de processos permite observar que os modelos de processos formalmente estabelecidos tendem a seguir o recomendado pelo ITIL. Porém, o modelo de processo real, por diferentes motivos e variáveis externas ao modelo previsto, destoa em maior ou menor intensidade do que foi formalmente definido. Mesmo em organizações com áreas de governança de tecnologia da informação estruturadas, os indicadores que derivam de uma análise do processo costumam apresentar informações superficiais e imprecisas sobre os incidentes e seus tempos de resolução e conclusão. Esse fato interfere negativamente no estabelecimento de prioridades e ações necessárias para tornar o processo mais eficiente e eficaz.

As situações descritas, caracterizam o cenário do problema tratado neste trabalho, ou seja, a utilização de métodos mais eficientes para análise do processo de tratamento de incidentes. É necessário realizar avaliações que considerem o processo real e permitam identificar quais atributos e suas combinações de fato influenciam a execução durante o processo de tratamento de incidentes. Este trabalho atua no estudo de um processo real de tratamento de incidentes e apresenta meios automatizados para escolha dos atributos que mais bem os descrevem no sentido de permitir a geração de estimativas realistas sobre o tempo necessário para a conclusão de um incidente.

1.1 Definição do problema

Quando ocorre um incidente, ele é identificado e informado por um solicitante. Depois disso, a principal expectativa é conhecer o tempo de conclusão do incidente. As estimativas normais geralmente seguem as indicações de melhores práticas do ITIL, que são baseadas em alguns atributos específicos do incidente, como urgência, categoria, etc. Esta abordagem é bastante genérica e imprecisa porque agrega um grande número de situações distintas e, ao mesmo tempo, tempos de conclusão comuns. À medida que o processo evolui da fase de identificação e classificação para a fase de suporte inicial, depois para

[1] Qualquer situação não prevista que cause impacto (degradação ou indisponibilidade) a um serviço de tecnologia

[2] Definições básicas para os termos necessários para a compreensão desse texto encontram-se no glossário, após as referências bibliográficas. Recomenda-se a leitura prévia.

a investigação e diagnóstico, alguns dos atributos são atualizados e novos atributos são informados e adicionados. Dependendo do escopo da implementação do sistema, o número pode chegar a um total próximo de 100 atributos descritivos. Considerando esse cenário, há um problema em aberto relacionado ao fornecimento de estimativas assertivas sobre o tempo de conclusão de incidentes que não são adequadamente resolvidos por métodos estatísticos simples.

Os sistemas de gerenciamento de incidentes armazenam informações descritivas de instâncias de processo e informações de auditoria sobre o histórico de atualizações do processo em andamento. A combinação de ambos os tipos de informação permite executar uma avaliação detalhada do processo e, portanto, derivar estimativas para cada atividade registrada no sistema.

A análise mencionada, realizada a partir do log resultante, pode ser feita por meio de um procedimento de mineração de processos capaz de descobrir um modelo do processo sob a forma de um sistema de transição de estados anotado (AALST; SCHONENBERG; SONGA, 2011), com informações estatísticas sobre o tempo de execução. Porém, o log resultante da combinação descrita, possui uma granularidade de informação que pode ser demasiadamente extensa, e o número de registros gerados nessa granularidade de informação pode ser proibitivo para execução da análise e gerar modelos de processos sobreajustados, dificultando generalizações importantes para realização de estimativas assertivas. Dessa forma, este trabalho atua nos seguinte itens:

1. a combinação ao log de eventos do sistema de gerenciamento do processo com o log de auditoria para construção do log de eventos enriquecido Log_E;
2. a criação de uma estratégia de seleção de atributos S_A compondo uma lista de atributos L_A;
3. uso do log de eventos enriquecido Log_E e da lista de atributos L_A para descobrir um sistema de transição anotado M_T, que representa o modelo de processo real executado no sistema de gerenciamento do processo;
4. o uso das informações estatísticas de M_T para fornecer uma estimativa de tempo E_T para execução do processo;
5. avaliação da estimativa E_T quanto à sua assertividade, de forma a dar subsídios para o direcionar o processo de otimização em S_A.

1.2 Hipótese

O processo de gerenciamento de incidentes gera um conjunto de informações sobre os incidentes (atributos descritivos) e sobre o processo de tratamento e resolução associado (log de auditoria e eventos do sistema que suporta o gerenciamento). O sistema de transições anotado, proposto por Aalst, Schonenberg e Songa (2011), oferece estatísticas sobre o tempo restante de execução do processo de incidentes a cada transição de estado, ou seja, a cada interação realizada com o incidente por meio do sistema de gerenciamento.

A hipótese deste trabalho defende que há uma lista ótima de atributos descritivos do incidente L_A, que pode ser obtida a partir de uma lista completa de atributos L_C, que ao ser usada de forma combinada ao log de eventos enriquecido do sistema Log_E, permite a criação de um sistema de transição anotado M_T capaz de gerar de estimativas de tempo para conclusão E_T otimizadas em termos de assertividade.

1.3 Objetivos

O objetivo geral deste trabalho é aplicar estratégias de seleção de atributos para encontrar a lista de atributos descritivos de um incidente, que viabiliza a construção de um sistema de transição anotado do qual estimativas assertivas de tempo de execução do processo podem ser obtidas. A assertividade das estimativas produzidas a partir do modelo gerado neste trabalho deve ser avaliada contra recomendações para utilização de atributos, na construção de modelos de processos, propostas na literatura (AALST; SCHONENBERG; SONGA, 2011),

Como objetivos específicos tem-se:

- criar um ambiente de experimentação referente ao processo de gerenciamento de incidentes, no qual estejam presentes todos os elementos necessários para estudo da seleção de atributos proposta neste projeto;
- estabelecer uma estratégia de avaliação do processo de seleção de atributos, a partir de técnicas de filtro ou invólucro (do inglês *wrapper*), que seja orientada pelas estimativas de tempo derivadas do sistema de transição anotado;

- modelar e implementar o processo de seleção de atributos de maneira que seja possível avaliar as diferentes formas de seleção de atributos e suas variações nos modelos de representação (AALST; SCHONENBERG; SONGA, 2011).

1.4 Resultados obtidos

A estratégia resultante da aplicação de técnicas de seleção de atributos foi capaz de propiciar a geração de sistemas de transição anotados mais precisos e com algum grau de generalização para os casos de uso em processo de gerenciamento de incidentes. Os modelos resultantes da estratégia aqui discutida apresentaram um resultado superior àqueles obtidos com técnicas guiadas pelas boas práticas definidas no *framework* ITIL e com as técnicas de seleção dos atributos adotadas na literatura atual de mineração de processos. Desta forma, a solução construída neste trabalho representa uma melhoria na mineração de processos, no contexto específico da criação de sistemas de transição anotados e no seu uso como um gerador de estatísticas de predição para o processo nele modelado.

O *framework* ITIL, no qual o processo de incidentes e outros similares estão estabelecidos, é amplamente utilizado em diversas organizações (MARRONE et al., 2014). No mercado, há ferramentas que implementam soluções com referência neste *framework*, logo, a solução tem potencial para complementar ou ser integrada a produtos de software com boa aceitação na área de gestão de processos.

1.5 Método de pesquisa

A natureza da pesquisa é aplicada, no contexto do processo de gestão de incidentes. É caracterizada como sendo do gênero de pesquisa prática, com a utilização de dados de um processo real de incidentes oriundo da plataforma $ServiceNow^{TM}$ utilizada por uma empresa de tecnologia. O ambiente de experimentação, utilizou esses mesmos dados provenientes do processo de gerenciamento de incidentes.

Para identificação do conhecimento referente à área de aplicação da pesquisa, bem como à área de seleção de atributos, o procedimento escolhido foi a pesquisa bibliográfica via estudos exploratórios e revisão sistemática da literatura.

Como ferramenta para geração de estimativas de tempo de execução do processo foi aplicado o sistema de transição anotado, proposto por Aalst, Schonenberg e Songa (2011). Os experimento iniciais foram gerados a partir das plataformas ProM e Disco. Posteriormente, os demais experimentos foram executados a partir da implementação realizada em linguagem R. Para implementação da seleção de atributos, foram utilizadas técnicas de filtro e invólucro com as técnicas de busca Subida da encosta e Primeira melhora, usando a seleção incremental (do inglês, *forward selection*). Dessa forma, foi possível avaliar a performance de cada um dos atributos descritivos de incidentes disponíveis, isoladamente e em conjunto. Adicionalmente, foi modelado o processo de busca com algoritmo genético com objetivo de apresentar outra estratégia de seleção complementar.

A avaliação de resultados foi realizada aplicando estatística descritiva e inferencial. Informações utilizadas para acurácia foram o MAPE (do ingles, *Mean absolute percentage error*) e para avaliação da capacidade de generalização, a taxa de não reprodutibilidade do sistema de transições anotado.

1.6 Organização deste documento

Esta dissertação é composta por seis capítulos, considerando esta introdução, e três apêndices:

- O capítulo 2 apresenta os principais conceitos teóricos referentes a mineração de processos, sistema de transição anotado, seleção de atributos considerando seleção por filtro e por invólucro implementada com os algoritmos de busca heurística (subida de encosta e busca pela primeira melhora) e com a meta heurística (algoritmos genéticos), e *framework* ITIL com foco no gerenciamento de incidentes.
- No capítulo 3 é apresentada a abordagem de seleção de atributos, estabelecida neste trabalho, para uso no contexto de predição de tempo de conclusão de instâncias de incidentes. A abordagem é apresentada em termos de sua arquitetura geral e também a partir do detalhamento sobre a construção de um log de eventos aqui chamado de "log de eventos enriquecido", sobre decisões tomadas em relação ao uso de abstrações de representação de estados e de intervalo de valores para o horizonte máximo, sobre as funções de predição estabelecidas para uso no sistema de transição anotado, sobre os procedimentos de avaliação adotados nos experimentos

e, finalmente, sobre detalhes aplicados aos procedimentos de busca heurística e meta heurísticas utilizados na abordagem.

- Experimentos exploratórios que visam fornecer um ambiente propício para o entendimento detalhado do comportamento do processo de gerenciamento de incidentes são apresentados no capítulo 4. Nesse capítulo é também discutido, com base na exploração do contexto com sistemas de transição, simples e anotados, o quão importante é a seleção de atributos adequados para construção de modelos que possam ser usados como preditores de tempo de conclusão de incidentes. Essa análise exploratória é apresentada em termos de modelos criados com as ferramentas DISCO e ProM.
- Na sequência, no capítulo 5 são organizados os quatro experimentos realizados para validação das opções de seleção de atributos propostas na abordagem de seleção de atributos, quais sejam: pelo conhecimento do especialista, por filtro com *ranking*, por invólucro com buscas heurísticas, por invólucro com algoritmo genético. Os resultados são apresentados no decorrer da seção juntamente com análises mais específicas. Um conjunto de análises mais gerais é apresentado ao final do capítulo.
- O capítulo 6 apresenta as conclusões do trabalho acompanhadas da enumeração das principais contribuições obtidas, das limitações do trabalho e das possibilidades de trabalhos futuros.
- O apêndice A é dedicado a fornecer detalhes do modelo de dados da relação incidente, usado na plataforma $ServiceNow^{TM}$.
- O apêndice B é dedicado a fornecer detalhes do log de auditoria, usado na plataforma $ServiceNow^{TM}$.
- O apêndice C é dedicado a fornecer a lista detalhada de atributos usados no log de eventos enriquecidos, organizada por classe de atributos e acompanhada do domínio de cada atributo.

2 Referencial teórico

Este capítulo apresenta as informações teóricas referentes aos principais conceitos e técnicas utilizados no desenvolvimento desse trabalho. Ele inicia com uma visão geral sobre mineração de processos, seguindo principalmente o trabalho de van der Aalst (AALST, 2011), atualmente o principal pesquisador da área. Na sequência, é apresentado um detalhamento dos sistemas de transição de estados (seção 2.1.1), das abstrações usadas nele (seção 2.1.2), e da sua versão com transições anotadas, constituindo o sistema de transição de estados anotado (seção 2.1.3). Posteriormente, são apresentadas as técnicas de seleção de características (seção 2.2) acompanhadas de uma breve explicação sobre as buscas heurísticas usadas no trabalho. A seleção utilizando a busca meta-heurística com algoritmos genéticos é apresenta na seção 2.3. Informações sobre o *framework* ITIL são apresentadas na seção 2.4. Na última parte, seção 2.5, são apresentados os trabalhos recentes e relevantes relacionados à mineração de processos operacionais em tecnologia.

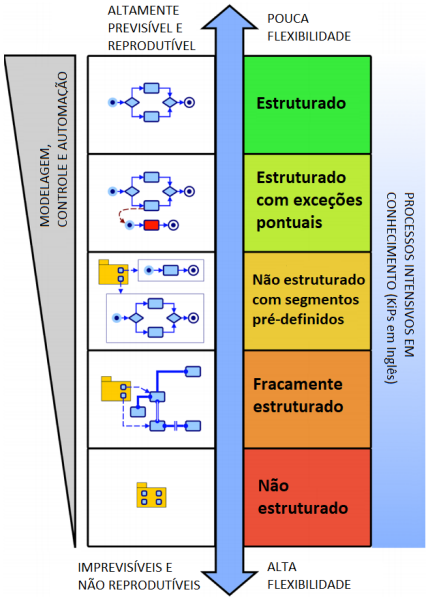
2.1 Mineração de processos

Atualmente, há um grande número de empresas que utilizam sistemas de informação orientados a processos para suportar suas operações. Esses sistemas registram logs de execução, chamados de log de eventos, com informações sobre as atividades executadas. Geralmente, esses registros, além das atividades executadas, possuem atributos adicionais. Neste último caso, espera-se que os dados armazenados e usados em uma análise do processo sejam especificamente referentes a execuções de processos de negócio, e essa suposição é feita ao longo das fases de análise da mineração.

Em Ciccio, Marrella e Russo (2015) é apresentada uma organização para classes de processos e tipo de análise a que eles podem ser submetidos. Essa organização é dada na forma de um espectro, como apresentado na figura 1. Essa organização foi construída com base no nível de estruturação e previsibilidade que os processos possuem, em sua influência direta no nível de automação, controle e suporte que podem ser fornecidos, bem como no grau de flexibilidade requerido.

Na parte superior do espectro, na figura 1, estão os processos estruturados que são caracterizados por abrangerem tarefas rotineiras altamente previsíveis e com requisitos de

Figura 1 – Espectro do gerenciamento de processos



Fonte: Adaptado de Ciccio, Marrella e Russo (2015)

baixa flexibilidade. As interações entre os participantes do processo são controladas. A lógica do processo é conhecida, sendo definida previamente em termos das atividades a serem executadas, suas dependências e os recursos envolvidos da realização das atividades.

Os processos estruturados com exceções pontuais têm características semelhantes aos processos estruturados, pois refletem atividades operacionais que geralmente cumprem um plano predefinido. Porém, a ocorrência de eventos e exceções externas pode tornar a estrutura do processo mais flexível e gerar desvios das definições de trabalho de referência. Há também os desvios não previstos, que só podem ser identificados durante a execução de uma instância do processo.

Nos processos não estruturados com segmentos predefinidos, a lógica geral do processo não é definida explicitamente, mas a existência de políticas e regulamentos permite identificar fragmentos pré-definidos e estruturados. Esses fragmentos podem se referir a procedimentos explícitos e prescritivos, ou podem assumir a forma de modelos e diretrizes não especificados. As partes do processo que não são definidas só podem ser especificadas e incorporadas ao modelo de processo existente à medida que o processo evolui.

Há muitos processos com o comportamento fracamente estruturado, ou seja, embora as rotinas de trabalho não estejam sujeitas a procedimentos de referência prescritivos, a existência de políticas e regras de negócio induzem restrições que implicitamente delimitam o

escopo de ação dos participantes do processo. O conjunto das atividades pode ser conhecido e predefinido, mas sua ordem de execução não é totalmente previsível, pois existem alternativas. Ao invés de usar uma linguagem processual para expressar as sequências permitidas de atividades, os processos são descritos por meio do uso de restrições, que restringem o comportamento de execução indesejável.

Finalmente, o espectro chega aos processos não estruturados, caracterizados por um baixo nível de previsibilidade e requisitos de alta flexibilidade. Os participantes do processo decidem sobre as atividades e sua ordem de execução, fazendo com que a estrutura do processo evolua dinamicamente. Esses processos refletem diretamente o trabalho relacionado ao conhecimento e as atividades de colaboração conduzidas por regras e eventos, para os quais nenhum modelo predefinido pode ser especificado e pouca automação pode ser fornecida. É sabido que para tarefas específicas, há que se considerar mudanças inesperadas no contexto operacional. Os processos com essas características têm apenas seu objetivo final conhecido previamente. Um exemplo desse tipo de processo é o citado em Aalst (2011) referente ao processo de diagnóstico e tratamento de pacientes em um hospital alemão, que possui um total de 619 atividades distintas, executadas por 266 indivíduos em um total de 2.765 instâncias de processos.

Alternativamente às definições construídas com a análise do espectro, há uma abordagem mais resumida, usada por Aalst (2011), na qual os processos podem ser classificados como sendo do tipo **lasanha** ou **espaguete**. O primeiro tem uma estrutura clara e a maioria dos casos (instâncias do processo) são tratados de maneira conhecida. Há relativamente poucas exceções e os atores do processo têm um entendimento claro sobre o fluxo de trabalho. Contudo, mesmo com esse comportamento, segundo o mesmo autor, é impossível fazer uma definição formal de todos os requisitos que caracterizam um processo. Processos do tipo lasanha, normalmente, fazem parte do espectro dos processos estruturados, cujas as atividades são passíveis de repetição e possuem um conjunto de entradas e saídas bem definidas. Alternativamente, também podem fazer parte do espectro dos processos semiestruturados, nos quais os requisitos de informação das atividades são conhecidos e é possível esboçar os procedimentos seguidos, entretanto, algumas atividades requerem uma interpretação e podem sofrer desvios, dependendo das informações ou características do caso. Uma regra geral, é que mais de oitenta por cento dos eventos ocorrem de maneira conhecida e os atores participantes podem confirmar a validade do modelo. Um característica importante é que esse tipo de processo pode ser utilizado para

análise de suporte operacional. O segundo tipo de processo, espaguete, está no espectro dos processos não estruturados, no qual é difícil definir as pré e pós condições para as atividades. São processos guiados pela experiência, intuição, tentativa e erro com informações vagas sobre qualidade.

A **mineração de processos** tem como objetivo descobrir, monitorar e melhorar processos reais por meio da extração de conhecimento a partir dos logs de eventos existentes nos sistemas atuais (AALST, 2011). Porém, apesar da informação existir em grande quantidade, sua obtenção geralmente requer uma etapa de pré-processamento. Com os logs, assume-se ser viável ordenar os seus eventos de maneira que cada evento aponte para uma atividade e seja relacionada a uma instância de processo (caso). Esses eventos podem ter ocorrido em diferentes momentos, portanto, precisam de uma ordenação pela data de ocorrência. Esses são então agrupados e todos os eventos relacionados a um caso passam a compor um bloco chamado de **traço**. Um **log de eventos** é composto por um ou mais traços.

Com as informações do log de eventos é possível executar a mineração de processos, a qual é, por definição, dividida em três tipos principais de tarefas:

- a **descoberta** de modelos com fluxos de controle descrevendo o processo real em execução (ou seja, a descoberta do processo). Essa atividade é realizada sem que exista informação prévia sobre o modelo. Há vários algoritmos que podem ser utilizados, um deles é o $\alpha - algorithm$ (AALST, 2011);
- a avaliação da **conformidade** de um determinado evento no log em relação a um modelo pré-determinado do processo (ou seja, verificação de conformidade). Esse tipo de avaliação permite que sejam identificados desvios nos processos executados quando comparados com as definições formais;
- e a extensão de modelos de processos existentes com informações adicionais, ou seja, a **melhoria** do processo. A melhoria pode ser obtida ao realizar a comparação da execução real contra aquela prevista no modelo. Ao realizar essas comparações, diversas situações de otimização podem ser identificadas, como por exemplo, atividades que são executadas sequencialmente e poderiam ser transformadas em atividades paralelas (AALST, 2011).

Complementando as tarefas anteriores, geralmente há dois tipos de análises realizadas em mineração de processos. A primeira diz respeito à utilização de registros do

log de eventos para casos já encerrados. Essa análise produz resultados conhecidos como dados *post mortem*, ou seja, é possível inferir informações, porém, não será possível tomar ações para interferir no resultado final da execução do processo. A segunda análise utiliza o que é chamado de registros de log parciais. Esses registros são informações de casos em execução, não concluídos. Esse tipo de análise permite que informações sejam avaliadas e utilizadas para tomada de decisão a respeito de casos em andamento. Um exemplo de decisão seria a troca do técnico responsável pelo tratamento de um incidente para tentar diminuir o tempo de resolução.

Além das atividades clássicas, também é possível explorar o log de eventos a fim de criar modelos preditivos, ou seja, modelos que são úteis para prever as atividades e tempos futuros de instâncias de processos ainda em andamento. Alguns casos de uso típicos dessa atividade seriam: predição do tempo de execução restante para instâncias de processo que estão atrasadas, de modo que a qualidade do serviço possa ser melhorada; prover recomendação para alocação de recursos de maneira a otimizar a utilização dos colaboradores.

Segundo Aalst (2011), outra forma de análise de processos pode ser realizada utilizando o conceito de perspectivas. As principais são:

- Controle do fluxo: perspectiva focada na ordenação de atividades, visando encontrar uma boa caracterização de todas as sequências possíveis. O resultado é expresso em termos de uma rede de Petri ou alguma outra notação de processos.
- Organizacional: com foco em informações implícitas no log de eventos e relacionadas a recursos, ou seja, os atores envolvidos (pessoas, sistemas, papéis ou departamentos) e como se relacionam. Busca-se estruturar a organização classificando as pessoas e suas funções, e visualizar a rede social;
- Caso (instância de processo): com foco nas propriedades específicas de casos. Uma instância pode ser caracterizada pelo seu caminho no processo ou pelos atores que a executam. Os casos também podem ser descritos pelos domínios e valores contidos em seus atributos;
- Temporal: focada na temporização e na frequência de eventos. Nesse caso, referências temporais são anotadas nos eventos, possibilitando a descoberta de gargalos, a medição de níveis de serviço, o monitoramento do uso de recursos, e a estimativa do tempo de processamento necessário para concluir as instâncias existentes.

Nas subseções seguintes serão apresentados os conceitos referentes aos sistemas para representação de processos relevantes para este trabalho: sistema de transições e sua versão com estados anotados.

2.1.1 Sistema de transições

O objetivo da mineração de processos é extrair conhecimento sobre um processo a partir da análise de um log de eventos. Nesse trabalho, essa análise é feita por meio da criação de um sistema de transições a partir de um log de eventos do processo de incidentes. Para sua construção, uma série de definições e conceitos precisam ser estabelecidos. A abordagem de Aalst, Schonenberg e Songa (2011) para desenvolvimento de um sistema de transições pressupõe, informalmente, que:

- Um log de eventos é um conjunto de eventos.
- Um traço em um log representa uma instância de processo específica (também conhecido como "caso").
- Um processo é constituído de uma ou mais instâncias.
- Cada evento no log está relacionado a um traço específico e é único, ou seja, não pode ocorrer mais de uma vez no log.
- Cada evento é uma referência para uma única atividade e está relacionado a um caso.
- É possível ordenar eventos de um caso de forma sequencial de acordo com o momento em que ocorrem.
- Os eventos podem ter atributos.

Para o estabelecimento do sistema de transições, uma formalização de conceitos e o estabelecimento de operadores se fazem necessários. Segundo Aalst, Schonenberg e Songa (2011), a maneira mais simples de apresentar os traços de um log de eventos é usando um modelo de sequências. Esse modelo torna possível descrever a semântica operacional dos sistemas de transições.

Considerando que $f \in A \rightarrow B$ é uma função com domínio no conjunto de elementos A e contradomínio no conjunto de elementos B e que f é uma função parcial, i.e., o domínio de f pode ser um subconjunto de A. Sendo A um conjunto de elementos, A^* é o conjunto de todas as sequências finitas que podem ser obtidas de A. Uma sequência finita

em A^* de tamanho n é obtida a partir de um mapeamento $(A) \rightarrow A^*$ que gera sequências representadas por strings $\sigma = \langle a_1, a_2, \ldots, a_n \rangle$ nas quais $a_i \in A$, $\sigma(i) = a_i$ para $1 \leq i \leq n$ e $|\sigma| = n$ é o tamanho da sequência.

Para a realização das operações com sequências, os seguintes operadores são necessários:

- $\sigma \oplus a' = \langle a_1, a_2, \ldots, a_n, a' \rangle$, i.e., adição de um elemento ao final de uma sequência, gerando uma nova sequência de tamanho $|\sigma| + 1$;
- $\sigma_1 \oplus \sigma_2$. i.e., concatenação de duas sequências, gerando uma nova sequência de tamanho $|\sigma_1| + |\sigma_2|$;
- $hd^k(\sigma) = \langle a_1, a_2, \ldots, a_k \rangle$, i.e., encontra a cabeça da sequência contendo os k, com $k \leq n$, primeiros elementos dessa sequência, gerando uma nova sequência de tamanho k;
- $tl^k(\sigma) = \langle a_{n-k+1}, a_2, \ldots, a_n \rangle$, i.e., encontra a cauda da sequência contendo últimos k elementos, gerando uma nova sequência de tamanho k. Ressalte-se que: $tl^0(\sigma)$ é uma sequência vazia e $tl^k(\sigma) = \sigma$ quando $k \geq n$;
- $\sigma \uparrow X$ é a projeção da sequência σ sobre um subconjunto $X \subseteq A$, gerando uma nova sequência de tamanho $\leq |\sigma|$. Como exemplo da execução desta operação considere o exemplo $\langle a, b, c, d, a, b, e \rangle \uparrow \{a, b\} = \langle a, b, a, b \rangle$;
- $\partial_{conjunto}(\sigma)$ faz a conversão de uma sequência σ de tamanho n em um conjunto X de tamanho $\leq n$. Por exemplo, $\partial_{conjunto}(\langle a, b, b, c, d, d, e \rangle) = \{a, b, c, d, e\}$.

Além do conceito referente a conjunto de elementos e sequência de elementos, o conceito de multiconjunto (também conhecido como *bag* ou *multiset*) de elementos também é usado no contexto de sistemas de transição. No multiconjunto, elementos podem ocorrer múltiplas vezes. Seja $B(A) = A \rightarrow \mathbb{N}$ o conjunto dos multiconjuntos sobre um domínio finito A, ou seja, $\mathcal{X} \in B(A)$ é um multiconjunto no qual para cada elemento $a \in A$, $\mathcal{X}(a)$ representa o número de vezes que a está incluído no multiconjunto. Como exemplo, considere $\mathcal{X} = [a, b^5, c^2, d, e]$, no qual o número sobrescrito indica a quantidade de ocorrências do elemento, sendo que se a ocorrência é única, não há necessidade de representá-la explicitamente. No contexto de multiconjunto, $\partial_{multiconjunto}(\sigma)$ faz a conversão de uma sequência σ de tamanho n em um multiconjunto $\mathcal{X}$ de tamanho $\leq n$, por exemplo $\partial_{multiconjunto}(\langle a, b, b, c, d, d, e \rangle) = [a, b^2, c, d^2, e]$.

Para concluir a formalização dos conceitos necessários para construção de um sistema de transições, é necessário definir: caso, traço e evento. Seja ε o universo dos casos de um processo e E^* o universo de eventos associados. Desde que eventos e casos precisam assumir identificadores únicos, $c \in \varepsilon$ será usado como identificador de um caso e $e \in E^*$ será usado como identificador de um evento. Para um universo de casos ε há um conjunto de atributos $AN = \{an_1, an_2, \cdots\}$ associada, sendo assim $\#_{an}(c)$ é o valor do atributo an para o caso c. Se um caso não possui atributo de nome an, $\#_n(c) = \perp$.

Todos os casos possuem um atributo obrigatório especial chamado "traço". Assim, $\#_{traco}(c) \in E^*$ e $\hat{c} = \#_{traco}(c)$ é uma forma abreviada para fazer referência ao traço de um caso. Um traço é uma sequência finita de eventos $\sigma \in E^*$ tal que cada evento aparece uma única vez, ou seja, $1 \leq i \leq j \leq |\sigma| : \sigma(i) \neq \sigma(j)$.

Um log de eventos é um conjunto de casos $L \subseteq \varepsilon$ tal que cada evento aparece no máximo uma única vez em todo o log, i.e., para quaisquer casos $c_1, c_2 \in L$ tal que $c_1 \neq c_2 : \partial_{conjunto}(\hat{c_1}) \cap \partial_{conjunto}(\hat{c_2}) = \emptyset$. Se um log de eventos possui atributos de data e hora, a ordem do traço deve respeitar esses atributos, ou seja, para qualquer $c \in L$, e quaisquers i e j, tais que $1 \leq i < j \leq |\hat{c}|$: $\#_{time}(\hat{c}(i)) \leq \#_{time}(\hat{c}(j))$. Um evento e é descrito por um identificador único e pode ter várias propriedades. Embora para este trabalho o interesse se concentre nas propriedades de data e hora do evento, outras propriedades como nome, recurso associado, etc, podem ser utilizados nos mais diversos objetivos em mineração de processo. Considerando a propriedade de tempo no domínio do tempo $\mathbb{T}$, que se refere ao momento de registro do evento, há uma função $prop_{\mathbf{T}} \in E^* \to \mathbb{T}$ que associa os registros de tempo aos eventos. Como simplificação, $\overline{e} = prop_{\mathbf{T}}(e)$ refere-se à data e hora do evento $e \in E^*$.

A partir da formalização de todos esses conceitos, a definição de sistemas de transições pode ser apresentada. Um sistema de transições é uma tripla $TS = (S, E, T)$, sendo S o espaço de estados possíveis do processo, E é o conjunto de rótulos dos eventos, e $T \subseteq \{S \times E \times S\}$ é o conjunto de transições que descreve como o sistema pode se mover de um estado a outro. Uma transição $(s1, e, s2) \in T$ significa que o processo pode se mover do estado s1 para o estado s2 pelo ocorrência de um evento e. Um sistema de transições tem um estado inicial e um conjunto de estados finais, sendo que o conjunto de comportamentos possíveis de um sistema de transições é dado por todos os caminhos do estado inicial até algum estado final.

Um traço é possível de acordo com o sistema de transições se ele corresponde a um caminho existente no sistema de transições. Logo, o objetivo na mineração de processos é obter um sistema de transições que, dado um log de eventos, seja capaz de caracterizar adequadamente todos os comportamentos (traços) registrados. É natural supor que, tomando-se um instante no tempo de uma instância de processo, esta esteja em algum estado que dependa totalmente de seu histórico anterior e uma função de representação de estados é responsável por construir esse comportamento.

Uma função de representação de estados l^{estado} é uma função que, dado um traço σ e um número k indicando o número de eventos (elementos) de σ que já ocorreram, produz uma representação de estado. Formalmente, $l^{\text{estado}} \in C \rightarrow R$, em que C é o conjunto de todos os traços possíveis e R é o conjunto das representações possíveis (representação por sequência, representação por conjunto e representação por multiconjunto). Como exemplos de funções, considerando $\sigma = \langle a_1, a_2, \ldots, a_n \rangle \in L$ como sendo um traço de tamanho n, tem-se:

- $l_1^{estado}(\sigma, k) = hd^k(\sigma) = \langle a_1, a_2, \ldots, a_k \rangle$ é uma função que retorna a sequência dos primeiros k elementos de σ. Dessa forma, descreve o estado atual utilizando todo o histórico do caso dada a ocorrência de k eventos.
- $l_2^{estado}(\sigma, k) = \partial_{multiconjunto}(hd^k(\sigma)) = [a_1, a_2, \ldots, a_k]$ é uma função que converte o histórico completo de um traço em um multiconjunto. Para a representação de estado por esta função a ordem dos eventos não é importante, apenas a frequência com a qual aparecem no log.
- $l_3^{estado}(\sigma, k) = \partial_{conjunto}(hd^k(\sigma)) = \{a_1, a_2, \ldots, a_k\}$ é uma função de representação que utiliza a representação de conjuntos e o histórico completo do traço. Nesta função, a ordem e frequência não são importantes, apenas a ocorrência.

Da mesma forma que os estados são representados, também é necessário representar os eventos. Uma função de representação de eventos l^{evento} é uma função que dado um evento, produz uma representação para ele. Formalmente, $l^{\text{evento}} \in E \rightarrow Q$, em que E é o conjunto de todos os eventos possíveis e Q é o conjunto das representações possíveis para o evento (por exemplo, nome da atividade).

Qualquer evento no log estende um traço parcial σ_1 em um traço $\sigma_2 = \langle \sigma_1 \oplus e \rangle$ (concatenação do traço em questão com a sequência contendo o evento e). No sistema de

transições deve existir uma transição conectando o estado $l^{\text{estado}}(\sigma_1)$ ao $l^{\text{estado}}(\sigma_2)$. Essa transição tem o nome de $l^{\text{evento}}(e)$, baseado em uma função de representação l^{evento}.

Baseado nas funções de representação l^{estado} e l^{evento} é possível construir o sistema de transições. Os estados do sistema correspondem aos prefixos no log mapeados para a representação desejada utilizando uma função de representação de estados l^{estado} escolhida. A relação de transição é calculada pela leitura dos traços no sistema de transições utilizando a função de representação l^{evento}.

Assim, de forma mais detalhada, dado um log de eventos $L \subseteq C$ e as funções de representação l^{estado} e l^{evento}, o sistema de transições $TS = (S, E, T)$ é tal que:

- $S = \{l^{\text{estado}}(hd^{\text{k}}(\sigma))|\sigma \in L \wedge 0 \leq k \leq |\sigma|\}$ é o espaço de estados;
- $E = \{l^{\text{evento}}(\sigma(k))|\sigma \in L \wedge 1 \leq k \leq |\sigma|\}$ é o conjunto de rótulos de eventos;
- $T \subseteq S \times E \times S$ é o conjunto de transições descrito como $T = \{(l^{\text{estado}}(hd^{\text{k}}(\sigma)), l^{\text{evento}}(\sigma(k+1)), l^{\text{estado}}(hd^{\text{k+1}}(\sigma)))|\sigma \in L \wedge 0 \leq k < |\sigma|\}$;
- $S^{\text{inicio}} = \{l^{\text{estado}}(<>)\}$ é o conjunto dos estados iniciais;
- $S^{\text{fim}} = \{l^{\text{estado}}(\sigma)|\sigma \in L\}$ é conjunto dos estados finais.

O conjunto de estados do sistema de transição é determinado pelo domínio da função l^{estado} quando aplicada aos dados do log e o sistema de transições tem os nomes baseados na função l^{evento}.

O algoritmo para gerar um sistema de transições é resumidamente composto dos seguintes passos:

- para cada traço $\sigma \in L$:
  1. faça uma iteração sobre k, com $1 \leq k \leq |\sigma|$ crie um novo estado $l^{\text{estado}}(hd^{\text{k}}(\sigma))$ e o insira em S caso ele não exista;
  2. faça uma segunda iteração sobre k, com $1 \leq k \leq |\sigma|$ e crie uma nova transição $l^{\text{estado}}(hd^{\text{k}}(\sigma)) \overset{l^{\text{evento}}(\sigma(k+1))}{\longrightarrow} l^{\text{estado}}(hd^{\text{k+1}}(\sigma))$ e insira em T caso ele já não exista no sistema.

As funções de representação de estados l^{estado} e eventos l^{evento} são as responsáveis pelo formato do sistema de transições e sua apresentação em termos de representação, respectivamente. Em ambas as funções, um ou mais atributos presentes no log de eventos podem ser usados. Um exemplo de representação que poderia ser utilizada é a função de

estados refletindo diretamente o traço σ completo e a função de eventos utilizando todos os atributos para nomear as transições. Com essa estrutura de representação, cada evento seria mapeado em um único estado e todos os estados (com exceção do inicial) seriam visitados uma única vez. Logo, cada nova instância de processo seria única e não seria possível utilizar o histórico (padrão) dos casos anteriores, tornado o sistema ineficaz ao generalizar um modelo para o processo.

2.1.2 Abstrações no sistema de transições

Para alcançar a construção de sistemas eficazes, utiliza-se o conceito de abstração (AALST et al., 2008), que possibilita a construção de sistemas com capacidade de generalização e o alcance do equilíbrio entre um sistema de transição que é muito específico e sobreajustado e outro que é muito genérico e subajustado em relação ao log.

Para a função de representação de estados, há algumas formas de abstração que podem ser utilizadas. A primeira delas é a definição do **horizonte máximo** de seleção do número de eventos aplicado ao prefixo (ou pós-fixo) completo ou parcial de um traço. Pode-se usar o valor de horizonte $h = 1$, que usa apenas o último evento como entrada para a função de representação de estados, um valor > 1, como $h = 4$ que apresenta os quatro últimos eventos no log ou um valor $h = \infty$ que representa o prefixo completo com todos os eventos.

A segunda forma de abstração está relacionada ao formato como é representado um estado e o nível de detalhe desejado para essa representação. As três formas de representação usadas na seção 2.1.1 geram representações com as seguintes características:

- **Sequência**: apresenta o histórico completo e a ordem na qual as atividades foram realizadas - a ordem é importante;
- **Multiconjunto**: apresenta quais as atividades foram realizadas e o número de vezes em que cada uma delas foi executadas - a ordem não é importante;
- **Conjunto**: apenas a execução da atividade é registrada, não importando a ordem de execução e a quantidade.

Além dessas abstrações há outras conhecidas como: filtros de eventos, número máximo de eventos após o filtro, atividades visíveis, etc. Tais abstrações podem fazer com que os sistemas de transição criados tenham comportamentos bem diferentes quanto

ao número de estados gerados e sua capacidade de capturar a diversidade de situações existentes nas instâncias de processo.

2.1.3 Sistema de transições de estado anotado

Nessa seção é apresentado como um sistema de transições pode ser anotado para realizar estimativas de conclusão. A ideia geral é utilizar o log de eventos e gerar o sistema de transições de estados anotado (neste trabalho tratado como STA). Para fazer a estimativa de tempo de execução em uma instância de processo. Toma-se o traço parcial e utiliza-se a função de representação de estado l^{estado} para realizar o mapeamento do traço parcial em um estado no sistema de transições. A partir desse ponto, é possível utilizar a informação coletada de outras instâncias que passaram por esse estado para fazer a estimativa baseada em estatísticas, como o tempo médio para conclusão. Na sequência, será apresentado como construir um sistema de transições anotado e utilizá-lo para fazer estimativas.

O objetivo é adicionar informações de estimativas aos estados do sistema de transição e para isso, os estados são "anotados" com informações de medições. As instâncias são avaliadas em seu histórico de eventos e para cada situação em que estavam em um determinado estado **s**, o tempo restante até a conclusão é registrado nesse estado. Dessa forma, os estados tem informações armazenadas em multiconjuntos de medições que são a base para a realização das estimativas.

Uma função de medição l^{medicao} é uma função que, dado um traço prefixado σ_1 (parte executada anteriormente) e um traço pós-fixado σ_2 (parte que está no futuro do caso) produz uma informação de medição $l^{\text{medicao}}(\sigma_1, \sigma_2)$, como por exemplo o tempo estimado para conclusão. Formalmente, $l^{\text{medicao}} \in (C \times C) \rightarrow M$ no qual C é o conjunto de dos traços possíveis e M é o conjunto dos valores medidos. Funções de medição diferentes podem ser utilizadas.

Para ilustrar, considerando uma estimativa de tempo para execução, na literatura, a função a seguir pode ser usada:

$$l^{\text{medição restante}}(\sigma_1, \sigma_2) = \begin{cases} 0, \text{se } \sigma_2 = \langle\rangle, \\ max_\tau(\sigma_2) - min_\tau(\sigma_2), \text{se } \sigma_1 = \langle\rangle \text{e } \sigma_2 \neq \langle\rangle, \\ max_\tau(\sigma_2) - max_\tau(\sigma_1), \text{se } \sigma_1 \neq \langle\rangle \text{e } \sigma_2 \neq \langle\rangle. \end{cases}$$

na qual,

$$max_\tau(\sigma) = max\{\overline{e}|e \in \sigma\}$$

$$min_\tau(\sigma) = min\{\overline{e}|e \in \sigma\}$$

.

Outras funções podem ser criadas como o tempo gasto, o tempo de permanência (do inglês *sojourn time*), o tempo total do caso etc. Além das funções relacionadas à duração, outros tipos de função podem ser criadas com atributos diferentes como recursos, custo entre outros.

Para a construção do STA, considere $L \subseteq C$, um log de eventos e TS = (S, E, T), um sistema de transições obtido a partir de uma função de estados l^{estado} e uma função de representação l^{evento}. A função de medição $l^{\text{medicao}} \in (C \times C) \to M$, constrói-se uma anotação $A \in S \to B(M)$ em que para qualquer $s \in S$:

$$A(s) = \sum_{\sigma \in L} \sum_{0 \leq k \leq |\sigma|} [l^{medicao}(hd^k(\sigma), tl^{|\sigma|-k}(\sigma))]$$

$$s = l^{estado}(hd^k(\sigma)),$$

assim, a quádrupla (S,E,T,A) é o STA parametrizado por L, l^{estado}, l^{evento} e $l^{medicao}$. A função A associa um multiconjunto de medições a cada um dos estados. Os somatórios duplos percorrem todos os prefixos que correspondem a um estado especifico s. Para cada prefixo mapeado em s, uma medição é adicionada ao seu multiconjunto correspondente.

Um STA pode ser usado como um preditor no contexto de gerenciamento de incidentes. Para isso é usada uma função que, dado um multiconjunto de medições, produz uma predição para conclusão da execução. A medida mais comumente usada nesse contexto é a média, mas também podem ser utilizadas a mediana e os valores mínimo e máximo

entre outros. Formalmente, $predicao \in B(M) \rightarrow M$, em que para algum multiconjunto de medições b, $predicao(b)$ retorna uma predição.

Seja $L \subseteq C$ um log de eventos e (S,E,T,A) um STA parametrizado por L, l^{estado}, l^{evento} e $l^{medicao}$. Além disso, seja $predicao \in B(M) \rightarrow M$ uma função de predição. Para qualquer traço parcial, σ_N, o valor estimado para predição será

$$predicao(A(l^{estado}(\sigma_N)))$$

se

$$l^{estado}(\sigma_N) \in S.$$

Seja $b = [b_1, b_2, \ldots, b_n]$ um multiconjunto associado a um estado, uma função de predição que utiliza a média amostral pode ser definida como,

$$\overline{b} = \frac{\sum_{i=1}^{n}(b_i)}{n}$$

e

$$predicao_{media}(b) = \overline{b}$$

Na tabela 1, há um log de eventos exemplo. Cada uma das linhas representa uma instância de processo em execução, isto é, o primeiro traço $\langle A^{00}, B^{06}, C^{12}, D^{18}\rangle$ é referente a uma instância na qual a atividade A foi executada no instante 0, a atividade B no instante 6 e assim sucessivamente. Cada instância inicia sua contagem de tempo com a execução do primeiro evento.

Tabela 1 – Frequência dos estados nas instâncias de processos de gerenciamento de incidentes

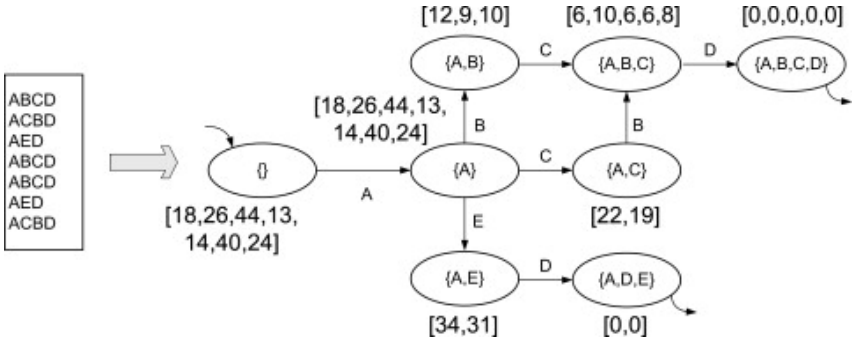
Traço	Sequência de atividades σ
1	$\langle A^{00}, B^{06}, C^{12}, D^{18}\rangle$
2	$\langle A^{10}, C^{14}, B^{26}, D^{36}\rangle$
3	$\langle A^{12}, E^{22}, D^{56}\rangle$
4	$\langle A^{15}, B^{19}, C^{22}, D^{28}\rangle$
5	$\langle A^{18}, B^{22}, C^{26}, D^{32}\rangle$
6	$\langle A^{19}, E^{28}, D^{59}\rangle$
7	$\langle A^{20}, C^{25}, B^{36}, D^{44}\rangle$

Fonte: Adaptado de Aalst, Schonenberg e Songa (2011)

Suponha que foi utilizada uma função de representação de estados l^{estado} que representa os traços parciais por meio de um conjunto de atividades já executadas. Agora considere como exemplo todos os prefixos do primeiro traço $\langle A^{00}, B^{06}, C^{12}, D^{18}\rangle$. O prefixo

vazio $\langle\rangle$ é mapeado para um estado nulo $\emptyset$ e tem um valor de tempo restante calculado pela função de medição $l^{\text{medição restante}}$ igual a 18 unidades de tempo. Logo, o multiconjunto de medições para o estado $\emptyset$ tem esse valor adicionado. O prefixo $\langle A^{00}\rangle$ é mapeado para o estado {A} e também tem o mesmo valor 18. Continuando a avaliação, o prefixo $\langle A^{00}, B^{06}\rangle$ é mapeado para o estado {A,B} e tem o tempo restante de 12 unidades de tempo, logo, este é o valor adicionado à anotação do estado. O prefixo $\langle A^{00}, B^{06}, C^{12}\rangle$ tem o valor de anotação 6 ao estado {A,B,C} e finalmente o prefixo $\langle A^{00}, B^{06}, C^{12}, D^{18}\rangle$ tem o valor de anotação 0 acrescentado ao estado {A,B,C,D}. Este procedimento é repetido para os demais 6 traços do log de eventos. A figura 2 apresenta o resultado desse processamento realizado e apresenta o sistema de transição anotado (STA) gerado.

Figura 2 – Exemplo de STA com abstração conjunto e horizonte infinito



Fonte: Aalst, Schonenberg e Songa (2011)

Agora, vamos utilizar o STA da figura 2 para realizar a predição do tempo de conclusão utilizando a função com a estatística da média amostral. Considere um novo caso N que ainda não foi concluído. O traço parcial observado é $\sigma_N = \langle A^{85}, E^{95}\rangle$ (utilizando a mesma notação da tabela 1), com a atividade A tendo ocorrido no instante 85 e a atividade E no instante 95. A função de representação de estados gera o valor $l^{estado}(\sigma_N) = \{A, E\}$. Nesse estado {A,E}, há duas medições de tempo restante, sendo que $A(l^{estado}(\sigma_N)) = [34, 31]$. Logo, a predição de tempo restante é dada pela função $predicao_{media}([34, 31]) = 32, 5$.

No trabalho original (AALST; SCHONENBERG; SONGA, 2011), van der Aalst cita as limitações e informa que "o desafio é selecionar os atributos corretos", evitando que se perca em uma busca exaustiva quando há potencialmente outros fatores contextuais gravados no log que influenciam as estimativas.

2.2 Seleção de atributos

A seleção de atributos é importante na mineração de processos quando o objetivo é identificar subconjuntos de atributos que possam conduzir à geração de sistemas ou modelos que generalizam melhor, são mais relevantes e possuem condições melhores de representar um processo ou de estimar alguma característica relacionada ao processo, por exemplo, em termos de características dos traços.

O problema caracterizado pela seleção de atributos é o de encontrar um subconjunto de atributos, a partir do conjunto de atributos completo de um conjunto de dados. Dessa maneira, assumindo que o melhor subconjunto de atributos foi encontrado, um algoritmo de indução, executado no conjunto de dados utilizando o subconjunto de atributos, teria condições de gerar um modelo de predição com a melhor acurácia possível. Logo, o problema de seleção de atributos pode ser reduzido ao problema de encontrar um subconjunto ótimo de atributos.

A definição de um subconjunto ótimo de atributos pressupõe que existe um algoritmo de indução e um conjunto de dados rotulado sobre o qual esse algoritmo será aplicado. Segundo Kohavi e John (1997), se, a partir de um subconjunto de atributos, o algoritmo de indução alcança a acurácia máxima de predição, então esse subconjunto de atributos é ótimo.

Ainda seguindo tal definição, para a construção um modelo de predição com a melhor acurácia, o melhor subconjunto de atributos deve ser selecionado por um algoritmo de seleção. O cenário mais complexo para utilização dessa definição, em termos práticos, é a impossibilidade de acesso à distribuição real dos dados subjacentes a um problema de predição, logo, estimativas devem ser realizadas a partir de resultados obtidos com o uso dos dados existentes.

Usualmente os atributos de um conjunto de dados são classificados em relevantes e irrelevantes, sendo que na literatura, habitualmente, dois níveis de relevância são definidos: fraca e forte. A relevância deve ser definida em termos de um classificador de Bayes ótimo para um determinado problema. Um atributo é **fortemente** relevante se a remoção deste atributo resulta em uma redução de desempenho em um classificador Bayes ótimo. Um atributo é **fracamente** relevante se ele não é fortemente relevante e existe um subconjunto de atributos, tal que o desempenho do classificador Bayes, neste subconjunto, é pior do

que o desempenho na união do primeiro atributo com esse subconjunto de atributos. Um atributo é **irrelevante** se não e fortemente relevante ou fracamente relevante.

Geralmente, resultados ótimos de predição são obtidos com o uso dos atributos fortemente relevantes e alguns atributos fracamente relevantes. Há relatos na literatura de situações na qual atributos irrelevantes fazem parte de um conjunto de atributos ótimo, porém, são situações pouco frequentes.

Características importantes da seleção de atributos são a medida de relevância de um subconjunto de variáveis (ou atributos) e a estratégia de otimização que encontra o subconjunto ótimo com referência a subconjuntos selecionados. Os procedimentos de seleção de subconjuntos de variáveis podem ser divididos em três grupos: por filtro, por invólucro (do inglês *wrapper*) e incorporados (do inglês *embedded*) (GUYON; ELISSEEFF, 2003). Neste trabalho, serão aplicados os métodos de filtro e invólucro.

No caso dos procedimentos de filtro, a medida de relevância é definida independentemente do algoritmo de aprendizagem. O procedimento de seleção de subconjuntos pode ser visto como um passo de pré-processamento. No caso de procedimentos de invólucro, a medida de relevância é diretamente definida a partir do algoritmo de aprendizado, tal como o custo de aprendizagem e capacidade de generalização. Embora as abordagens de filtro sejam mais rápidas, sua principal desvantagem é que um subconjunto ótimo de variáveis pode não ser independente do viés da representação usada no algoritmo aplicado na fase de aprendizado. No caso de procedimentos por invólucro, o algoritmo de aprendizagem deve atender a duas condições principais: o número de parâmetros a serem otimizados deve ser o menor possível e o algoritmo deve ser eficiente computacionalmente (KOHAVI; JOHN, 1997). A seleção de subconjuntos de atributos é feita usando o algoritmo de indução como uma “caixa preta”.

De acordo com Blum e Langley (1997), antes de iniciar as atividades de aprendizado automático, há duas tarefas que precisam ser realizadas: decidir quais atributos podem ser usados para descrever o conceito a ser aprendido e como combiná-los. Tomando por base essa suposição, a seleção de atributos é proposta neste documento como uma fase essencial para a construção de modelos de predição capazes de prever adequadamente o tempo de conclusão dos chamados – “incidentes”.

2.2.1 Filtros e *ranking*

O objetivo principal dos métodos de filtro é selecionar os atributos relevantes que têm potencial para produzir um resultado otimizado e remover os atributos irrelevantes. Estes métodos são vistos como um passo de pré-processamento, uma vez que são aplicados de forma independente e antes da escolha do modelo de aprendizagem. Devido à sua independência, os métodos de filtro são tidos frequentemente como competitivos em tempo de execução quando comparados com outros métodos de seleção de atributos e podem fornecer um formato de seleção de atributos genérico, livre da influência do comportamento dos modelos de aprendizagem.

Considere um conjunto de dados com: dados, atributos e uma variável dependente (rótulo para cada dado do conjunto). Para a criação do *ranking*, faz-se uso de uma função de avaliação aplicada sobre os valores que cada dado assume em cada atributo e sobre os valores da variável dependente associada a cada dado.Por padrão, assume-se que o valor mais alto é o indicativo de um atributo mais relevante e os resultados dos atributos são ordenados de maneira decrescente de acordo com o resultado da função de avaliação. Na utilização do *ranking* para construção de preditores, os subconjuntos de atributos são criados progressivamente por meio da incorporação dos atributos em ordem decrescente de relevância.

Um outro ponto a ser tratado no processo de seleção de atributos é a influência dos atributos redundantes (ou perfeitamente correlacionados) sobre o desempenho dos preditores. Há na literatura relatos (GUYON; ELISSEEFF, 2003; KOHAVI; JOHN, 1997) indicando que a remoção dos atributos perfeitamente correlacionados, geralmente, resulta na construção de preditores de melhor desempenho.

Vários trabalhos utilizam filtros como um método de referência (BEKKERMAN et al., 2003; CARUANA; SA, 2003; WESTON et al., 2003). Conforme citado por Hastie, Tibshirani e Friedman (2009), estatisticamente, os métodos de filtro são robustos contra o sobreajuste. Tal como citado por Guyon e Elisseeff (2003), esses métodos são eficientes computacionalmente, pois requerem uma execução para cada um dos atributos existentes e a ordenação dos resultados.Usando como referência a classificação de Kohavi e John (1997), o *ranking* de atributos é um dos tipos dos métodos de filtro.

Nesse trabalho, foi aplicado um método de filtro baseado em análise de correlação. Em uma primeira etapa foram avaliadas as correlações entre os atributos com o objetivo de remover os atributos perfeitamente correlacionados. Na etapa seguinte, cada atributo foi avaliado individualmente de acordo com sua correlação com o atributo dependente (isto é, o tempo para conclusão do incidente). A análise de correlação utilizou a estatística *eta ao quadrado* (η^2) para cálculo com atributos categóricos e o *coeficiente de correlação de Pearson* (R) para os atributos contínuos. Os resultados obtidos foram então ordenados de maneira decrescente para criação do *ranking*.

Segundo KENNEDY (1970), o coeficiente eta (η), originalmente proposto por Karl Pearson como uma medida da relação entre uma variável categórica e contínua, foi reintroduzido como uma medida a posteriori para ANOVA (KERLINGER, 1964; COHEN, 1973). No caso da situação em estudo neste trabalho, o "one-way ANOVA" (uma única variável categórica independente), a interpretação clássica de eta pode ser aplicada. Ou seja, o coeficiente eta ao quadrado (η^2) serve como um índice descritivo que, para um dado conjunto de dados, pode ser usado para avaliar a extensão em que a variância na variável dependente é explicada pela variável independente.

A fórmula para cálculo proposta por Kerlinger (1964), é:

$$\eta^2 = \frac{SS_{effect}}{SS_{total}}$$

,

e o valor do η pode ser calculado como:

$$\eta = \sqrt{\frac{SS_{effect}}{SS_{total}}}$$

,

sendo que SS_{effect} é a soma dos quadrados das diferenças entre os dados de um grupo e a média deste grupo, sendo o grupo formado a partir da variável categórica independente; e SS_{total} é a soma dos quadrados das diferenças entre cada dado da amostra e a média da amostra.

2.2.2 Invólucro

Nos métodos de invólucro, a seleção de atributos é realizada por meio da interação com uma interface do modelo de aprendizado escolhido (neste trabalho o STA), que é

visto com o conceito de caixa preta. Efetivamente, há um espaço de estados que precisa ser explorado utilizando alguma estratégia de busca. A busca é dirigida pela acurácia obtida com a aplicação do modelo de aprendizado em cada um dos estados, nesse trabalho, considerando a combinação de atributos e eventualmente outros parâmetros como o horizonte do log de eventos e o tipo de abstração. Frequentemente há duas formas mais comuns de inicializar o processo de busca: a seleção incremental (do inglês *forward selection*) que parte de um conjunto vazio e acrescenta atributos gradativamente e a outra opção é a remoção seletiva (do inglês *backward elimination*) que parte do conjunto completo de atributos e vai eliminando os atributos gradativamente. Nesse trabalho serão utilizadas duas técnicas de busca amplamente conhecidas:

- Subida da encosta (do inglês *hill-climbing*) é uma das técnicas de busca mais elementares; A busca é feita pela expansão do estado atual com a geração de novos estados e a movimentação na direção do estado com a melhor acurácia. A busca é interrompida quando nenhum dos novos estados (estados filhos) consegue apresentar melhoria na acurácia sobre o estado atual.
- A busca pela primeira melhora (do inglês *Best-first*) difere da subida da encosta no sentido que o processo não é interrompido quando deixa de haver incremento sobre o estado atual, mas quando não há incremento em um número pré-determinado de passos de expansão. Isto significa que mesmo que não exista uma melhoria no estado atual, a busca tenta realizar a expansão do estado com a melhor avaliação lista de estados com expansão em aberto (KOHAVI; JOHN, 1997).

2.3 Algoritmos genéticos

John Holland, em seu livro "*Adaptation in Natural and Artificial Systems*" (HOLLAND, 1975), apresentou uma estrutura geral para representar todos os sistemas adaptativos (naturais ou artificiais) e então mostrou como um processo evolutivo pode ser aplicado a sistemas artificiais. Qualquer problema com características adaptativas pode, geralmente, ser formulado em termos genéticos. Uma vez formulado nesses termos, o problema pode ser, frequentemente, resolvido por meio de um algoritmo genético.

Os algoritmos genéticos (AG) são uma família de modelos computacionais inspirados na evolução, que incorporam uma solução potencial para um problema específico numa

estrutura semelhante a de um cromossomo e aplicam operadores de seleção e cruzamento a essas estruturas de forma a preservar informações críticas relativas à solução do problema. Normalmente os AGs são vistos como otimizadores de funções, embora a quantidade de problemas para o qual os AGs se aplicam seja bastante abrangente (KOZA, 1996).

Uma das vantagens de um algoritmo genético é a simplificação que eles permitem na formulação e solução de problemas de otimização. AGs simples normalmente trabalham com descrições de entrada formadas por cadeias de bits de tamanho fixo ou variável. Eles possuem um paralelismo implícito decorrente da avaliação independente de cada uma dessas cadeias de bits.

Uma implementação de um AG começa com uma população aleatória de cromossomos. Essas estruturas são avaliadas e associadas a uma probabilidade de reprodução de tal forma que as maiores probabilidades são associadas aos cromossomos que representam uma melhor solução para o problema de otimização A avaliação (do inglês *fitness*) da solução é tipicamente definida com relação à composição da população corrente e ao final do procedimento a melhor avaliação é retornada.

A representação de um indivíduo em um AG é determinada pela necessidade de seu emprego em determinado problema, podendo ser feita por meio de uma string, um conjunto de bits ou até mesmo uma árvore. A literatura mostra, em geral, sua representação como uma sequência binária (MICHALEWICZ, 1996). A codificação geralmente é dividida em codificação por estruturas binárias, números reais e números inteiros. A escolha da codificação a ser usada é de fundamental importância para o sucesso na execução do método

Os operadores genéticos têm por objetivo realizar transformações em uma população, fazendo com que, a cada nova geração, indivíduos cada vez mais capazes sejam criados, contribuindo assim para que as populações evoluam a cada nova geração. Com isto, os operadores genéticos são classificados em: inicialização, avaliação, seleção, reprodução, cruzamento, mutação, atualização e finalização (MITCHELL, 1996). Destes operadores, destacam-se os de seleção, cruzamento e mutação, responsáveis por conduzirem a busca da melhor solução. Com a finalidade de entender melhor o contexto em que se constroem os algoritmos genéticos, breves explicações para cada um deles são dadas como segue:

- Inicialização: A inicialização básica de um algoritmo genético clássico se resume à síntese de uma população inicial, sobre a qual serão aplicadas as ações dos passos

subsequentes do processo. Tipicamente faz-se uso de funções aleatórias para gerar os indivíduos, sendo este um recurso simples que visa fornecer uma maior diversidade e fundamental para garantir a abrangência do espaço de buscas. Há alternativas ao método aleatório, destinadas a contornar dificuldades existentes quanto à criação aleatória de indivíduos em representações mais complexas e à melhora no desempenho. Como exemplo, considere o uso de algoritmos de busca heurística como geradores de populações iniciais, especialmente em casos que apresentem um alto grau de restrições, no quais o AG recebe uma população que ainda não possui indivíduos ótimos, mas que apresentam pelo menos algumas das características desejadas. Os operadores de inicialização mais tradicionais são, segundo GOLDBERG (1989), Geyer-Schulz (1996):

  - Inicialização aleatória uniforme: cada gene do indivíduo receberá como valor um elemento do conjunto de alelos, sorteado de forma aleatória com distribuição uniforme;
  - Inicialização aleatória não uniforme: determinados valores a serem armazenados no gene são escolhidos com uma probabilidade maior do que os demais;
  - Inicialização aleatória com "dope": indivíduos otimizados são inseridos em meio à população aleatoriamente gerada. Esse tipo de abordagem pode causar convergência prematura.

- Avaliação: Nesta etapa, cada indivíduo da população é avaliado para que seja determinado o seu grau de adaptação. Nos problemas de busca e otimização deve-se também determinar o quão boa é uma solução (indivíduo), para que se possa definir se ele contribuirá para a resolução do problema. Esse trabalho é realizado pelo operador função de avaliação (do inglês *fitness*). Assim, este operador fornece uma medida de desempenho no contexto de um conjunto de parâmetros atribuindo uma nota para cada cromossomo de acordo com o problema. Esta nota é posteriormente utilizada no operador genético de seleção. O cálculo da avaliação é o elo entre o AG e o problema proposto e deve ser capaz de identificar todas as restrições e objetivos, ou seja, a função de deve ser específica para cada problema. Atualmente, várias formas de avaliação são utilizadas: em casos de otimização de funções matemáticas, o próprio valor de retorno é utilizado; em problemas com muitas restrições as funções

baseadas em penalidades são mais comuns. A função de avaliação também é chamada de função objetivo ou função *fitness* em um grande número de trabalhos.

- Seleção: É no estágio de seleção que os indivíduos são escolhidos para posterior reprodução, cruzamento ou mutação. Neste ponto, fazendo uso do grau de adequação de cada um, é efetuado um sorteio no qual os mais aptos possuem maior probabilidade de se reproduzirem. Este grau é calculado a partir da função de avaliação de cada indivíduo, e determina o quão apto ele está para a reprodução em relação à população a qual ele pertence. Alguns dos métodos mais utilizados são:
  - *Ranking*: os indivíduos da população são ordenados de acordo com seu valor da função da avaliação e então sua probabilidade de escolha é atribuída conforme a posição que ocupam;
  - Roleta: o método de seleção por roleta utiliza o cálculo do somatório da avaliação da população (total) e distribui os indivíduos de acordo com sua proporção nesse intervalo; sorteia um valor aleatoriamente que pertence ao intervalo $[0; total]$ e seleciona o indivíduo que corresponda à faixa do intervalo sorteado;
  - Torneio: Grupos de indivíduos são escolhidos sucessivamente e os mais adaptadas dentro de cada um destes grupos são selecionados (GOLDBERG, 1989; GEYER-SCHULZ, 1996);
- Reprodução:A reprodução é a operação responsável por aplicar o processo de seleção de acordo com o critério parametrizado (geralmente roleta, *ranking*, etc) na escolha dos indivíduos que farão parte da geração seguinte. Dependendo do valor dessa taxa, a população pode convergir mais lentamente (valores altos), pois haverá um limite na inserção de diversidade ou poderá ocorrer a perda de material genético de boa qualidade (valores baixos) dada a alta probabilidade de troca de material para a geração de novas estruturas. Sua aplicação é definida na probabilidade dada pela taxa de reprodução.
- Cruzamento: O cruzamento é o operador responsável pela recombinação de características dos pais durante a reprodução, permitindo que as próximas gerações herdem essas características. Ele é considerado o operador genético predominante, por isso é aplicado com probabilidade dada por uma taxa de cruzamento, que geralmente é muito maior que a taxa de mutação. Este operador pode ser utilizado de várias maneiras e as mais utilizadas são:

  - Ponto único: um ponto de cruzamento é escolhido e a partir deste ponto as informações genéticas dos pais serão trocadas. As informações anteriores a este ponto em um dos pais são ligadas às informações posteriores à este ponto no outro pai;
  - Múltiplos pontos: é uma generalização da proposta de troca de material genético através de ponto único de cruzamento, com a utilização de um valor maior que um para os pontos de cruzamento;
  - Uniforme: para cada alelo a ser preenchido nos cromossomos filhos, o operador de cruzamento uniforme seleciona de forma aleatória qual dos pais deve ser utilizado.

- Mutação: A operação de mutação consiste na alteração de um ou mais genes visando assim a geração de material genético diversificado e, por consequência, a obtenção de novos indivíduos modificados a partir de um previamente escolhido como base. Permite a fuga de um espaço de busca limitado evitando a estagnação na permanência de um mínimo local, contudo, uma taxa muito elevada, a busca se torna essencialmente aleatória e assemelhada a um procedimento de busca exaustiva. Sua aplicação é definida na probabilidade dada pela taxa de mutação.
- Atualização: os indivíduos resultantes da aplicação de um operador genético - reprodução, cruzamento ou mutação - são inseridos na população nova, segundo a política adotada pelo AG e seus respectivos parâmetros. Na forma mais tradicional do AG, a população mantém um tamanho fixo e os indivíduos são criados em mesmo número que seus antecessores e os substituem por completo. Há outras alternativas a essa abordagem, por exemplo: o número de indivíduos gerados pode ser menor, o tamanho da população pode sofrer variações e o critério de inserção pode ser variado de acordo com a avaliação e evolução da população; o conjunto de x indivíduos com melhor avaliação pode ser mantido (elitismo).
- Finalização: A decisão de encerramento de execução do AG é feita utilizando alguns critérios, sendo os mais comuns: o número máximo de gerações; o alcance de um valor ótimo ou sub-ótimo pré-estabelecido; quando o algoritmo não apresenta melhora nas avaliações do melhor indivíduo ou da soma dos indivíduos durante um determinado número de gerações.

O algoritmo genético possui um conjunto de parâmetros que devem ser analisados buscando inferir sobre como os mesmos podem influenciar no comportamento do AG. Dentre os parâmetros destacam-se os seguintes:

- Tamanho da população: representa o número de indivíduos que participarão do processo de evolução. Um valor pequeno gera maior possibilidade do obtenção de resultados como máximos locais. Com um número maior, o espaço de soluções avaliadas torna-se maior e por consequência mais possibilidades de alcançar um ponto de ótimo, porém, como efeito colateral há o aumento do tempo de execução e consumo de recursos computacionais.
- Número de gerações: define o número máximo de gerações no qual o algoritmo genético vai criar novas populações e seguir com o processo de busca;
- Taxa de reprodução: probabilidade com a qual o operador de reprodução tem possibilidade de ser selecionado no processo de geração da nova população;
- Taxa de cruzamento: probabilidade com a qual o operador de cruzamento tem possibilidade de ser selecionado;
- Taxa de mutação: probabilidade com a qual o operador de mutação tem possibilidade de ser selecionado.

Segundo a definição de Koza (1996), os três passos na execução do algoritmo genético simples podem ser resumidos da seguinte maneira:

1. Crie aleatoriamente uma população inicial de indivíduos;
2. Execute iterativamente os seguintes passos na população até o critério de parada seja atingido:
    a) Obtenha o valor de avaliação para cada indivíduo na população.
    b) Crie uma nova população de indivíduos aplicando pelo menos as duas primeiras das três operações descritas a seguir. As operações são aplicadas nos indivíduos da população escolhida com uma probabilidade baseada na avaliação:
        i. Copie os indivíduos existentes para a nova população;
        ii. Crie dois novos indivíduos recombinando geneticamente via cruzamento os dois indivíduos escolhidos aleatoriamente;
        iii. Crie um novo indivíduo a partir de um existente pela aplicação do operador de mutação.

3. O indivíduo mais bem avaliado em qualquer geração é designado como o resultado da execução do algoritmo genético. Esse resultado pode representar uma solução ótima ou aproximada para o problema.

2.4 ITIL - gestão de incidentes

A *Information Technology Infrastructure Library*, (ITIL) é um conjunto de boas práticas para serem aplicadas na infraestrutura, operação e gerenciamento de serviços de tecnologia da informação (ITSM). Foi desenvolvido no final dos anos 1980 pela CCTA (*Central Computer and Telecommunications Agency*), hoje OGC (*Office for Government Commerce*) do Reino Unido. O ITIL pode ser dividido em três grupos: processos estratégicos, táticos e operacionais, totalizando 26 processos. O ITIL V3, publicado em maio de 2007, e atualizado em 2011, é composto de cinco volumes: Estratégia de Serviço, Desenho (ou Projeto) de Serviço, Transição de Serviço, Operação de Serviço e Melhoria Contínua de Serviço.

No grupo de "Operação de Serviço" que se coordena e realiza as atividades e processos necessários para fornecer e gerenciar serviços em níveis acordados com o usuário e clientes do negócio. Os processos descritos nesse grupo são: Gerenciamento Incidentes, Gerenciamento de Eventos, Gerenciamento de Problemas e Gerenciamento de Acessos. O processo de gerenciamento de incidentes é o mais adotado pelas empresas (MARRONE et al., 2014).

Devido à sua criticidade, incidentes precisam receber um tratamento eficiente, e o processo de gerenciamento de incidentes busca organizar as ações a serem executadas de modo que seja possível reestabelecer o serviço a um patamar de qualidade aceitável no menor tempo possível.

Baseado no impacto que pode causar e por consequência na urgência a ele relacionada, geralmente há um prazo limite esperado para a resolução do incidente. Entretanto, a variedade de fatores que estão envolvidos no processo de gerenciamento de incidentes torna difícil o acompanhamento da execução das diferentes instâncias desse processo com o fim de monitorar o tempo que está sendo gasto (e o quanto ainda é necessário gastar) para chegar ao fim do processo com sucesso.

2.5 Estado da arte - mineração de processos operacionais e ITIL

A área de tecnologia da informação tem buscado aprimorar seus processos operacionais, fato que pode ser comprovado pela adoção *frameworks*, como o ITIL, citado anteriormente. Dessa forma, a revisão do estado da arte foi conduzida, sob o formato de revisão sistemática de literatura, para avaliar os trabalhos existentes que abordam a aplicação de técnicas de mineração de processos para resolução de situações especificas da área de tecnologia e dos processos suportados pelo ITIL. Também foram incluídas situações e estudos que versam sobre processos operacionais que possuam similaridade em conceito com o processo de incidentes. Os trabalhos descritos a seguir foram os mais relevantes, no contexto desse trabalho, resultantes da revisão realizada.

Em 2013, foi lançado o "*BPI challenge 2013 - Applied process mining techniques for incident and problem management*". Esse contexto foi gerado para que pudessem ser propostas soluções que permitissem a identificação e melhorias na área de TI da Volvo na Bélgica. Bautista et al. (2013), realizaram um estudo dos logs e apresentaram um detalhamento das três áreas de mineração de processos, essencialmente focado nos registros de log disponibilizados. Foram utilizados a DISCO (FLUXICON, 2018), o Microsoft Excel e o RStudio para construção dessa avaliação. Já o estudo de Dudok e Brand (2013), foi voltado ao processo de incidentes. Spiegel, Dieltjens e Blevi (2013), apresentaram uma avaliação extremamente profunda do processo de incidentes, onde ressalta-se um número significativo de indicadores extraídos. Os três trabalhos citados, são exemplos de uma aplicação relacionada aos processos de incidentes e problemas, porém, focam apenas em aplicar ferramentas e técnicas já utilizadas e conhecidas para descoberta e conformidade, sem informações relacionadas a estimativas ou predições.

Lamine et al. (2015), apresenta um trabalho voltado à avaliação do processo de atendimento de emergências medicas na França (SAMU) com a utilização de mineração de processos atuando nos registros de log das centrais de atendimento. Esses logs foram processados com a ferramenta DISCO e utilizados para obtenção de um modelo atual e identificação de pontos de melhoria. A contribuição mais significativa está na forma de proposição do processo de melhoria, feita pela utilização de técnicas de simulação de eventos discretos para testar os cenários de proposição dos processos melhorados. Apesar

de não abordar processos de tecnologia, há similaridades entre o processo de incidentes e de ocorrências médicas por conta de sua característica forma aleatória e imprevisibilidade.

O trabalho de Bevacqua et al. (2014), faz a proposição uma nova arquitetura de análise de processos de negócio, onde os modelos de performance preditiva obtidos a partir do processo de aprendizado são utilizados como base para o provisionamento de um processo avançado de análise de funcionalidades e monitoramento de performance. A informação é capturada com a identificação de padrões, modelos de cluster preditivos e regressões baseados em clusterização e segmentações baseados em contexto. Os modelos podem ser utilizados para implementar serviços avançados de previsões e são capazes de estimar em tempo real os resultados de novas instâncias de processos e também gerar notificações de possíveis violações de acordos de nível de serviço (do inglês *Service Level Agreement*) com antecedência. Utiliza técnicas como o algoritmo kNN, árvores de decisão e regressões não paramétricas.

Em Polato et al. (2014): uma vez que há uma ampla variedade de fatores que influenciam a predição do tempo de término, esses autores propõem enriquecer o sistema de transições anotado com informações sobre data e hora e executar a predição por meio de uma combinação da probabilidade de ocorrência das próximas atividades (utilizando o "Naive Bayes") com um modelo de regressão (através da Regressão por Vetores de Suporte, com Kernel Polinomial e Radial). Os experimentos foram executados em um log de eventos real (com dois períodos de cobertura com respectivamente 1.500 e 5.000 traços). Os autores reportaram melhorias de 25%, 30% e 50% na acurácia da predição no log menor quando comparados com o modelo de transições anotado original.

Müller et al. (2013), apresenta o foco em uma evolução da mineração de processos, que é a descoberta de serviços. Também trata das quatro dimensões de qualidade (AALST, 2011) - fitness, simplicidade, precisão e generalização - e propõe uma técnica para reduzir o espaço de busca para um contexto finito. É implementada uma técnica utilizando algoritmos genéticos, na forma de protótipo. São realizados experimentos em diversos modelos de serviços de padrão da indústria e demonstra-se que o algoritmo encontra a solução próxima da ótima em tempo aceitável, segundo os autores.

Uma técnica estudada por Naseri e Ludwig (2013), são os Processos de decisão markovianos parcialmente observáveis (sigla em inglês, POMDP). Usa ainda programação dinâmica, *Simple Additive Weighting* (SAW) e o estimador por máxima verosimilhança. Os autores apresentam a argumentação de que a composição de serviços pode ser entendida

como um problema de planejamento devido à natureza dinâmica desses casos. Os métodos mostraram desempenho similar a outros já tratados e os autores informam que pode ser visto como trabalho em estágio inicial, com possibilidade de evolução tanto em performance quanto na utilização mais ampla.

Folino, Guarascio e Pontieri (2012). propõem a utilização de agrupamentos (do inglês *clustering*) para analisar e detectar diversos tipos distintos de contextos de execução de processos em logs de eventos de processos reais da área logística e transformar essa análise em um modelo de predição. São utilizadas árvores de predição de *clusters* (sigla em inglês, PCT), sob a forma de árvores de decisão. Um modelo de predição de performance é criado e as predições são viabilizadas por meio dos sistemas de transição de estados anotado. Um mapeamento dos resultados informa que a abordagem pode contribuir na identificação de violações nos acordos de níveis de serviço (SLA). Os resultados foram considerados eficazes pelos autores.

No estudo de Weerdt et al. (2012), é proposta uma metodologia para mineração de processos com uma combinação de "trace cluster" e mineração de textos. O agrupamento dos traços é usado para separar os logs de execução em diferentes grupos para os quais um modelo de processo mais preciso pode ser descoberto. Em seguida, uma combinação de mineração de textos e árvore de decisão é empregado para obter informações sobre comportamentos atípicos existentes no sistema. A solução foi implementada como plugin do ProM (VERBEEK et al., 2011). A abordagem semi-supervisionada adotada apresenta resultados importantes na identificação de atributos, ainda que às custas de alta complexidade computacional.

Outra abordagem diversa é a utilizada por Abbaci et al. (2011), na seleção e ranqueamento de serviços. As preferências de usuários são modeladas como predicados fuzzy e quantificadores linguísticos são utilizados para modelar a similaridade entre processos. Os resultados são apresentados como uma abordagem possível de utilização no processo de exploração nas etapas de descobertas em mineração de processos.

Um trabalho realizado por Liu et al. (2011), faz uso de modelos de predição baseados em series temporais. Nesse caso, as amostras referem-se à duração histórica de outras instâncias de *workflows*. Essas series são de certa forma intercambiáveis. Nesse trabalho, ao invés de aplicar modelos tradicionais multivariados, é feita uma análise de séries temporais univariadas que possibilitam analisar o comportamento das series de modo a construir um modelo de correlação entre as amostras mais próximas (vizinhança). Em outras palavras,

prever as durações de atividades futuras baseadas nas durações de atividades anteriores. Foi utilizado um padrão de series estatísticas temporais chamado de K–MaxSDev(L) onde o L significa longa duração. Segundo os autores, o K–MaxSDev é capaz de atingir as melhores performances com a descoberta em termos de um padrão potencial.

O trabalho de Rosso-Pelayo et al. (2010), introduz uma técnica para determinar e detectar regras, padrões e relações causa-efeito aplicadas a atividades de processos. O propósito do trabalho é apresentar uma alternativa baseada na análise de informação não estruturada, focada a suportar a avaliação e execução de um traço em processos de negócio. É proposta uma forma de executar mineração de processos utilizando dados não estruturados ao invés de registros de log gerados pelas aplicações com um *framework* de classificação de processos (do inglês, process classification *framework* - PCF). A técnica é composta de duas partes: O objetivo da primeira etapa é obter a associação entre documentos e processos e é composto de três procedimentos principais: Detecção e classificação das atividades relacionadas a atividades de processos e refere-se a uma preparação previa para mineração de textos, sendo que os processos envolvidos já são conhecidos; o segundo procedimento envolve a construção de uma linguagem de modelagem estatística (em inglês, *Statistical Language Model*); A etapa final consiste na análise de uma regressão logística para encontrar associações entre documentos e processos. A segunda parte, tem como objetivo a identificação de regras relacionadas às atividades dos processos que estão presentes nos documentos. Segundo os autores, os testes iniciais apontam que o método produz um modelo aceitável, porém, ainda carece de uma evolução nos estudos para comprovação de sua viabilidade.

Aalst, Schonenberg e Songa (2011): Os STAs possuem perspectivas alternativas para a representação de estados que permitem tratar os problemas de sobreajuste e subajuste, que são frequentes nas tarefas de predição. Os experimentos foram realizados em um log de eventos sintético (400 traços) e dois logs de eventos reais (com 796 traços e 5.187 eventos; outro com 1.882 traços e 11.985 eventos). Os autores concluíram que o modelo proposto de predição supera as abordagens baseadas em heurística simples.

Rogge-Solti, Vana e Mendling (2015) e Rogge-Solti e Weske (2015): propõem a modelagem de series temporais com Redes de Petri, que tem a possibilidade de integrar fluxo de controle com a predição para series temporais e permitem resolver tarefas que consideram os aspectos temporais de um processo; modelos de densidade probabilísticos são propostos para fazer a predição do tempo de execução de um traço e para fazer a

estimativa do risco de ultrapassar a previsão limite estimada. Como item mais importante, os autores propuseram observar o tempo transcorrido desde o último evento observado anterior ao momento da predição'. Foram executados testes de acurácia (com logs de eventos reais) e de escalabilidade (com modelos gerados com uma rede de Petri estocástica distribuída, modificada aleatoriamente com a inserção de tarefas sequenciais, paralelas e caminhos exclusivos de bloqueio). Com os resultados obtidos, os autores apontaram a superioridade de sua proposta para predição de tempo em traços que tem uma duração longa.

Berti (2016): o termo em inglês *concept drifts* tem sido usado para identificar e tratar a natureza dinâmica dos processos. O algoritmo considera uma medida parcial de similaridade entre os traços (dentro de um intervalo de tempo específico), que tem o objetivo de calcular o quanto um traço mais antigo é adequado para gerar informação sobre um traço atual. A abordagem sugerida foi testada em um log de eventos conhecido e foi capaz de superar, para alguns casos de testes, a qualidade de predição de outros métodos que utilizam apenas uma avaliação puramente estatística do processo.

Hinkka et al. (2018): trata da "seleção de características estruturais em logs de eventos" e consiste na criação de características estruturais, tais como: contagem de ocorrências de atividades e transições e suas respectivas ordenações de modo que possam ser utilizadas como recursos para a tarefa de classificação. Os autores defendem que o objetivo da seleção de características é reduzir a dimensionalidade e a complexidade computacional dos métodos a serem aplicados. Informam também que pode levar a uma melhor precisão de previsão e são menos propensos a gerar sobreajuste. São avaliados seis tipos de técnicas de seleção que foram implementados, na linguagem R - Seleção Aleatória, Agrupamento de características, Mínima redundância e máxima relevância, LASSO, Markov Blanket, Importância de variáveis e Eliminação recursiva. Os experimentos são realizados com dois conjuntos de dados conhecidos da literatura (Rabobank Group ICT BPI Challenge 2014 e Dutch academic hospital) e utilizam a divisão dos conjuntos de dados em treinamento e testes. As avaliações para os cenários são realizadas para prever se a duração de um caso pode ser superior a 7 dias e no outro cenário se a solicitação é uma requisição de informação ou um incidente. Para o primeiro caso, a acurácia obtida chega a 82% com os melhores resultados no algoritmo de Eliminação recursiva. No segundo caso, a acurácia chega a 84%. Os valores obtidos para acurácia são muito próximos aos obtidos com a utilização de todos os atributos. Como conclusão apontam que a seleção de

características estruturais pode fornecer meios adicionais para melhorar a precisão das classificações feitas para casos registrados em log de eventos.

Evermann, Rehse e Fettke (2017): apresenta uma abordagem baseada na utilização de "Deep learning" – Redes neurais recorrentes e LSTM (do inglês, *Long Short Term Memory*) – para predição da próxima atividade e do tempo de conclusão. Essa abordagem foi avaliada com a aplicação a dois conjuntos de dados conhecidos na literatura. Os resultados reportados pelos autores são descritos como robustos e suplantando os demais apresentados como estado da arte, porém, com dificuldades para realizar a comparação direta, dada as diferentes métricas aplicadas nos demais trabalhos correlatos. Outro ponto de destaque neste trabalho é a utilização direta no modelo de predição ao invés de fazer a construção de um modelo de processos e a predição a partir desse modelo.

Tax et al. (2017): também faz a utilização de Redes neurais recorrentes e LSTM para predição da próxima atividade e o tempo restante para conclusão. São relatados trabalhos da literatura que apresentam diferenças na acurácia das previsões dependendo do conjunto de dados utilizado. A abordagem de predição do tempo restante é tratada como um caso específico de previsão das próximas atividades. São criadas características para tratar o tempo por dia e semana úteis. A implementação usa o framework Keras e na construção dos experimentos utiliza os dados ordenados de maneira cronológica e separados em treinamento (2/3) e validação (1/3), sendo que primeiros eventos de cada traço não geram predições. Os conjuntos de dados utilizados são: BPI Challenge 2012 e Helpdesk (de uma companhia italiana de software). O modelo STA (AALST; SCHONENBERG; SONGA, 2011) é utilizado como base de comparação e a métrica utilizada é o MAE. A acurácia de classificação é superior à reportada por Evermann, Rehse e Fettke (2017) no mesmo conjunto de dados. No caso da comparação com o STA, os resultados são melhores, exceto para traços com número de eventos reduzido.

As abordagens tem sido frequentemente avaliadas com a utilização de validação cruzada e métricas de acurácia: o erro quadrático médio, do inglês "*Mean Squared Error*" (MSE), a raiz quadrada do erro quadrático médio, do inglês "*Root Mean Squared Error*" (RMSE), o erro médio absoluto, do inglês "*Mean Absolute Error*" (MAE) e o erro médio absoluto percentual "*Mean Absolute Percentage Error*" (MAPE) (ARMSTRONG; COLLOPY, 1992) são os mais utilizados.

Ao analisar o estado da arte, observa-se que o esforço de pesquisa tem-se concentrado na expansão para o suporte operacional - predição e melhorias - com a utilização em

conjunto de técnicas estatísticas, algumas técnicas de aprendizado de máquina, sendo que um número significativo tem atuado na tentativa de tratamento de informações de registros de log parciais ou interação *online*. Nos cenários avaliados, os resultados apresentados mostram uma consistência mais significativa quando utilizam a composição com diversidade de técnicas. Nota-se que os estudos acerca do processo de gerenciamento de incidentes são restritos às avaliações de conformidade e melhorias e poucos casos tratam do suporte operacional.

Embora existam diferentes estratégias abordando a predição do tempo de conclusão, uma lacuna comum nestes trabalhos é a pouca (às vezes falta) atenção na escolha da configuração do log de entrada para a indução do preditor. Um trabalho de pré-processamento para seleção de atributos, conforme proposto neste artigo, tem potencial para melhorar os resultados dos trabalhos relacionados, bem como em outras abordagens relacionadas à indução de preditores.

3 Seleção de atributos em processos de gestão de incidentes

Este capítulo apresenta a estratégia de seleção de atributos para uso na construção de modelos de predição de tempo de execução de instâncias de processos de gerenciamento de incidentes[1]. O intuito é escolher atributos para construir modelos mais acurados. Na literatura da área de mineração de processos, foram propostos modelos que descrevem processos de forma a dar suporte para estimar o tempo restante para conclusão de instâncias do processo descrito. Contudo, devido à variabilidade inerente às instâncias de processo, tais modelos carecem de estratégias que analisam os descritores do processo, escolhem aqueles que de fato influenciam no tempo de execução, e portanto fornecem condições melhores para a realização das estimativas.

O capítulo segue organizado em quatro partes: a primeira (Seção 3.1) diz respeito à modelagem do problema geral de seleção de atributos em um contexto de gerenciamento de incidentes; a segunda (Seção 3.2) descreve o contexto real no qual foi realizado o presente estudo; a terceira parte (Seção 3.3) apresenta a aplicação do STA no contexto da seleção de atributos aplicados ao ambiente de gerenciamento de incidentes; por fim (Seção 3.4) são detalhados os métodos de seleção baseados no conhecimento do especialista, filtro e invólucro com a utilização das definições das seções anteriores. A implementação dessa solução foi executada e está descrita no capitulo 5.

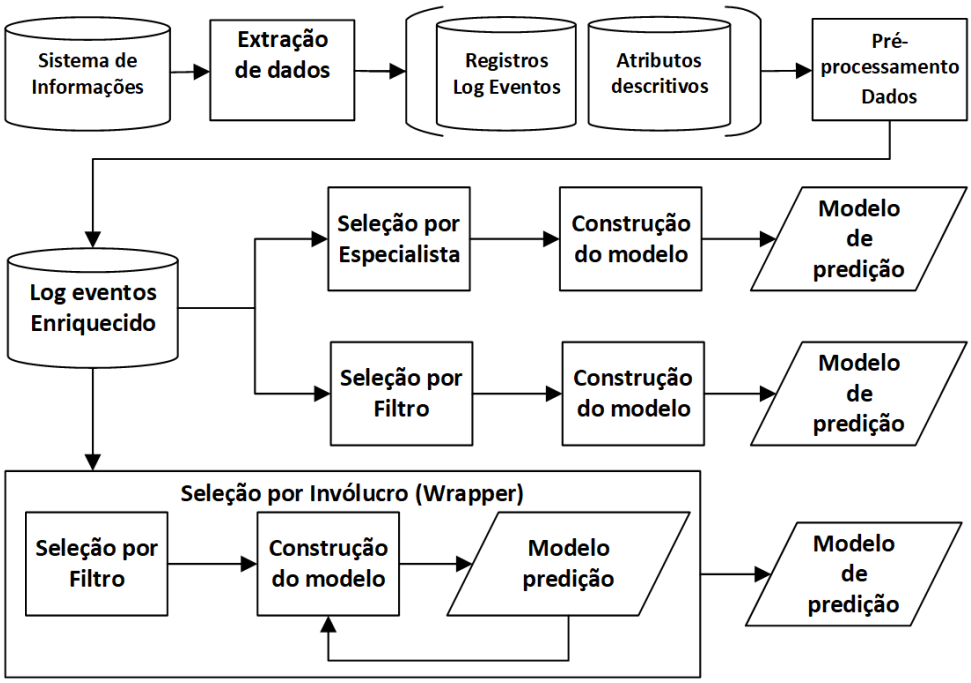
3.1 Modelagem proposta para seleção de atributos

A figura 3 apresenta uma visão geral do contexto de estudo deste trabalho. Na figura estão organizados os elementos que fornecem e aqueles que geram informação na estratégia em estudo, o tipo de informação envolvida em cada passo da estratégia, e os passos em si, com destaque para os que se constituem como a principal contribuição associada a este trabalho.

A fonte de dados primária para este estudo é um sistema de informação que trata do gerenciamento de um processo estruturado ou semiestruturado. No estudo atual, foi utilizada a plataforma $ServiceNow^{TM}$ [2], que é uma plataforma proprietária e implementa o gerenciamento de processos de tecnologia com referência no *framework* ITIL. A partir dela

[1] A partir deste ponto, por simplicidade, usar-se-á apenas tempo para execução do incidente
[2] Informações detalhadas podem ser obtidas em ⟨https://docs.servicenow.com/⟩

Figura 3 – Diagrama geral da solução



Fonte: Claudio Aparecido Lira do Amaral, 2018

é realizada a **extração de dados** do processo. Esses dados dizem respeito às informações sobre o registro do processo em estudo, nesse caso, o processo de incidentes (**atributos descritivos** - dados estruturados) e aos registros do log de auditoria da ferramenta (**log de eventos** referentes ao uso da ferramenta no processo de gerenciamento de incidentes - dados não estruturados). Sobre esses dois conjuntos de dados é executado um conjunto de funções de **pré-processamento** que permitem a geração de um log de eventos enriquecido e correspondem a um "log de incidentes". Esse log viabiliza a implementação da tarefa de mineração de processos (neste trabalho, predição de tempo de execução do incidente) e os procedimentos de seleção de atributos.

O principal interesse do trabalho está concentrado na melhoria dos resultados referentes à estimativa de tempo para execução de um processo até sua conclusão. Isto envolve a aplicação da funcionalidade de **construção do modelo de predição**. Neste trabalho o modelo utilizado é o STA e tem a finalidade de fornecer as estimativas de tempo para conclusão. Ele é gerado[3] conforme as definições de literatura citadas no capítulo 2. Entretanto, a depender de como um processo é descrito, e do quão refinada ou detalhada é essa descrição, o modelo resultante pode oferecer estimativas muito diferentes e pouco precisas (demonstrado nos testes da seção 4.1.1). Desta forma, faz-se justificável o

[3] A implementação foi construída em linguagem R.

estabelecimento de procedimentos para seleção de atributos que forneçam listas de atributos com potencial para minimizar tais diferenças e aumentar as chances de melhorar a precisão da predição obtida com os modelos gerados. A seleção de atributos foi implementada neste trabalho de forma **orientada pelo especialista** e a partir da aplicação de técnicas de seleção de atributos (seção 2.2) do tipo **filtro** e do tipo **invólucro**.

Na seleção orientada pelo especialista, o especialista usa o seu conhecimento sobre o processo de negócio associado (neste caso, o gerenciamento de incidentes) para escolher os atributos que entende serem os melhores para descrever o processo (incidentes) com fins de predição do seu tempo de execução. Então, o modelo (STA) é construído com base no conjunto de atributos selecionado pelo especialista, e as predições podem ser realizadas e avaliadas.

A **seleção por filtro** utiliza o conceito de correlação entre os atributos independentes e o atributo dependente para criação de um "ranking" ordenado do mais correlacionado para o menos correlacionado. Deste modo, é possível ter informações para decidir sobre a relevância de um subconjunto proposto. Após a escolha dos atributos a serem utilizados, os subconjuntos de atributos são criados de modo a utilizar a sequência do mais relevante para o menos relevante. A estratégia segue então da mesma maneira que no caso da seleção orientada pelo especialista com a construção dos modelos de predição para cada um dos subconjuntos definidos.

Na **seleção por invólucro**, o modelo (STA), construído com um subconjunto de atributos sugerido por um processo de busca, é utilizado para gerar predições cujas acurácias conduzirão à seleção de um subconjunto ótimo ou sub-ótimo de atributos. A partir de diferentes listas de atributos, modelos podem ser gerados e suas estatísticas de tempo de execução de instâncias de processos podem ser avaliadas. Seguindo o fluxo de trabalho proposto no diagrama da figura 3 para a seleção por invólucro, essa avaliação retro-alimenta o processo de seleção de atributos, proporcionando a melhoria do processo de seleção de atributos. Para implementar a seleção por invólucro, foram usadas as estratégias de buscas heurísticas *subida de encosta* e *primeira-melhora* e a estratégia de busca *meta-heurística com algoritmos genéticos*. As características específicas de cada método são fornecidas na seção 2.2. De maneira prática, a ideia básica é que o modelo de predição possa ser gerado a partir de um subconjunto de atributos que descreve adequadamente os casos concluídos. A partir deste ponto, esse modelo pode ser aplicado para prever o tempo para conclusão dos novos casos – incidentes, no processo em questão.

Vale ressaltar que, embora uma estratégia genérica para seleção de atributos esteja sendo buscada, o processo de gerenciamento de incidentes é usado neste trabalho como um ambiente de testes, principalmente por ser um gerenciamento crítico em relação a tempo. A solução proposta neste trabalho pode ser utilizada em outros tipos de processos, desde que respeitadas as definições de utilização em processos estruturados ou semi-estruturados e que tenham uma variável dependente contínua relacionado ao tempo (com alguns ajustes, poderia ser utilizados com variáveis dependentes categóricas).

No processo de gerenciamento de incidentes é recomendada a distribuição do tratamento de um incidente de acordo com várias informações: fases do processo de tratamento, perfil dos recursos humanos envolvidos, conhecimento técnico exigido no tratamento, etc. Todas essas informações geram um grande conjunto de atributos associados aos registros dos incidentes, e é inviável usar todos eles na geração dos sistemas de transição, visto que os logs de eventos tornam-se muito grandes e tais sistemas assumem alto grau de ramificação. Um outro comportamento observado é que as instâncias de processos tornam-se muito específicas causando um problema de superajuste para as estimativas. Assim, a questão de criticidade e necessidade de estimativas de tempo precisas (minimização de estatísticas de acurácia e seus respectivos desvios-padrão) e a presença de vários atributos gerando numerosas possibilidades de combinações nas instâncias de processos proporcionam um ambiente adequado para a validação da estratégia de seleção construída neste trabalho.

3.2 Contextualização do ambiente de estudo

Esse trabalho diz respeito a um estudo no contexto do processo de gerenciamento de incidentes e mineração de processos. Para realização desse estudo, foi construído um ambiente de experimentação no qual informações estruturadas associados à incidentes e informações não estruturadas associadas a logs de eventos provenientes do processo de gerenciamento são combinadas. Esse ambiente foi preparado como parte integrante da execução deste trabalho, a partir da plataforma $ServiceNow^{TM}$, usada em uma empresa de tecnologia da informação. O ambiente segue descrito nesta seção e na sequência os dados de incidentes e dos registros de log são detalhados.

3.2.1 Ambiente de gerenciamento de incidentes

O processo de gerenciamento de incidentes suportado pela plataforma $ServiceNow^{TM}$ exige o envolvimento de três atores:

- o solicitante: frequentemente, o solicitante é o indivíduo afetado pela indisponibilidade (ou degradação) do serviço, causada pela ocorrência de um incidente. Ele também é a pessoa responsável por relatar o evento ocorrido, seja em uma interação direta com o sistema de automação do processo de gerenciamento , ou pelo contato com uma central de serviços que recebe requisições referentes às ocorrências;
- o analista: tradicionalmente o analista de uma central de serviços (*service desk*) tem a função de registrar (ou complementar) a informação fornecida pelo solicitante, validar os dados fornecidos e executar o diagnóstico inicial sobre as causas da ocorrência ou sobre o encaminhamento a ser dado ao incidente no processo de gerenciamento;
- o suporte: são grupos de agentes que recebem os incidentes registrados pela central de serviços, atuam na investigação detalhada do incidente, no diagnóstico de suas causas, e indicam ou propõem soluções de contorno até que o serviço seja reestabelecido ou sejam encontradas as soluções definitivas.

O envolvimento desses atores no processo de gerenciamento de incidentes é apresentada na figura 4. Essa atuação está dividida nas seguintes etapas:

1. Identificação e classificação: Nessa etapa é realizado o registro inicial do incidente na ferramenta, a identificação do solicitante, o registro de informações sobre o incidente e sobre o contexto no qual o incidente ocorre (impacto, urgência, categorização, etc). Essas informações são o insumo para a sequência de ações que serão executadas no processo de gerenciamento do incidente e também constituem-se como a entrada para a realização da previsão de tempo para a resolução do evento. No contexto de uso da plataforma, considerado neste trabalho, geralmente os tempos de resolução de um incidente variam de 4 a 48 horas, mas não estão restritos a este intervalo.
2. Suporte inicial: O suporte inicial é feito com a pesquisa por incidentes semelhantes que tenham ocorrido e já estejam registrados na ferramenta. Esses registros podem ter sido reportados por diferentes usuários, a diferentes analistas e tratados por diferentes grupos de suporte, e podem ter ocorrido em um passado recente ou não. Se situações semelhantes são encontradas, soluções semelhantes às aplicadas no passado

podem ser usadas no presente (exigindo adaptações ou não). A ferramenta também possibilita ao suporte inicial, a pesquisa em base de erros/problemas conhecidos e a identificação de soluções de contorno adequadas a determinadas situações.

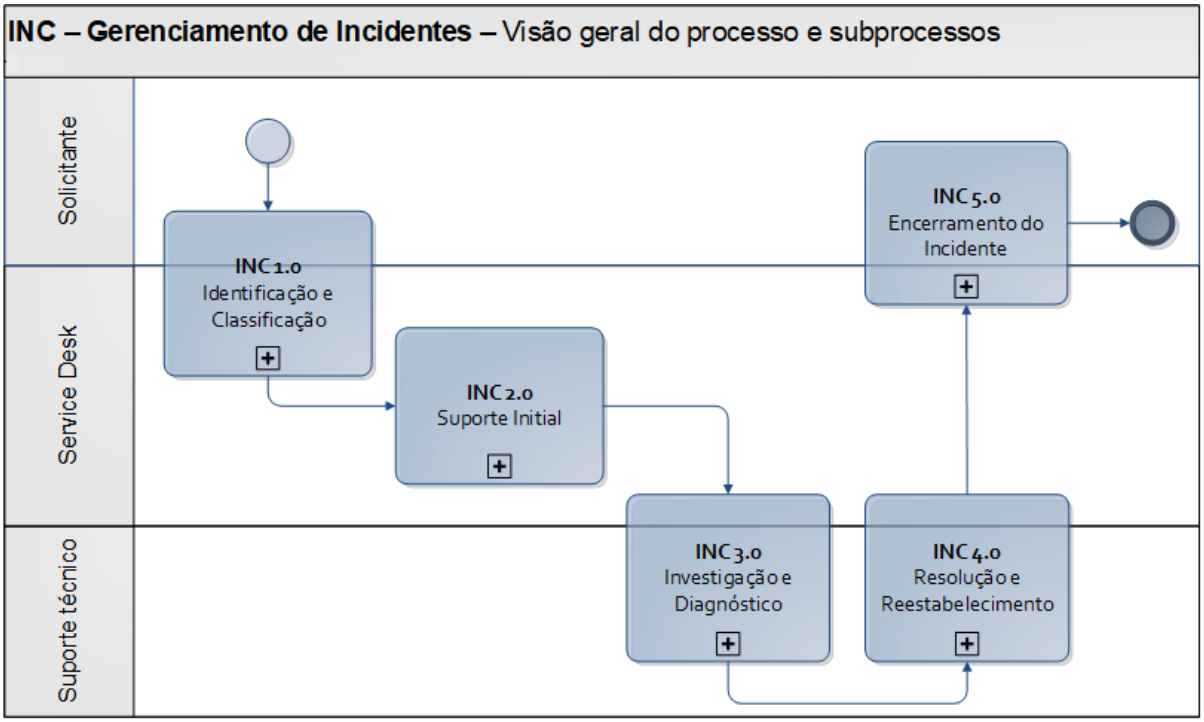
3. Investigação e diagnóstico: A investigação consiste em realizar a aplicação de procedimentos técnicos mais elaborados, fazer uso de outros documentos da base de dados interna dos artigos conhecimento e aplicar o conhecimento de domínio do analista de suporte (ou de especialistas), de modo a identificar qual a causa técnica que gerou o comportamento de indisponibilidade ou degradação do serviço de tecnologia.
4. Resolução e reestabelecimento: Após a identificação das causas do incidente e da solução a ser aplicada, o serviço é reestabelecido a uma condição que possa ser utilizado por seus usuários. Em caso de impossibilidade na identificação das causas ou apresentar uma solução, outras ações podem ser geradas, como: criação de uma requisição de mudanças, registro de um novo problema para tratamento nesse processo, ou acionamento de fornecedores externos para o desenvolvimento de correções.
5. Encerramento: Uma vez que a correção - seja uma solução definitiva ou uma solução de contorno - tenha sido aplicada com sucesso, a solução é registrada na ferramenta, o incidente é considerado resolvido e o solicitante é notificado que o serviço está disponível. Eventualmente, podem existir situações nas quais o processo não leva a uma solução satisfatória, então, o solicitante informa a situação de rejeição da solução e o processo retorna para a etapa de investigação e diagnóstico.

A interação direta ou indireta dos atores, mediante o processo de gerenciamento de incidentes implementado na plataforma $ServiceNow^{TM}$ gera informações referentes aos incidentes (dados estruturados) e ao próprio processo de gerenciamento de incidentes (dados não estruturados - logs). Essas informações compõem um arcabouço de dados que está organizado na plataforma e que foi usado para fins de mineração de processo neste trabalho. As duas próximas seções são dedicadas a descrever esses repositórios de dados.

3.2.2 Dados estruturados - atributos descritivos de incidentes

A informação sobre os registros de incidentes na plataforma $ServiceNow^{TM}$ está armazenada em uma relação denominada *incident*, a qual pertence a um modelo de dados relacional que suporta uma série de outras funcionalidades no sistema. Em relação

Figura 4 – Visão geral do processo de incidentes



Fonte: Adaptado de $ServiceNow^{TM}$, 2018

aos atributos descritivos de um incidente, a $ServiceNow^{TM}$ possui, na implementação utilizada, 91 atributos. A descrição detalhada desses atributos e o modelo relacional da relação *incident* são apresentados no apêndice A. Entretanto, foi necessário realizar uma atividade de pre-seleção dos atributos, pois alguns deles não puderam ser utilizados para os propósitos de mineração de processos deste trabalho por terem valores inconsistentes e/ou incompletos ou representar informação não estruturada de natureza textual. Após a remoção dos atributos indesejados, o conjunto final de atributos descritivos é composto de 34 atributos, sendo 27 categóricos e sete numéricos.

3.2.3 Dados não estruturados - log de eventos do processo de gerenciamento de incidentes

A informação dos registros de log da plataforma $ServiceNow^{TM}$ é armazenada em uma relação chamada *sys_audit* (apresentada em detalhes no apêndice B). Essa relação armazena dados referentes às atualizações realizadas em todas as relações do modelo de dados que estão configuradas para serem auditadas (o que inclui a relação *incident* - interesse deste projeto). Assim, na realidade, trata-se de um sistema preparado para realização de auditoria nos processos por ele suportados. Para os objetivos deste

trabalho, essa relação é suficiente para obtenção do log de eventos referente ao processo de gerenciamento de incidentes. A tabela 2 apresenta exemplos de registros existentes na tabela de log, referentes à atualizações realizadas na relação *incident*. Para cada atualização podem ser gerados um ou mais registros de log. A tabela apresenta os campos mais relevantes para identificação dos registros de atualização.

Tabela 2 – Trecho de log de auditoria referente às atualizações de registros de incidente

field name	new value	old value	updated	created by	created
reassignment_count	1	0	1	usr3@dom1.com.br	25/08/2016 02:25:04
reassignment_count	2	1	2	usr4@dom1.com.br	25/08/2016 08:18:06
assignment_group	34370bb96f119a0041b0d7aabb3ee4cd	05370bb96f119a0041b0d7aabb3ee4f2	2	usr4@dom1.com.br	25/08/2016 08:18:06
activity_due	2016-08-28 11:18:06	2016-08-28 05:25:04	2	usr4@dom1.com.br	25/08/2016 08:18:06
work_notes	O mesmo não possui acesso ao sistema. Para liberação seguir o script de criação de novos usuários	JOURNAL FIELD ADDITION	2	usr4@dom1.com.br	25/08/2016 08:18:06
u_priority_confirmation	1	0	3	usr2@dom1.com.br	25/08/2016 08:52:04
calendar_stc	23280		3	usr2@dom1.com.br	25/08/2016 08:52:04
close_code	Solved (Permanently)		3	usr2@dom1.com.br	25/08/2016 08:52:04
business_stc	0		3	usr2.@dom1.com.br	25/08/2016 08:52:04
resolved_by	91db21136ff9160099def46abb3ee467		3	usr2.@dom1.com.br	25/08/2016 08:52:04
assigned_to	91db21136ff9160099def46abb3ee467		3	usr2.@dom1.com.br	25/08/2016 08:52:04
business_duration	1970-01-01 00:00:00		3	usr2.@dom1.com.br	25/08/2016 08:52:04
resolved_at	2016-08-25 11:52:04		3	usr2@dom1.com.br	25/08/2016 08:52:04
close_notes	Prezado cliente, O Senhor não possui acesso ao sistema XX, para a criação do seu acesso, será necessário o acordo gerencial e login modelo. Após coletar essas informações entre em contato com o YYYY para abertura de um chamado para acesso ao sistema.		3	usr1@dom1.com.br	25/08/2016 08:52:04
estado	6	1	3	usr1@dom1.com.br	25/08/2016 08:52:04
incident_state	6	1	3	usr1@dom1.com.br	25/08/2016 08:52:04

Fonte: Claudio Aparecido Lira do Amaral, 2018

3.2.4 Pré-processamento do log

Para fins de mineração de processos, é necessário combinar os dados descritos nas seções 3.2.2 e 3.2.3. Como resultado dessa combinação, é gerado um arquivo cujo conteúdo contempla os atributos fundamentais para identificação das instâncias do processo de gerenciamento de incidentes, os estados pelos quais o processo passa, as atividades executadas durante o processo, a data e a hora de realização das atividades e outras informações úteis para descrição dos incidentes e potencialmente importantes para mineração de processos. Esse arquivo é chamado neste trabalho de "log de eventos enriquecido".

Para geração dos registros no formato esperado foi desenvolvido um procedimento[4] para ler os arquivos referentes às relações da plataforma $ServiceNow^{TM}$, referentes aos incidentes (relação *incident*) e aos registro de log (relação de auditoria da relação *incident*), e fazer uma transformação para sequenciamento dos registros de log referentes a cada instância do processo de gerenciamento de incidentes. Esse procedimento envolve ações de filtragem para selecionar apenas as informações referentes a incidentes, ações de transformação e derivação de dados, e ordenação temporal dos registros de log por instância de processo.

A lógica para construção dos registros de log de incidentes exige a varredura da relação de log mantida pela plataforma $ServiceNow^{TM}$, agrupando os seus registros por número de incidente e número de *checkpoint*. Cada *checkpoint* consiste em uma atualização de registro, porém, como o log tem a atualização individual de cada campo, é necessário considerar várias entradas do log (linhas) para gerar um novo registro de atualização no log de incidentes. Os valores anteriores e atuais do atributo sob atualização são armazenados de modo que seja possível construir toda a sequência de atualização. Na tabela 3 é apresentado um exemplo de um trecho do log de incidentes usado para os testes realizados neste trabalho. O log de eventos enriquecido é a entrada para a seleção de atributos e é composto por 37 atributos. Uma descrição detalhada dos atributos utilizados neste trabalho é apresentada no apêndice C e inclui: 3 atributos de auditoria, 33 atributos descritivos e o atributo "*closed*" (a variável dependente na predição do tempo para execução do incidente).

Durante a realização desse pré-processamento, foi necessário realizar as seguintes transformações/derivações:

[4] O programa para pré-processamento do log foi escrito em linguagem R.

Tabela 3 – Log de eventos enriquecido

number	**incident_state**	**updated**	**category**	**assignment_group**
	New	2/3/2016 04:57:49	Internet	Field Service
	New	2/3/2016 16:52:02	Internet	Field Service
	Active	2/3/2016 18:13:46	Internet	Field Service
	Active	2/3/2016 19:14:04	Internet	Field Service
	Awaiting UI	2/3/2016 19:14:11	Internet	Field Service
	Awaiting UI	3/3/2016 11:24:18	Internet	Field Service
	Awaiting UI	3/3/2016 11:25:56	Internet	Field Service
	Awaiting UI	3/3/2016 12:33:23	Internet	Field Service
	Awaiting UI	3/3/2016 12:43:11	Internet	Field Service
INC001	Active	3/3/2016 12:43:23	Internet	Field Service
	Active	3/3/2016 12:54:14	Internet	Field Service
	Active	3/3/2016 12:57:14	Internet	Inf. Security
	Active	3/3/2016 13:14:53	Internet	Inf. Security
	Active	3/3/2016 13:16:35	Internet	Service Desk
	Active	3/3/2016 19:57:53	Internet	Field Service
	Active	4/3/2016 10:56:24	Internet	Field Service
	Resolved	4/3/2016 11:02:07	Internet	Field Service
	Closed	9/3/2016 12:00:03	Internet	Field Service

Fonte: Claudio Aparecido Lira do Amaral, 2018

- a plataforma *ServiceNow*^{TM} não registra a inicialização dos atributos (a primeira inserção de dados em um atributo), por questões de performance. Então, foi necessário derivar essa informação a partir dos campos existentes na relação *incident* e dos valores anteriores registrados no primeiro *checkpoint*. Dessa forma foi possível construir o registro completo para o log de incidentes;
- no registro exportado da relação *incident*, o atributo *incidente_estado* estava valorado com informações descritivas, enquanto na relação de log o atributo estado estava valorado com seus identificadores numéricos. Assim, foi necessário fazer uma transformação de valores para padronizar o atributo considerando seu identificador numérico.

3.3 Utilização do Sistema de Transições Anotado

O STA é composto por uma função de representação de estados $\mathrm{l}^{\mathrm{estado}}$ que, dado um traço parcial σ produz uma representação para ele. Essa representação é parametrizada pelo traço de entrada, pelo tipo de abstração escolhida (conjunto, multiconjunto ou sequência), pelo tamanho do horizonte máximo aplicado e pelos atributos que são utilizados para identificar unicamente cada um dos estados (os atributos selecionados do log de eventos

enriquecido). Dessa forma, para adequadamente usar STA como modelo de predição e como meio de avaliação da seleção de atributos, foi necessário: compreender alguns comportamentos decorrentes da criação de STAs considerando a variação da abstração e do tamanho do horizonte; propor uma função de predição; e estabelecer uma forma de avaliação da predição. Essa análise está descrita nesta seção.

3.3.1 Abstrações – conjunto, multiconjunto e sequência

As abstrações – conjunto, multiconjunto e sequência – influenciam a forma de construção da representação dos estados no STA e, consequentemente, o número de estados gerados. Para exemplificar, um caso simples de um traço considerando uma situação genérica, considere

$$\sigma =< A, B, C, B, C, C, C, D >$$

que possui apenas um atributo descritivo, o qual pode assumir os valores $\langle A, B, C$ ou $D\rangle$ (não citando o identificador de tempo) e, considere o uso de um horizonte máximo $h = 3$. O uso das abstrações conjunto, multiconjunto e sequência gerará STAs com respectivamente 5, 7 e 8 estados. A quantidade de estados gerada pela abstração "sequência" é a que produz uma quantidade maior de estados. A tabela 4 apresenta as quantidades por tipo de abstração e de acordo com a variação do horizonte para o atributo *inciden_state*, no conjunto de testes com 56.503 eventos.

Em uma primeira análise, então, poder-se-ia dispensar abstrações mais caras, i.e. que geram mais estados no STA. No entanto, a geração de mais ou menos estados não está necessariamente correlacionada com a qualidade do sistema gerado em termos de acurácia de predição e sobre ou subajuste do modelo aos dados. Sendo assim, as três abstrações devem ser exploradas nos experimentos com seletores de atributos.

Considerando a situação presente no log de eventos enriquecido usado neste projeto, a quantidade de estados gerada em um STA pode variar significativamente dependendo de quais e quantos atributos são utilizados para fazer a identificação do eventos. Por exemplo, se for utilizado um campo de controle como *incidente_state*, cujo domínio é restrito, será produzido um comportamento bem diferenciado daquele produzido se for utilizado um outro campo como o *assigned_to* (analista responsável pelo incidente), pois este possui um domínio muito maior.

Tabela 4 – Estatísticas sobre o número de estados do STA para um log com 56.503 eventos utilizando o atributo *incident_state* como chave para identificação do estado e as três formas de abstração.

Horizonte	**Conjunto**	**Multiconjunto**	**Sequência**
1	8	8	8
3	64	106	150
4	86	215	352
6	101	592	1.100
7	103	840	1.623
Inf	95	3.522	5.023

Fonte: Claudio Aparecido Lira do Amaral, 2018

3.3.2 Horizonte máximo

Outro parâmetro relevante na construção do STA é o horizonte máximo (ou simplesmente, horizonte, cf. apresentado n capítulo 2). Um horizonte que não seja capaz de refletir adequadamente o número de situações distintas em um log de eventos pode fazer com que o STA seja demasiadamente sobreajustado.

A maior parte dos estudos citados no capítulo 2 utiliza valores extremos, ou seja, horizonte = 1 ou infinito (todos os registros do traço). Para essa questão, uma avaliação detalhada foi realizada, envolvendo uma análise de comportamento do STA de acordo com o número de eventos em cada um dos traços que compõem o log de eventos enriquecido, e um conjunto simples de estatísticas (tabela 5) foi derivado desta análise.

Tabela 5 – Estatísticas sobre o número de eventos nos traços presentes no log de eventos enriquecido

Mínimo	**1º Quartil**	**Mediana**	**Média**	**3º Quartil**	**Máximo**
2	3	5	6	7	58

Fonte: Claudio Aparecido Lira do Amaral, 2018

As estatísticas revelam que para a maior parte dos casos, ou seja, até o 3º quartil, o processo de gerenciamento de incidentes segue um comportamento regular. Porém, no último quartil, o número máximo de eventos é alto, o que indica a presença de "*outliers*". Esse comportamento pode fazer com que a precisão do sistema varie significativamente, uma vez que a predição faz uso dos registros do log que são agregados ao estado.

Dadas essas estatísticas, evidencia-se a necessidade de explorar uma quantidade maior de valores para o parâmetro horizonte máximo. Desta forma, os experimentos deste trabalho contemplaram seis valores para o horizonte máximo: os valores usados na literatura (1 e infinito) e os valores apontados na tabela 5, até o 3º quartil. Unindo essa

decisão, à decisão de usar as três abstrações (seção 3.3.1, os experimentos deste trabalho geraram 18 STAs para cada subconjuntos de atributos gerado pelos seletores.

3.3.3 Funções de predição

A função de predição usada no STA, tal como citado na seção 2.1.3, calcula o valor de predição, baseado em um multiconjunto de medições a cada um dos estados durante a construção do sistema. O mais comum é a utilização do valor da média amostral do conjunto de medições referente ao tempo restante até a conclusão das instâncias de processos. Outra métrica frequentemente utilizada é o tempo de permanência (do inglês, "*sojourn time*") da instância do processo em um determinado estado do STA. A métrica de permanência faz sentido quando há um número seguido de atualizações das atividades sem que exista efetivamente uma transição para um próximo estado. Em um modelo de incidentes, esse item pode ser identificado na etapa de investigação e diagnóstico quando o analista permanece registrando no sistema, sob a forma de texto, as avaliações realizadas.

Com base num cenário que considera o tempo restante para execução de um incidente, o tempo de permanência e as estatísticas mais comuns (média e mediana), neste trabalho, é proposta a utilização de duas funções de predição no STA, da seguinte forma:

$$l^{\text{tempo1}}(estado, p) = \overline{r}(estado) + \overline{p}(estado) - p$$

em que $\overline{r}(estado)$ é a média dos tempos restantes do conjunto de medições do estado, $\overline{p}(estado)$ é a media dos tempos de permanência do multiconjunto de medições do estado e p = tempo de permanência no estado da instância para a qual a predição está sendo realizada. E,

$$l^{\text{tempo2}}(estado, p) = med_r(estado) + med_p(estado) - p$$

em que $med_r(estado)$ é a mediana dos tempos restantes do conjunto de medições do estado, $med_p(estado)$ é a mediana dos tempos de permanência do multiconjunto de medições do estado e p = tempo de permanência no estado da instância para a qual a predição está sendo realizada.

As métricas de média e mediana para os tempos de permanência e restante fazem parte do STA construído. A variável **p** é calculada a cada atualização do registro da

instância de processo sob análise e calculada a partir do estado atual e do estado da atualização imediatamente anterior. A razão para propor, neste trabalho, a utilização da média e mediana do tempo de permanência no sistema é proporcionar predições mais realísticas, capazes de adequadamente tratar casos nos quais os estados abrangem um número de instâncias de incidentes distintas com longa permanecia, ou seja, possuem várias atualizações embora não exista alteração no estado atual da instância de processo. Nesse contexto, agrega-se a capacidade de especialização em cada um dos casos sem que haja perda relacionada à generalização.

3.3.4 Procedimentos de avaliação

Assumindo a utilização do STA como um preditor, faz-se necessário estabelecer uma forma de avaliação da qualidade do sistema gerado e da acurácia da predição fornecida. Isso também se faz necessário devido a decisão de usar o STA como parte integrante de um processo de seleção de atributos. Na literatura correlata, a qualidade do sistema é avaliada pela medida de "*non-fitting*". Já a a acurácia da predição é, geralmente, avaliada por meio das medidas MAPE, MSE e RMSPE.

A escolha pela medida MAPE foi feita considerando os estudos de Armstrong e Collopy (1992). Segundo esses autores, a MAPE é uma medida de avaliação sensível às variações dos parâmetros do modelo, capaz de construir uma validação no modelo e gerar um valor relativo normalizado. Também foi considerado o estudo de Myttenaere et al. (2016), que mostra como o processo de aprendizagem guiado pelo MAPE é viável tanto do ponto de vista prático como do ponto de vista teórico, considerando a minimização do risco empírico. Outro fator complementar foi o contexto do processo de gerenciamento de incidentes, em análise neste trabalho, o qual possui um tempo para conclusão medido em segundos, com a mediana em 33.840 segundos (0,4 dia), porém, com variações distintas no quartil superior. Esse contexto requer uma medida de avaliação que uniformize variações. Aliado a isso, não há necessidade de realizar a predição para os estados finais (terminais) do STA, ou seja, quando a instância do processo está concluída e o resultado da predição é zero. Essa seria uma situação na qual o resultado poderia ser distorcido, uma vez que o MAPE apresenta viés elevado nesse tipo de situação. A medida MAPE é dada por:

$$MAPE = \frac{1}{n} * \sum_{t=1}^{n} \frac{|F_t - A_t|}{A_t}$$

,

considerando: n como o número de eventos existente no log; os valores F_t, obtidos com a aplicação da função de predição l^{tempo1} ou l^{tempo2} a cada evento do log; os valores de A_t, que representam o tempo restante calculado no instante em que o evento foi registrado no log.

Além do MAPE, a qualidade do STA, em termos de completude (registros de *non-fitting*) foi avaliada com a contabilização do número de registros de log que não possuem um estado correspondente no STA. Esse indicador – NF – é o índice de não reprodutibilidade do STA. Ou seja, representa os registros de eventos do conjunto de testes que não são mapeados para um estado correspondente no STA criado. Esse valor é um indicativo da capacidade de generalização do modelo construido com os atributos selecionados. Quanto menor esse valor, melhor é a capacidade de tratar eventos que não foram utilizados para construção do modelo STA. O cálculo é realizado da seguinte forma:

$$NF = \frac{m}{n} * 100$$

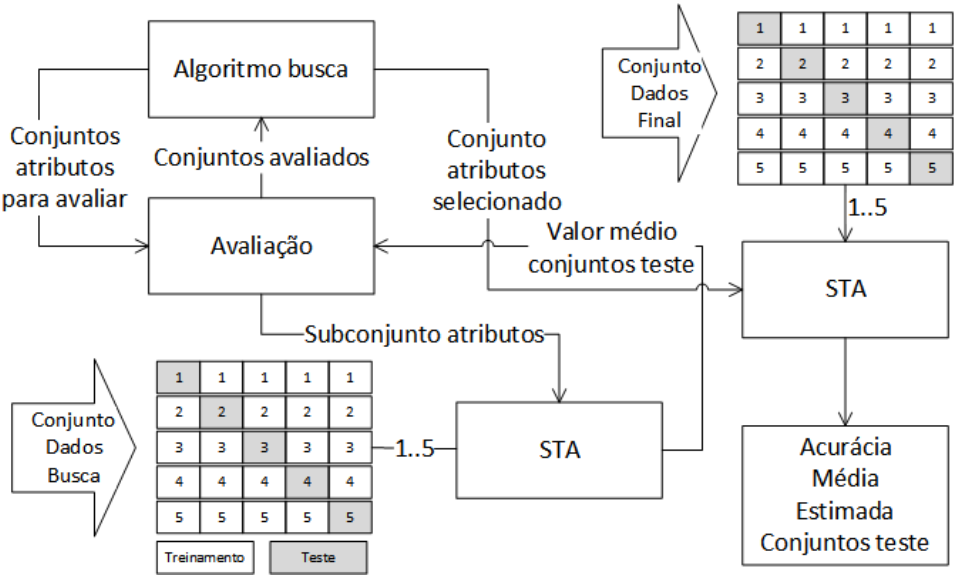
sendo m o número de eventos não reprodutíveis pelo STA (*non-fitting*) e n o número total de eventos no subconjunto de testes.

A proposição para estruturação dos experimentos foi a criação de um procedimento que permite a seleção aleatória de um subconjuntos de instâncias de incidentes a partir do log de eventos enriquecido. Esse subconjunto do log de eventos enriquecido pode ser dividido em **v** partes (**v** sublogs, maiores ou iguais a 2), para ser usado em um método de validação cruzada na construção dos STA. Na figura 5, é apresentada a forma de seleção utilizada pelo invólucro, com a validação utilizando 5 subconjuntos (folds). Esse é o padrão adotado nos experimentos conduzidos e descritos no capítulo 5.

O STA é construido com 4 subconjuntos de dados e testado em 1 subconjunto que é utilizado para fazer a predição e tem o valor do MAPE calculado. Esse procedimento é repetido novamente até que os 5 subconjuntos tenham o valor do MAPE calculado. O cálculo do valor é feito utilizando as funções de predição baseadas nas estatísticas média e mediana. Além do MAPE também é feito o cálculo do NF. Os 5 valores de MAPE e non-fitting reportados são então utilizados para o cálculo da média amostral dos valores

obtidos. A acurácia do STA, para o conjunto de dados, é apresentada em termos de um MAPE médio e são também são calculados os valores dos desvios-padrão para que se possa fazer uma avaliação da estabilidade das medidas obtidas nas diferentes execuções de predições.

Figura 5 – Visão geral do processo de seleção do invólucro com validação cruzada



Fonte: Claudio Aparecido Lira do Amaral, 2018

3.3.5 Testes estatísticos

Uma vez que os resultados foram obtidos com a validação cruzada será necessário avaliar de forma comparativa os resultados obtidos nos experimentos, mais especificamente as informações referentes aos resultados de MAPE médio (calculado a partir dos 5 valores obtidos no processo de validação cruzada). Diante desse cenário, dadas as condições dos experimentos, para realizar essa análise, será utilizada a inferência estatística não-paramétrica (GIBBONS; CHAKRABORTI, 2011), que é válida sob premissas menos restritivas do que as inferências estatísticas clássicas (paramétricas).

O método a ser utilizado para comparação é o teste de Wilcoxon pareado (GRACZYK et al., 2010). Este é um equivalente não paramétrico do teste t de Student pareado. Costuma ser utilizado para testar a diferença na média (ou mediana) de observações emparelhadas - sejam medidas em pares de unidades ou antes e depois de medições na mesma unidade. Ele também pode ser usado como um teste de amostra única para avaliar se uma determinada amostra veio de uma população com uma mediana especificada.

O teste será aplicado sobre os resultados obtidos nos valores de MAPE de cada um dos experimentos. Desta forma, é possível comparar os resultados obtidos em cada um dos conjuntos (folds) de teste e apresentar um parecer a respeito da comparação entre as duas distribuições de probabilidade obtidas.

O teste de hipóteses será realizado utilizando as hipóteses a seguir:

- Hipótese nula: H_0:$\mu_0 = \mu_1$.
- Hipótese alternativa: H_1:$\mu_0 < \mu_1$.

Ou seja, estar-se-á testando se as populações diferem em localização da média ou não, utilizando a seguinte conceito: se houver falha em rejeitar a hipótese nula, não há diferença significativa entre as populações. Já, se a hipótese nula for rejeitada, ou seja, se a mediana da diferença não for nula, as populações diferem em localização. O teste utilizará um nível de significância α de 0.05 no formato unicaudal. Desta forma, se os valores obtidos para o *p-value* no teste forem inferiores, a hipótese nula poderá ser rejeitada.

3.4 Seleção de atributos

O processo de seleção de atributos usado nesse trabalho tem por objetivo tratar algumas das necessidades da mineração de processos no contexto de gerenciamento de incidentes, dentre as citadas na literatura (GUYON; ELISSEEFF, 2003): redução do conjunto de atributos de modo a reduzir os recursos computacionais necessários no processo de predição; e melhoria de desempenho com ganho na acurácia preditiva. A seleção de atributos, à parte da seleção via conhecimento do especialista, é realizada por meio de técnicas do tipo filtro e do tipo invólucro. Nesta seção, é apresentada a forma como as três estratégias foram aplicadas neste trabalho. Os resultados obtidos com a aplicação de todas as estratégias são apresentados e discutidos no capítulo 5.

3.4.1 Seleção por conhecimento do especialista

O propósito da seleção por conhecimento do especialista é utilizar o padrão existente na literatura de mineração de processos, que direciona a construção do STA para o uso do atributo de controle da atividade (no presente caso, o estado do incidente), aliado ao

conhecimento de domínio do processo em estudo, ou seja, as boas práticas e recomendações do framework ITIL (ITSMF, 2013).

Esse contexto foi o ponto de partida ("*baseline*") para a comparação dos resultados obtidos nas proposições de seleção de atributos descritas nas subseções 3.4.2 e 3.4.3. Esta abordagem foi adotada também, pela dificuldade em se obter resultados de referência disponíveis publicamente para serem usados em análise comparativas.

3.4.2 Seleção por filtro

A seleção de atributos por filtro foi aplicada utilizando uma estratégia de *ranking*. Esta abordagem segue conceitos consolidados da literatura especializada (GUYON; ELISSEEFF, 2003; KOHAVI; JOHN, 1997; BLUM; LANGLEY, 1997) e descritos na seção 2.2.1. Nela, uma lista ordenada com todos os atributos é gerada utilizando um critério de relevância.

De forma mais detalhada, o *ranking* foi aplicado com uma etapa de pré-processamento, tal como sugerido por Kohavi e John (1997), para criação de um ponto de partida para a seleção de atributos, independente do modelo de predição escolhido. O critério de relevância foi implementado por meio da análise da variância, usando a correlação das variáveis independentes (i.e., atributos descritivos) e a variável dependente (i.e., o atributo "closed", que é o atributo alvo da predição). Uma vez que a maioria dos atributos descritivos são de natureza categórica, a estatística η^2 (RICHARDSON, 2011) foi selecionada.

A análise de variância deve ser executada em uma amostra representativa do log de eventos enriquecido, ou sobre a sua totalidade. Os atributos e seus respectivos valores de correlação são produzidos, os atributos são ordenados de acordo com esse valor e então um número k de atributos desejado para uso na construção dos STAs é escolhido. A construção dos STAs é realizada a partir do conjunto de atributos, considerando os parâmetros de abstração e horizontes máximos já discutidos, seguindo a ordem indicada pelo *ranking*, iniciando no primeiro atributo. Na sequência, o primeiro atributo é combinado ao segundo para construção do segundo STA, e assim sucessivamente (em um modelo de seleção incremental, do inglês *forward selection*) até que existam k conjuntos de atributos (e k STAs). Na construção dos STAs, deve ser definido um subconjunto de registros de log a ser utilizado.

Enfim, as medidas de qualidade do sistema (NF) e de predição (MAPE) avaliam os STAs gerados. O conjunto de atributos que gerou o sistema de maior qualidade é a resposta do seletor de atributos. Esse conjunto pode, então, ser usado para a geração do STA final, com um número maior de registros de log, para ser usado como o sistema de predição final. A expectativa de que apesar da simplicidade e independência desse método quanto ao modelo de predição, ele apresente um desempenho melhor que a seleção feita por especialistas.

3.4.3 Seleção por invólucro

A seleção de atributos por invólucro, neste trabalho, é realizada por meio de um processo de busca por um subconjunto ótimo ou sub-ótimo de atributos, fazendo uso da acurácia do algoritmo de STA como parte da função de avaliação da busca. O espaço de busca para a seleção por invólucro é composto por todas as combinações possíveis dos k atributos pré-selecionados pelo procedimento de *ranking*, também usado na seleção de atributos por filtro. Uma vez que cada combinação representa um estado em tal espaço, este possui um total de 2^k estados possíveis.

A busca foi implementada sob três estratégias, sendo duas no modelo de busca gulosa – subida da encosta e busca pela primeira melhora (KOHAVI; JOHN, 1997; RUSSELL; NORVIG, 2009) – e uma baseada em computação evolutiva - algoritmo genético (HOLLAND, 1975; MICHALEWICZ, 1996; KOZA, 1996). Nos dois primeiros casos, a opção escolhida para construção dos conjuntos de atributos foi a seleção incremental, como descrito na seleção por filtro [5].

Método subida de encosta

O algoritmo 1[6], é a implementação genérica realizada para os métodos heurísticos. Esta implementação segue a descrição apresentada no trabalho de referência de Kohavi e

[5] A alternativa utilizando o modelo de remoção seletiva (do inglês *backward elimination*) foi preterido porque o número de atributos é elevado e, portanto, o número de avaliações de estados para conclusão do processo é significativamente maior.

[6] A implementação para essa busca foi feita em linguagem R, com execução do processo de expansão utilizando o pacote de processamento paralelo foreach. A execução foi realizada em máquinas com 16 processadores virtuais Intel® Xeon 8168 e 32 GB de mémória RAM e outra máquina com 64 processadores virtuais Intel® Xeon E5 v3 e 128 GB de mémoria RAM.

John (1997), com algumas adaptações para utilização do modelo de predição STA. No caso do processo de busca de subida de encosta, utiliza-se o parâmetro do número máximo de expansões sem melhoria de performance em $j = 1$ e os pontos principais são:

- o conjunto de referência para construção do espaço de estados é composto pelos atributos selecionados na etapa de filtro, portanto, cada estado é um subconjunto desses atributos;
- a expansão do estado significa, tomar o conjunto de campos do estado atual, gerar um subconjunto a partir da subtração do conjunto completo (todos os atributos) desse conjunto [7] e então criar os novos estados executando a combinação do subconjunto de estados atual com o acréscimo de cada um dos campos que compõem o subconjunto resultante da subtração. A seguir é apresentado um exemplo com atributos da entidade *incident*:
  - Conjunto de campos – $[incident\_state, active, category]$
  - Estado inicial – $\emptyset$
  - Primeira expansão – $[\{incident\_state\}, \{active\}, \{category\}]$
  - Segunda expansão, considerando o melhor estado avaliado sendo {incident_state} – $[\{incident\_state\}, \{active\}, \{category\}, \{incident\_state, active\}, \{incident\_state, category\}]$;
- para cada um dos novos estados criados são construidos os STAs;
- a avaliação da acurácia é realizada em cada um dos novos estados gerados.

O processo de busca segue até que seja identificado que uma expansão do estado com melhor avaliação não gere um novo estado com uma acurácia melhor. Nesse momento o processo é interrompido e deve ser realizada a etapa seguinte do método que é a geração do STA com o subconjunto de atributos selecionado e a amostra do log de eventos enriquecido com um número maior de registros (se possível, todos os registros).

3.4.4 Busca pela primeira melhora

Tal como no método de subida de encosta, o algoritmo 1 é a implementação para a busca pela primeira melhora com algumas adaptações para utilização do STA.

A principal diferença com relação ao método de subida da encosta é a utilização da expansão a partir de um estado que não é necessariamente o de melhor avaliação.

[7] Observe que o subconjunto resultante começa com **k** elementos, depois **k-1** e assim sucessivamente

Algoritmo 1 Invólucro com Subida da encosta e Primeira melhora

Entrada: $search\_fields, j, folds, log\_eventos$
Saída: $best\_state, open\_list, closed\_list$

$open\_list, closed\_list, expanded\_list \leftarrow \emptyset$ ▷ Lista vazia
$initial\_state \leftarrow \emptyset$ ▷ Estado inicial começa com vazio
$best\_state \leftarrow initial\_state$
$k\_exp\_count \leftarrow 0$
$MAX\_K\_EXP\_COUNT \leftarrow j$ ▷ Valor 1 para Subida da encosta
$put\_state(open\_list, initial\_state)$ ▷ Insere estado inicial na lista de abertos
while $(k\_exp\_count \leq MAX\_K\_EXP\_COUNT) \wedge (len(open\_list) > 0)$ **do** ▷ Pesquisa enquanto há estados para avaliação e não atingiu o número máximo de expansões sem melhoria
    $v \leftarrow get\_arg\_min(open\_list)$ ▷ Seleciona e remove o estado que minimiza o erro
    $put\_state(closed\_list, v)$ ▷ Armazena na lista de estados explorados
    if $eval\_state(v) < eval\_state(best\_state)$ **then** ▷ Compara o melhor estado obtido com o atual
        $best\_state \leftarrow v$
        $k\_exp\_count \leftarrow 0$ ▷ Reinicia contador de expansões
    $k\_exp\_count \leftarrow k\_exp\_count + 1$
    if $(k\_exp\_count \leq MAX\_K\_EXP\_COUNT)$ **then**
        $expanded\_list \leftarrow expand\_state(v, search\_fields, folds, log\_eventos)$ ▷ Expande o estado, cria o modelo de predição calcula o valor de avaliação dos estados gerados
        for $expanded\_state \in expanded\_list$ **do**
            $put\_state(open\_list, expanded\_state)$

Dessa forma, ao não haver incremento da acurácia, o algoritmo tenta fazer a expansão do próximo estado com melhor acurácia e que ainda não foi totalmente explorado[8]. Esse processo continua até que seja encontrado um estado com melhor avaliação, atingido um número máximo **j** de expansões sem que haja modificação do melhor estado.

Este processo é mais robusto que o anterior, pois permite que o espaço de estados seja explorado de maneira mais ampla. Porém, as observações e resultados apontados na literatura não o relacionam necessariamente com uma melhor acurácia, apresentando, por vezes resultados idênticos à subida da encosta (QUINLAN; CAMERON-JONES, 1997; DOMINGOS, 1999; JENSEN; COHEN, 2000). Considerando que o número de expansões será maior, esse método necessita de mais tempo (ou capacidade de processamento) para execução quando comparado ao subida de encosta.

3.4.5 Algoritmo genético

Os métodos de busca heurística, listados nas seções 3.4.2 e 3.4.3, foram utilizados como forma de explorar o espaço de buscas de maneira estruturada, seguindo um formato incremental direcionado pelo resultado da acurácia obtida por cada conjunto de atributos. Estes modelos frequentemente apresentam resultados satisfatórios relacionados à otimização,

[8] Entenda-se a colocação *totalmente explorado* como a expansão de todos os estados vizinhos.

porém, limitam-se a um espaço guiado pela heurística e não exploram hipóteses alternativas à esse modelo. Com este cenário, faz-se necessário buscar um modelo alternativo para estender o processo de avaliação de outras combinações de atributos. Propõe-se portanto, a utilização dos algoritmos genéticos para realizar esta exploração e avaliar sua aplicação ao problema proposto. Esta proposição de utilização é feita de maneira genérica e não restrita apenas ao cenário de incidentes e seu respectivo número de atributos.

A seleção de atributos usando algoritmo genético como função de avaliação do invólucro segue o mesmo princípio utilizado no método de subida da encosta, porém, com o diferencial no qual o espaço de estados é gerado de forma aleatória. Dessa maneira, para execução, o algoritmo 2 precisa da lista de atributos selecionada na etapa de filtro(searh_fields), o número de estados (m), que será o tamanho da população, o número máximo de gerações (g) e as respectivas probabilidades de reprodução(pr), cruzamento(pc) e mutação(pm).

O cromossomo que representa cada indivíduo da população é definido por:

$$cromossomo = \{< attr_1, attr_2, \ldots, attr_n >, < horizonte >\}$$

com alfabeto binário para os genes que representam os atributos e, para o gene que representa o horizonte, o alfabeto contém todas as opções de tamanhos de horizonte máximo desejados pelo projetista do algoritmo. A presença (alelo 1) ou ausência (alelo 0) de cada um dos atributos é definida inicialmente de forma aleatória e posteriormente pelos operadores genéticos. Embora o cromossomo permita a variação no horizonte de forma ampla (de 1 até o valor de tamanho do maior traço no log), nos testes deste trabalho, foi utilizado o mesmo conjunto definido na seção 3.3.2. Esta definição foi realizada de maneira a permitir uma comparação mais direta com as demais abordagens. A utilização das abstrações também seguiu a mesma abordagem de padronização na utilização.

Como forma de melhoria do processo de busca, algumas estratégias de implementação foram usadas, otimizando o uso dos recursos computacionais. São elas:

- criação de uma lista de estados explorados que armazena todos os cromossomos que foram avaliados e evita que sejam geradas novas avaliações para esses estados já visitados;

- no processo de cruzamento e mutação, a lista de estados explorados e a nova população é consultada para evitar a ocorrência de estados "gêmeos" e consequentemente a convergência prematura para um grupo de estados.

Algoritmo 2 Invólucro com Algoritmo genético

Entrada: $m, g, pr, pc, pm, search\_fields, horizonte, folds, log\_eventos$
Saída: $best\_state, closed\_list$

1: $closed\_list \leftarrow \emptyset$ ▷ Lista vazia
2: $best\_state \leftarrow \emptyset$ ▷ Estado inicial começa com vazio
3: $geracao \leftarrow 1$
4: $pop\_list \leftarrow gera\_pop(m, search\_fields, horizonte)$ ▷ População inicial
5: **while** $geracao \leq g$ **do** ▷ Executa até o número máximo de gerações
6: $evaluated\_list \leftarrow gen\_eval\_list(pop\_list, folds, log\_eventos)$ ▷ Gera modelos e resultados do MAPE
7: $v \leftarrow get\_arg\_min(evaluated\_list)$ ▷ Seleciona o estado que tem o menor erro na lista
8: $put\_state(closed\_list, evaluated\_list)$ ▷ Armazena na lista de estados explorados
9: **if** $eval\_state(v) < eval\_state(best\_state)$ **then** ▷ Compara o melhor estado obtido com o atual
10: $best\_state \leftarrow v$
11: $sorted\_list \leftarrow sort(evaluated\_list, pop\_list)$ ▷ Gera lista ordenada para aplicação operadores genéticos
12: $pop\_list \leftarrow \emptyset$
13: **for** $i \leftarrow 1$ a m **do** ▷ Gera a nova população
14: $operacao \leftarrow seleciona\_oper(pr, pc, pm)$ ▷ Seleciona operador genético baseado nas probabilidades recebidas
15: **if** $operacao = R$ **then** ▷ Reprodução
16: $new\_state \leftarrow seleciona(sorted\_list)$ ▷ Seleciona com base no método de roleta
17: **else**
18: **if** $operacao = C$ **then** ▷ Cruzamento uniforme
19: $new\_state \leftarrow crossover(seleciona(sorted\_list), seleciona(sorted\_list), closed\_list)$
20: **else** ▷ Mutação simples
21: $new\_state \leftarrow mutacao(seleciona(sorted\_list), closed\_list)$
22: $put\_state(pop\_list, new\_state)$
23: $geracao \leftarrow geracao + 1$

No algoritmo genético, a função de avaliação Fitness é dada por:

$$Fitness(i) = build\_eval\_state(pop\_list(i), a, fn)))$$

com

$$a = \{conjunto, multiconjunto, sequencia\}, fn = \{media, mediana\}$$

.

Em termos de operadores genéticos, foram usados: seleção por roleta, cruzamento uniforme e mutação simples, sendo que a função *build_eval_state* recebe o cromossomo do o indivíduo i (campos e horizonte selecionado) para que sejam gerados os STAs nas representações de abstração a e funções de predição fn. Esta função faz a avaliação do

$MAPE$ para cada um dos modelos STA correspondentes e retorna o menor valor obtido que será o valor da função de avaliação $Fitness(i)$ para o indivíduo i.

Outro parâmetro a ser definido na utilização desse modelo é o número de instâncias do processo de incidentes(log_eventos) que serão utilizadas no algoritmo de buscas. Esse número influenciará diretamente o tempo de execução e acurácia do resultado final, ou seja, um valor pequeno permite uma exploração maior do número de estados, porém, pode ter como efeito colateral a não exploração por completo dos estados necessários do STA para que possa ser representativo. Ao contrário, um número muito grande pode fazer com que seja necessário a utilização de uma capacidade de processamento significativa para obter o resultado em um tempo aceitável para a demanda gerada.

Após concluído o processo de busca, o subconjunto resultante, tal como nos métodos anteriores, deve seguir o processo de geração do STA com o subconjunto de atributos selecionado e o log enriquecido com um número maior de registros.

4 Experimentos exploratórios

De posse dos registros organizados no log de eventos enriquecido, foi possível realizar alguns experimentos exploratórios, referentes à mineração de processos no contexto sob estudo neste trabalho. Esses experimentos tiveram o objetivo de:

- avaliar o comportamento do processo de gerenciamento de incidentes, de forma a identificar se o processo em execução na organização é um processo estruturado (do tipo *lasanha*), semi-estruturado ou não-estruturado (do tipo *espaguete*). O conhecimento sobre o enquadramento do processo é necessário para que possa ser avaliada a possibilidade de construção de preditores, tal como descrito por Aalst (2011). Para essa tarefa foi utilizada a ferramenta DISCO (FLUXICON, 2018);
- identificar o comportamento do processo de incidentes com relação às estimativas realizadas a partir do uso de sistemas de transições e da criação de STAs. Esse experimento foi realizado utilizando a ferramenta ProM (VERBEEK et al., 2011) e teve como objetivo validar a proposição de acréscimo de atributos para construção de STAs mais eficazes.

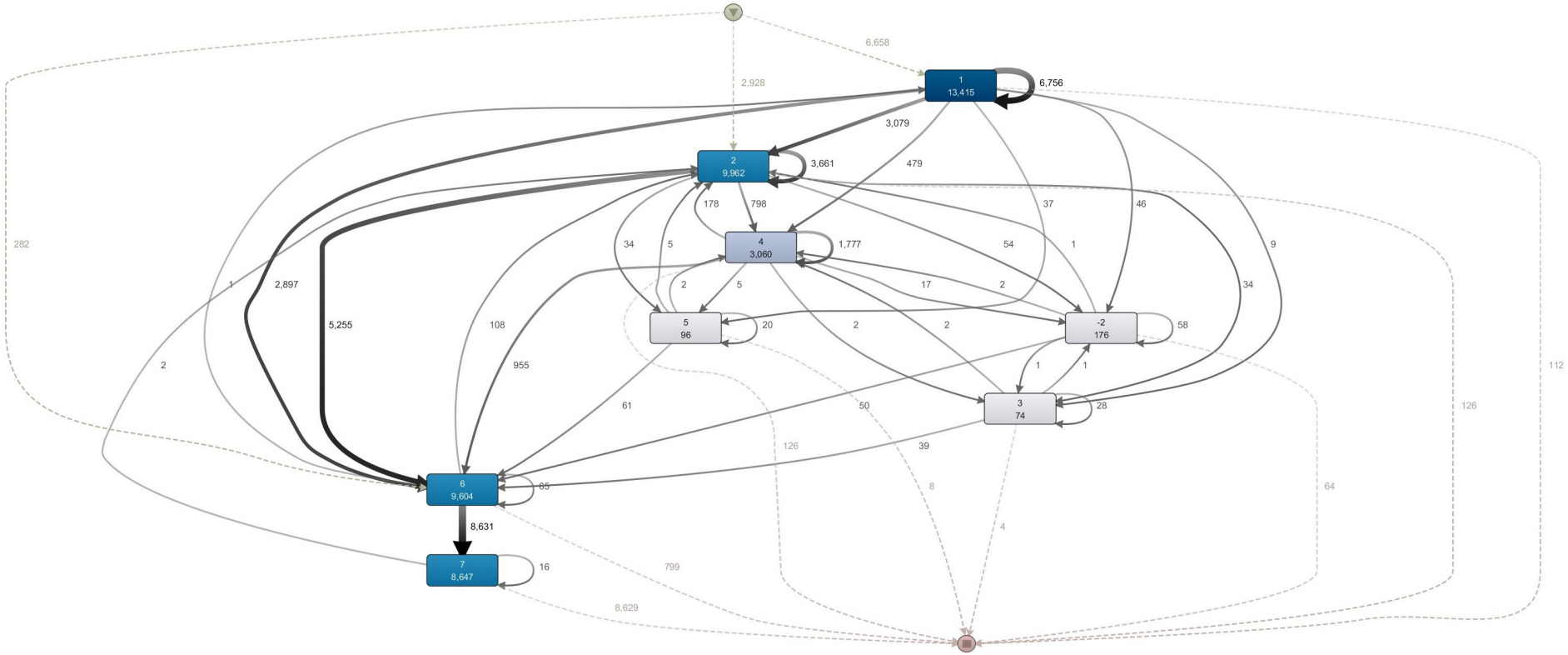
Este capítulo é dedicado a relatar os resultados obtidos nessa exploração e também discutir as dificuldades encontradas durante a realização da atividade.

4.1 Mineração de processos com a Disco - descoberta de modelo de processo

A ferramenta Disco (FLUXICON, 2018), da empresa Fluxicon Inc., em sua versão acadêmica, foi escolhida para execução de um experimento contextualizado no tipo de mineração de processos "descoberta". A Disco é uma ferramenta amplamente usada para experimentos acadêmicos por disponibilizar várias formas de visualização e análise de resultados para tarefas de descoberta de processos.

O log de eventos enriquecido, produzido a partir do procedimento descrito na seção 3.2.4, foi usado na ferramenta DISCO. Na figura 6 é apresentado o modelo do processo descoberto pela ferramenta. Nessa visualização, as atividades do modelo (retângulos) dizem respeito ao atributo de log *incident_state*, e as ligações entre as atividades dizem respeito às transições entre os estados do incidente. A espessura das linhas que constituem as ligações são relacionadas à frequência (absoluta) daquela transição no log de eventos enriquecido.

Figura 6 – Modelo de processo gerado a partir do log de eventos enriquecido usando a ferramenta DISCO. Visualização completa com atividades e frequência (absoluta) de transições



Fonte: Claudio Aparecido Lira do Amaral, 2018

Nessa visualização todos os estados da atividade bem como todas as transições estão representadas. As linhas pontilhadas que chegam no estado representado por um círculo com um quadrado no centro (na base da figura) indicam os casos ainda ativos, ou seja, que não haviam ainda sido concluídos no momento de coleta do log de eventos enriquecido.

Analisando o modelo descoberto, observa-se que trata-se de um processo semi-estruturado (seção 2.1), pois apesar das características de processo estruturado (mais de 80% das situações possuem um padrão), algumas atividades requerem uma interpretação e podem sofrer desvios, dependendo das informações ou características do caso.

Além disso, em uma análise preliminar, pode-se dizer que o processo descoberto possui um bom nível de conformidade com o processo previsto *a priori* para o gerenciamento de incidentes. Como esperado pelos conhecedores do modelo de processo conceitual para o gerenciamento de incidentes: a maior frequência de transições ocorre na sequência de estados - 1 (Novo) 2 (Ativo) 6 (Resolvido); a transição para o estado 7 (Encerrado) é realizada de forma automática pelo sistema após cinco dias no estado 6 (Resolvido); algumas transições são realizadas do estado 1 (Novo) para o 6 (Resolvido) diretamente.

A permanência dos processos, com maior frequência, nos estados 1 (Novo), 2 (Ativo) e 6 (Resolvido) pode ser observada na tabela 6, a qual lista a frequência relativa de passagem dos processos por todos os estados possíveis, extraída diretamente do log de eventos enriquecido. A soma das frequências dos três estados citados alcança mais de 70%. Este cenário permite verificar que a dinâmica do processo de incidentes se mantém centrada no fluxo: abertura, atuação com objetivo de resolução e encerramento.

Tabela 6 – Frequência dos estados nas instâncias de processos de gerenciamento de incidentes

Atividade	**Frequência**	**Frequência Relativa**
1	13,415	29.79 %
2	9,962	22.12 %
6	9,604	21.33 %
7	8,647	19.20 %
4	3,060	6.79 %
-2	176	0.39 %
5	96	0.21 %
3	74	0.16 %

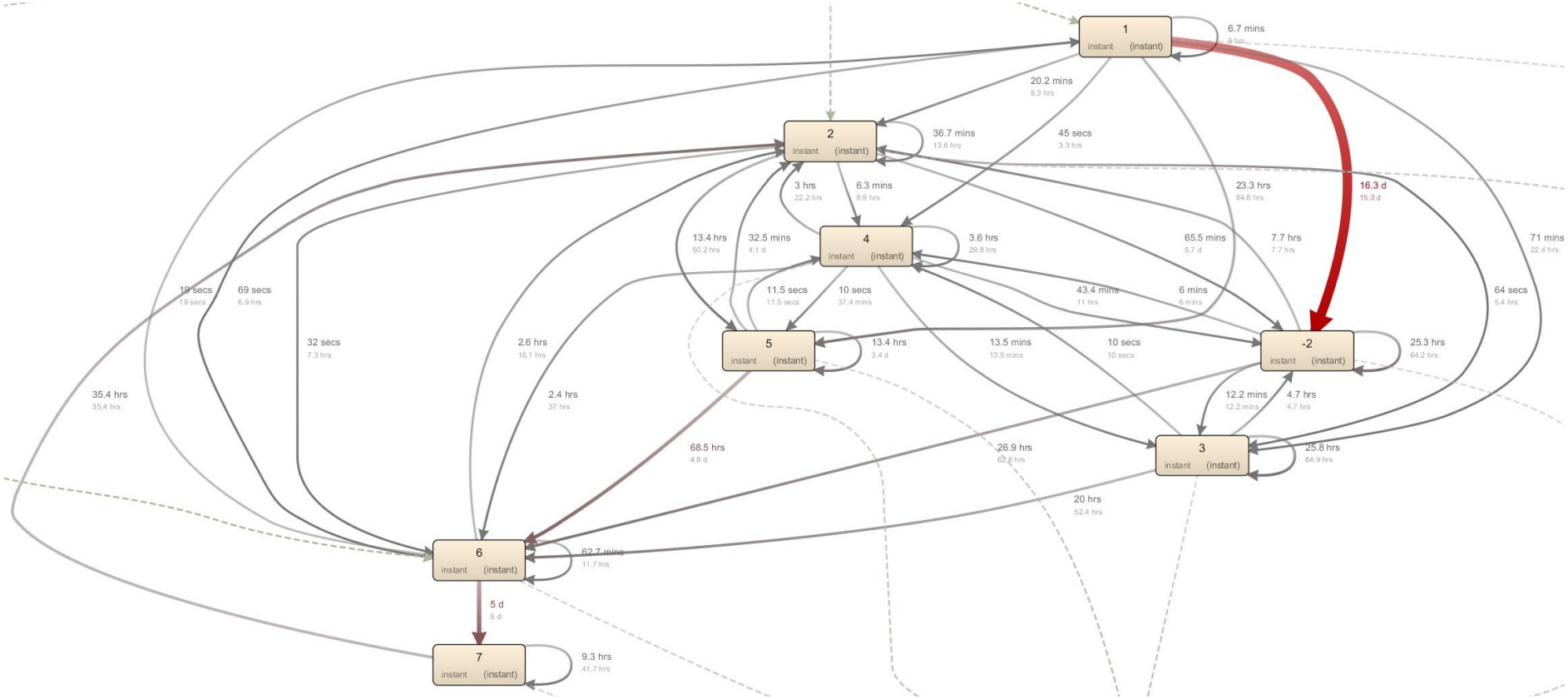
Fonte: Claudio Aparecido Lira do Amaral, 2018

Uma análise complementar foi realizada para avaliação dos tempos de permanência em cada atividade e dos tempos necessários para a ocorrência de cada uma das transições

entre as atividades. Esta é uma análise de interesse para esse projeto, visto que a motivação do projeto está relacionada com a tarefa de estimativa de tempo de execução, ou tempo restante para término, de uma instância de processo no gerenciamento de incidentes.

A figura 7 apresenta os tempos de transição no modelo de processo. A informação em destaque em cada uma das ligações na visualização é a mediana dos tempos daquela transição, a partir do que foi observado no log de eventos enriquecido. A informação com menos destaque é o tempo médio. Observa-se que, para grande parte dos casos, esses valores são bem diferentes, indicando que a distribuição dos tempos de transição é assimétrica, inserindo algum nível de complexidade em uma análise de estimativa de tempo. Por exemplo, a transição 2 para 6 tem uma mediana de 32 segundos e uma média de 7,3 horas, ou seja, a maioria das transições entre esses estados é rápida, mas há situações que talvez representem exceções (casos críticos). Disparidades como essas são esperadas, pois o processo de incidentes trata situações distintas que variam de uma simples troca de senha até a reconstrução por completo de um banco de dados corrompido que demanda várias ações complexas. A exceção nesse modelo de processo é a transição do estado 1 (Novo) para -2 (Aguardando fornecedor) que tem mediana de 16,3 dias e media de 15,3 dias.

Figura 7 – Modelo processos gerado a partir do log de incidentes por meio da ferramenta DISCO. Visualização completa com atividades e tempo (mediano e médio) de transições



Fonte: Claudio Aparecido Lira do Amaral, 2018

4.1.1 Mineração de processos com a ProM - Sistema de transição de estados anotado

A ProM (*Process Mining Framework*) (VERBEEK et al., 2011) é uma ferramenta de código aberto, desenvolvida a partir de colaborações entre profissionais da academia e da indústria, que se dedicam a estudar a área de mineração de processos e propor soluções para resolução dos problemas de descoberta, conformidade e melhoria de processo. Essa ferramenta é bastante usada tanto como um *framework* para disponibilização e teste de algoritmos que constituem o estado da arte na área, quanto para o estabelecimento do estado da prática da área.

O STA, desenvolvido por Aalst, Schonenberg e Songa (2011) e descrito no capítulo 2 é a base inicial para o desenvolvimento do projeto aqui apresentado. Para a construção desse sistema, a partir do log de eventos enriquecido, foi utilizado um *plugin* disponibilizado na ferramenta ProM[1] chamado "TS Miner" [2].

A ferramenta ProM trabalha com um formato de log padrão chamado XES - um padrão baseado em XML para logs de eventos (AALST; SCHONENBERG; SONGA, 2011). Sua finalidade é garantir interoperabilidade entre os diferentes *plugins* construídos para a ferramenta, e facilitar o desenvolvimento de funcionalidades para mineração de dados e análises estatísticas. A ferramenta ProM oferece funcionalidades de conversão de arquivos de forma a construir o arquivo XES. A conversão dos registros do log de eventos enriquecido para o formato XES permitiu a extração de algumas estatísticas simples que permitiram verificar a corretude da conversão ao compará-las com as estatísticas obtidas com a DISCO (4.1): são 9.868 incidentes (instâncias de processos), e 45.034 eventos (registros de log).

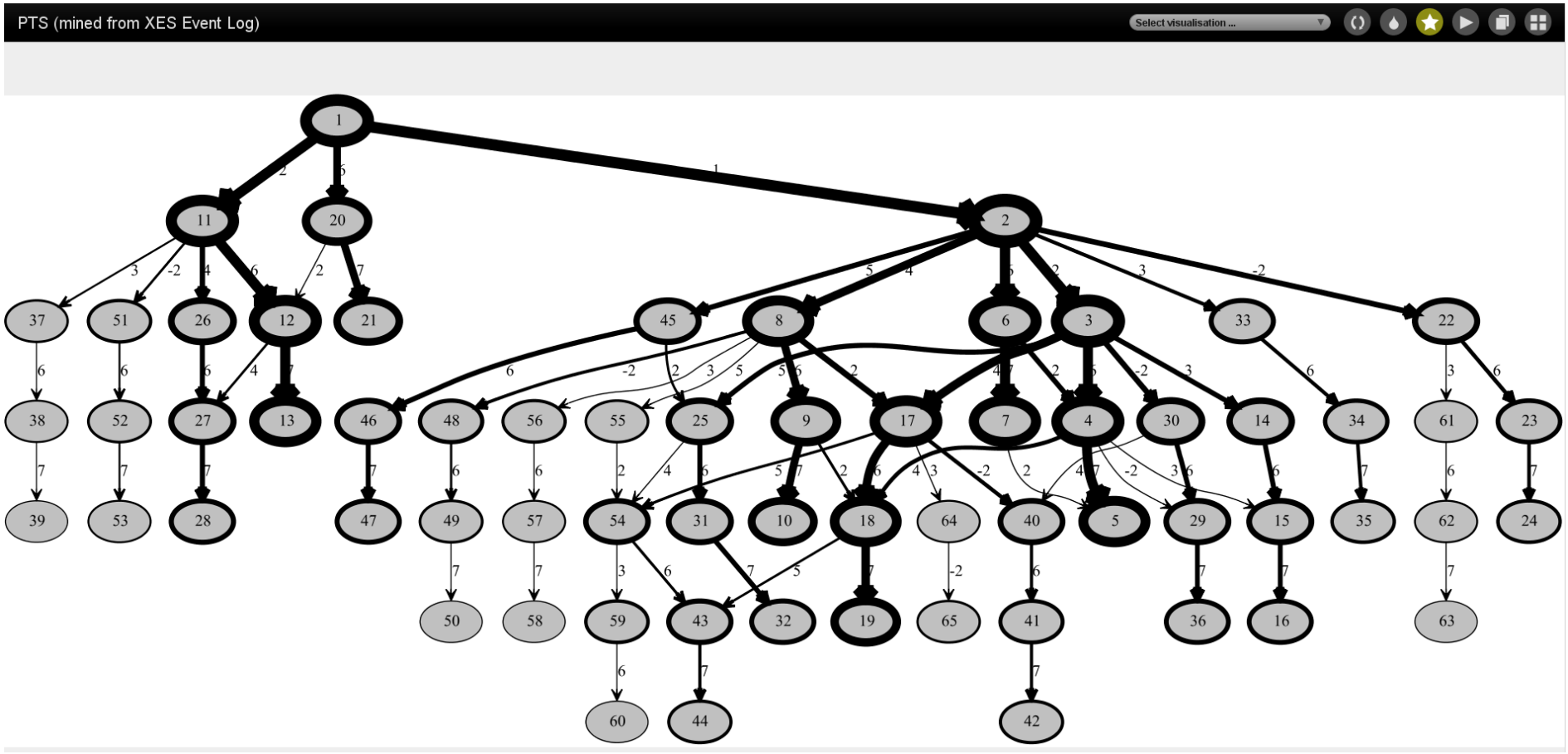
A partir da disponibilidade do log no formato XES foi possível seguir com a etapa de criação de sistemas de transição de estados. O primeiro modelo foi obtido a partir do uso do atributo *incident_state* como indicador da atividade em observação no sistema de transições. Essa escolha indica a "chave" usada pela ferramenta para representação dos eventos. Na figura 8 é mostrado o sistema de transição de estados resultante, com todas as transições observadas no log de eventos enriquecido, com exceção daquelas que não alteram o estado (*status*) do processo (que seriam representados por arcos que saem e chegam no mesmo nó do modelo). Foram identificados 63 estados distintos.

---

[1] A versão utilizada é a 6.6 (64 bits) revisão 28643 para plataforma Windows.

[2] O pacote TSMiner está disponível em http://www.promtools.org/.

Figura 8 – Sistema de transição de estados com atributo *incident_state* usado como chave, gerado com o plugin "TS Miner" / ProM



Fonte: Claudio Aparecido Lira do Amaral, 2018

Analisando o modelo, nota-se que as sequências de transições resultantes (1-6-7; 2-6-7; 1-2-6-7) são mais frequentes (arcos de maior espessura na figura). Trata-se de um cenário esperado considerando que o objetivo do processo de gerenciamento de incidentes é o reestabelecimento de serviços. Nas três sequências mais frequentes, o estado 6 (Resolvido) está presente. Essa observação corrobora com o que foi identificado a partir da ferramenta Disco (Seção 4.1), porém, no presente caso, a forma de estruturar a visualização é diferente e permite algumas análises mais específicas (mais sequências frequentes podem ser observadas a partir de uma inspeção visual do resultado).

Outra análise relevante que pode ser feita a partir deste sistema de transição de estados é referente ao número de variantes existentes no processo em relação às sequências de transições entre os estados. Esse sistema de transição de estados foi gerado a partir de um contexto no qual havia oito possibilidades diferentes para a variável *incident_state*, ou seja, são oito classes de eventos. Porém, a forma como o processo transita entre essas possibilidades gerou 63 possibilidades diferentes para as instâncias do processo. Considerando que há outros atributos que caracterizam as instâncias dos processos, esse teste fornece um indicativo de que, realmente, a seleção de atributos é um ponto importante para obtenção de um modelo refinado de predição de tempo restante de execução, para uma instância de processo.

A fim de conhecer melhor o problema sobre tratamento neste projeto, foi realizado uma análise a partir da geração do sistema de transição de estados com a utilização dos atributos *incidente_state* e *categoria* como chave. O objetivo foi avaliar o comportamento das transições com a inclusão de um atributo adicional na chave do sistema. A escolha do teste com o atributo *categoria* está baseada no conhecimento do especialista: no processo de gerenciamento, os incidentes são direcionados a grupos solucionadores de acordo com alguns critérios técnicos, e a categoria do incidente é um deles. Categorias podem ser entendidas como um agrupador de primeiro nível para os incidentes: “desktop”, “pacote office”, “rede”, “SAP CRM”, “SAP ECC”, “SAP GRC”, “SAP”, “senhas e acessos”, “software industrial” e “telefonia”.

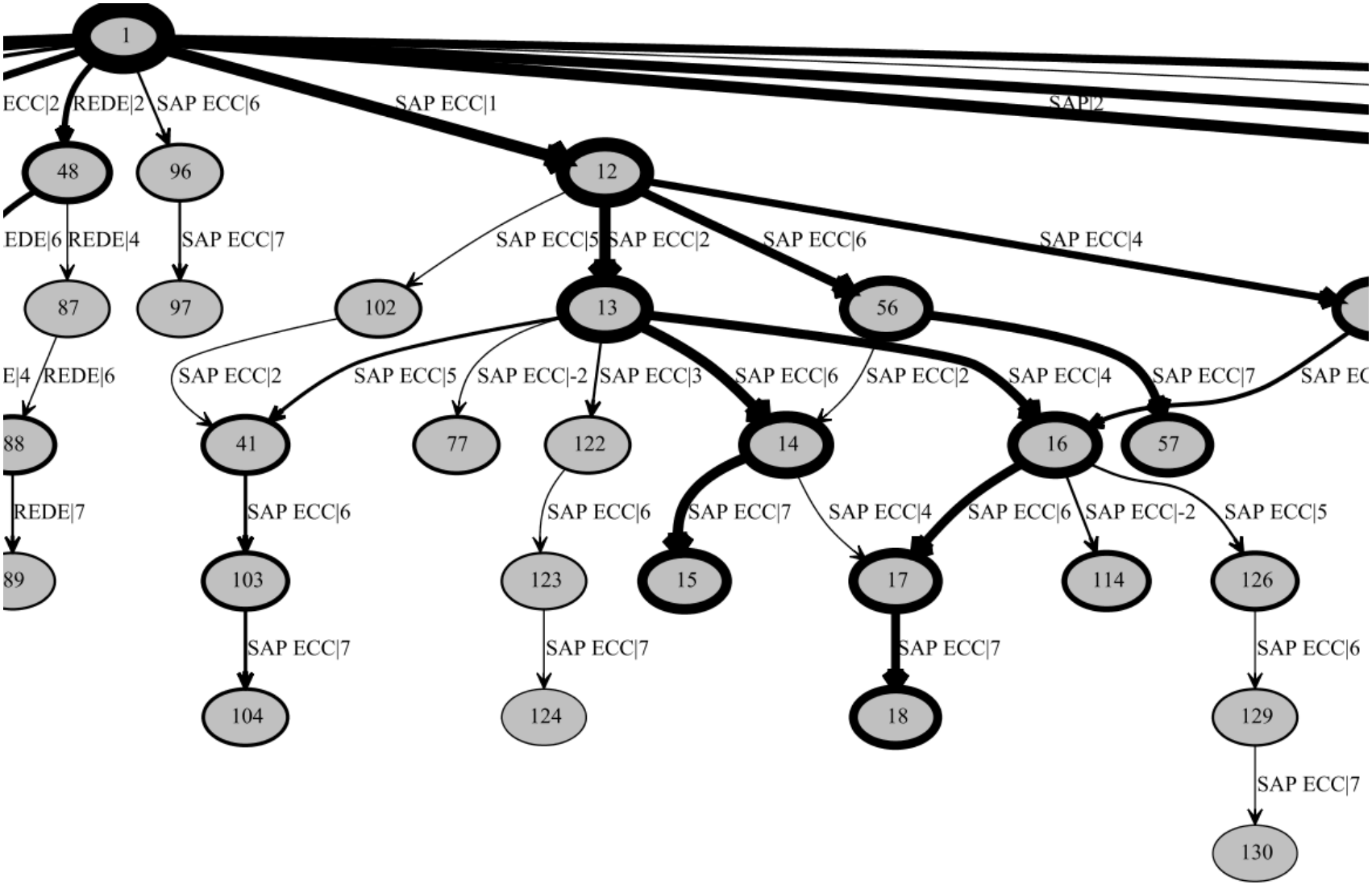
O log de eventos enriquecido foi novamente transformado para o padrão XES agora considerando a chave composta por dois atributos descritores do processo. Para esse caso, foram obtidas 269 classes distintas de eventos presentes no log, dentro de um total possível de 680 classes. Não foi possível gerar o sistema de transições para todo o conjunto de registros de log neste caso (limitações da ferramenta ProM, que apresentou um erro

durante o processamento) e, por isso, foram selecionados 28.754 eventos, escolhidos de forma empírica, totalizando 62,85% do total de registros disponíveis. O modelo resultante apresenta um total de 130 estados distintos. Na figura 9 é apresentado um recorte parcial com a representação do modelo gerado.

Concluída a etapa de construção dos sistemas de transição de estados, o passo seguinte foi a geração do modelo de estimativas baseado no conceito de sistema de transição de estados anotado, o STA detalhado na seção 2.1.3. Para geração do modelo, o *plugin* "TransitionSystems"[3], disponível na plataforma ProM, foi utilizado. Esse *plugin* usa as informações geradas na criação do sistema de transição de estados realizado pelo *plugin* "TS Miner". A visualização disponibilizada pelo *plugin* "TransitionSystems" permite observar o sistema de transição de estados acompanhado de informações referentes a cada estado. São elas: tempo restante para finalização da instância do processo a partir daquele estado, tempo gasto na instância do processo até alcançar aquele estado, tempo de permanência em um estado para aquela instância de processo, e estatísticas (média, desvio padrão, frequência e tempos máximo e mínimo) referentes a essas informações. Na figura 10 é apresentado o modelo gerado a partir da escolha do atributo incident_status como campo chave (o modelo de entrada para esse passo é aquele apresentado na figura 7. A escala de cores utilizada na figura fornece uma noção sobre os tempos máximos relacionados a cada estado da instância de processo.

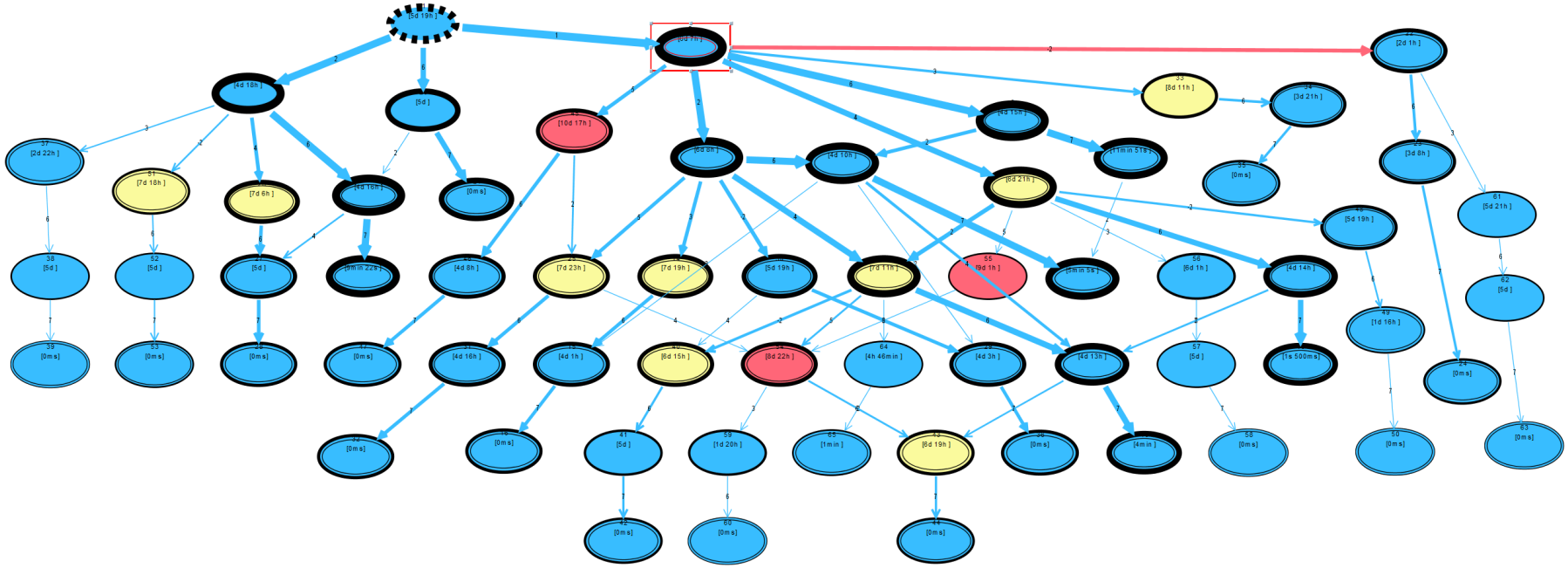
[3] O pacote TransitionSystems está disponível em http://www.promtools.org/.

Figura 9 – Recorte de modelo do sistema de transições com os atributos *incident_state* e *category* usados como chave, gerado com o *plugin* "TS Miner" / ProM



Fonte: Claudio Aparecido Lira do Amaral, 2018

Figura 10 – Modelo STA, com atributo *incident_state* usado como chave, gerado com o *plugin* “TransitionSystems” / ProM



Fonte: Claudio Aparecido Lira do Amaral, 2018

Para realizar uma avaliação referente a estimativa de tempo que o STA pode oferecer, foi avaliada a seguinte sequência de transição de estados do processo de gerenciamento de incidentes (1-2-6-7), que totaliza em 1.945 instâncias de processos. A ideia foi usar as informações do STA como um estimador de tempo, como proposto por Aalst, Schonenberg e Songa (2011).

A tabela 7 apresenta as informações sobre os tempos em cada um dos estados do grupo de instâncias de processos referente à sequência de transições 1-2-6-7. Para utilização como previsão de conclusão, observa-se que há uma incerteza alta ao realizar a estimativa a partir do estado 1. Neste caso, a previsão de permanência no estado é de 9 horas e 24 minutos, com um desvio padrão de quase dois dias (1 dia e 22 horas), ou seja, aproximadamente cinco vezes o valor estimado. O tempo restante é em média de 6 dias e 7 horas, com um desvio padrão de 4 dias e meio. Na sequência, no próximo estado (estado 2), há o tempo gasto de 14:45 horas e expectativa de conclusão em 6 dias e 8 horas em média. A permanência apresenta um resultado com desvio padrão três vezes o valor da média. Ao realizar a transição para o estado 6, o tempo gasto chega em média a 1,75 dias e o desvio padrão próximo de 4 dias, menor que os anteriores, porém, duas vezes maior que a média. Essas predições não são adequadas considerando o ponto de vista do negócio.

Tabela 7 – Estimativas via STA usando o atributo chave *incident_status*. O cenário é a sequência 1-2-6-7

Transição (1-2-6-7)					
Status	**Indicador tempo**	**Média**	**Desvio Padrão**	**Máximo**	**Frequência**
1	Gasto	0 ms	0 ms	0 ms	6658
	Restante	6d 7h	4d 12h	41d 5h	6658
	Permanência	9h 24min	1d 22h	38d 7h	6658
2	Gasto	14h 45min	1d 23h	34d 22h	3079
	Restante	6d 8h	4d 14h	41d 5h	3079
	Permanência	19h 24min	2d 13h	41d 1h	3079
6	Gasto	1d 18h	3d 19h	41d 5h	2213
	Restante	4d 10h	1d 19h	23d 4h	2213
	Permanência	4d 6h	1d 18h	5d 21h	2213
7	Gasto	6d 11h	3d 1h	40d 1h	1945

Fonte: Claudio Aparecido Lira do Amaral, 2018

Com base nos valores identificados, foi possível concluir que a utilização do STA apoiado apenas na utilização do campo de estado do incidente não é suficiente para produzir previsões com uma precisão relevante.

Já usando um STA construído com uma chave composta - *incident_state* e *categoria*, resultados diferentes são obtidos. O experimento foi realizado com um STA criado a partir da escolha de cinco categorias (REDE, SAP CRM, SAP ECC, SAP GRC e SAP). A

justificativa para a escolha da quantidade reduzida de valores para o atributo *categoria* é garantir a comparação da mesma sequência de transições (1-2-6-7) usada na análise anterior. Essas categorias somadas representam 494 eventos, ou seja, 25,40% do total de eventos identificados nessa sequência.

Tabela 8 – Análise via sistema de transição de estados anotado usando os atributos chave *incident_state, category*. O cenário é a sequência 1-2-6-7 e a variável em análise é o "Tempo Gasto"

Categoria	**Média**	**Desvio Padrão**	**Mínimo**	**Máximo**	**Frequência**
REDE	6d 03h	2d 08h	5d 00h	20d 19h	99
SAP CRM	6d 14h	3d 17h	5d	23d 22h	27
SAP ECC	5d 16h	1d 06h	5d	16d 00h	127
SAP GRC	4d 20h	1d 22h	13 min	8d 02h	10
SAP	5d 15h	2d 10h	11 min	27d 21h	231
Completo	6d 11h	3d 01h	11 min	40d 01h	1945

Fonte: Claudio Aparecido Lira do Amaral, 2018

A tabela 8 apresenta as informações obtidas sobre o tempo gasto para execução da sequência de atividades em cada uma das 5 categorias selecionadas. O valor geral obtido anteriormente no modelo completo (apenas com estado) está na ultima linha. Observando a média, percebe-se que a inclusão do atributo *categoria* levou a variações para cada um dos novos grupos de instâncias de processos. A variação é de cerca de 42 horas entre o grupo que tem o menor tempo gasto (SAP GRC) e o grupo que tem o maior tempo gasto (SAP CRM). No entanto, essa variação não parece ser significativa quando considera-se os desvios padrão associados, embora a ordem da diferença entre média e desvio padrão tenha diminuído em relação ao teste anterior.

Esses testes mostram a evidência de que a hipótese deste trabalho pode ser confirmada: a otimização da seleção de atributos a ser considerada para uma estimativa adequada de tempo relacionado à execução do processo se faz necessária, porque:

- aparentemente quanto mais atributos estiverem envolvidos na geração do STA, maior precisão na estimativa de tempo será alcançada (o que já era esperado);
- o uso de muitos atributos para geração do STA gera um ambiente de processamento grande o suficiente para inviabilizar a execução dos procedimentos de geração do modelo;
- considerando os dois itens anteriores, tem-se um problema de otimização com objetivos conflitantes que justifica o estudo de técnicas específicas para tal.

5 Experimentos e resultados

Neste capítulo são apresentados os experimentos executados para validar a abordagem de seleção de atributos estabelecida neste trabalho, bem como os resultados obtidos acompanhados de análises. A fim de organizar a apresentação do conteúdo, o capítulo está dividido em: uma seção na qual a instância de log usada nos experimentos é detalhada (seção 5.1); quatro seções para detalhamento de cada experimento executado e resultados produzidos (seções 5.1.1 - seleção por especialistas, 5.1.2 - seleção por filtro, 5.1.3 e 5.1.4 - seleção por invólucro). O capítulo é finalizado com algumas considerações finais (Seção 5.2) sobre os resultados obtidos.

5.1 Log de eventos enriquecido

Como primeiro passo, para realização dos experimentos, um log de eventos enriquecido relacionado ao processo de gerenciamento de incidentes foi extraído de uma instância da plataforma $ServiceNow^{TM}$, utilizada por uma empresa de tecnologia da informação, de acordo com as definições apresentadas no capítulo 3. Esse log possui informações obtidas a partir do sistema de auditoria e do modelo relacional da plataforma e segue resumidamente descrito aqui:

- **Registros do log de eventos**: Os dados principais relacionados aos registros de atualização dos incidentes são: identificador do evento, valor anterior, valor novo, data e hora da atualização e o usuário responsável pela atualização. Os dados de auditoria foram utilizados para gerar a estrutura do log de eventos a ser minerado. O período considerado foi de 12 meses – março de 2016 a fevereiro de 2017 – totalizando 24.918 traços e 141.712 eventos. Foi necessário realizar uma etapa de pré-processamento para filtrar registros inconsistentes e organizar os registros de auditoria em uma sequência compatível com um formato de log de eventos (ordem crescente de data), conforme explicado na seção 3.2.4. Atributos referentes a "datas de atualização" e "responsável pela atualização" foram derivados diretamente do sistema de auditoria (atributos *sys_updated_on* e *sys_updated_by* descritos no Apêndice B.
- **Atributos descritivos de incidentes**: A $ServiceNow^{TM}$ possui, na implementação utilizada, 91 atributos. Entretanto, alguns deles não puderam ser utilizados para os

propósitos de mineração de processos, porque possuem dados inconsistentes e/ou incompletos ou ainda representam informação não estruturada (i.e., texto), cuja utilização está fora do escopo proposto nesse trabalho. Após o processo de remoção dos atributos desnecessários, o conjunto final de atributos descritivos é composto de 34 atributos (27 categóricos e 7 numéricos).

O log de eventos enriquecido foi utilizado para criação de quatro conjuntos de amostras selecionadas aleatoriamente com 1.000, 8.000, 12.000 e 24.000 instâncias do processo de gerenciamento de incidentes. Os dois conjuntos de 8.000 e 12.000 instâncias são destinados aos procedimentos de avaliação da acurácia durante o processo de seleção de atributos nos métodos de seleção por filtro e por invólucro. O último conjunto, composto por praticamente todas as instâncias disponíveis no conjunto de dados, destinou-se a uma avaliação comparativa dos resultados obtidos pelos métodos utilizados. O conjunto reduzido, de 1.000 instâncias, destina-se ao experimento com algoritmo genético, especialmente por conta do custo computacional demandado por esse tipo de método.

Tabela 9 – Estatísticas log eventos enriquecido: distribuição do número de registros de log por incidente e duração em dias

	1º·Quart.	**2º·Quart.**	**3º·Quart.**	**Máximo**	**Média**	**Desvio Padrão**
Por eventos	3	5	7	58	6	3.67
Por duração	0,01	0,40	5,29	336,21	6,67	21,20

Seguindo a proposição descrita na capitulo 3, foram geradas algumas estatísticas sobre o log de eventos enriquecido, as quais seguem apresentadas na tabela 9. Além de observar o comportamento das informações, o objetivo dessa análise descritiva é produzir embasamento para a escolha dos parâmetros que foram utilizados na etapa de construção dos STAs. Das informações coletadas, pode-se observar um comportamento bem definido para o processo de gerenciamento de incidentes: a maioria dos incidentes (75%), tem até sete registros no log, ou seja, são realizadas até sete atualizações em uma instância de processo, considerando período de tempo compreendido entre o seu início e o seu encerramento; e em média, são necessárias seis atualizações para encerrar um incidente; os valores mais frequentes indicam cinco atualizações. Ao observar o tempo decorrido para a conclusão dos incidentes, nota-se que metade dos casos são concluídos dentro do mesmo dia de abertura, porém, ao observar a duração média de 6,67 dias, o valor é superior ao limite do 3º quartil (5,29 dias), mostrando que existe uma variação significativa nos casos do último quartil – influenciando o valor da média. Esse item pode ser mais bem avaliado

ao observar o comportamento do desvio-padrão que é de 21,20, ou pouco mais que três vezes o valor da média. Essa amplitude de formato confirma as avaliações iniciais, do capítulo 4, de que o processo tem um comportamento bem definido para a maioria dos casos, mas possui uma grande variação em um conjunto mais reduzido, evidenciando o modelo semi-estruturado apresentado no capítulo 2.

5.1.1 Experimento #1 – Seleção pelo conhecimento do especialista

A seleção de atributos foi orientada por informações sobre o domínio do processo fornecida por especialistas humanos. A seleção seguiu o direcionamento das melhores práticas do ITIL e sua implementação na ferramenta utilizada. Na primeira etapa do processo de gerenciamento de incidentes, o solicitante deve fornecer as informações iniciais sobre a situação de instabilidade ou degradação relacionada no incidente. Essa informação é complementada pelo agente do "*Service desk*", especialmente com as informações relacionadas à categoria e prioridade (definida pelo impacto e urgência no ITIL). Informações adicionais (anexos e descrições textuais) também são fornecidas para auxiliar os analistas de suporte que atuarão na etapa seguinte, porém essas últimas, de natureza não estruturada, não foram tratadas no experimento por estar fora do escopo deste trabalho. Seguindo essa orientação de aplicação da prática, os atributos *incident_state*, *category* e *priority* foram considerados os mais adequados para definir corretamente o modelo de processo no STA: o *incident_state* relata o estágio em que se encontra o incidente; *category* indica o tipo de serviço tecnológico ao qual o incidente está associado; e o atributo *priority* determina a necessidade de priorização demandada pelo negócio. Para esse cenário, foram gerados e utilizados como preditor de tempo de conclusão 18 STAs - variando os parâmetros de representação (abstração) do estado e o horizonte máximo. Foi considerada a amostra de registros do log de eventos enriquecido com 24.000 incidentes e os resultados são apresentados na tabela 10. Os melhores resultados, considerando a aferição nos conjuntos de teste, foram obtidos com horizonte máximo 3 e representação de estados *sequência*.

Diante do cenário apresentado na tabela 10, um comportamento observado nos STAs gerados com esse subconjunto de atributos, para todos os horizontes apresentados, é o melhor desempenho da função de predição com a utilização da mediana, quando comparado à utilização da função com a média. Outra observação, os valores nas três

Tabela 10 – Experimento #1 – resultados de predição média. Atributos utilizados: *incident_state*, *category* e *priority*. Amostra de log: 24.000 incidentes. Métrica: MAPE e DP = Desvio padrão. NF = % dos incidentes não reprodutiveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Hor. Máx.**	**Conjunto**			**Multiconjunto**			**Sequência**		
		Média	*Mediana*	NF	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF
MAPE	1	113,93	88,29	0,22	113,93	88,29	0,22	113,93	88,29	0,22
	3	106,93	77,46	0,98	91,35	75,87	1,23	**72,36**	**63,66**	**1,38**
	5	119,18	109,28	1,64	177,05	162,08	2,95	126,12	104,67	3,38
	6	183,52	115,59	1,83	122,54	98,74	3,72	102,73	84,01	4,41
	7	93,22	75,11	1,95	1190,87	1184,75	4,44	107,58	98,04	5,48
	Inf	1.146,57	1.123,24	2,31	92,12	75,21	8,03	88,32	72,98	9,00
DP	1	97,63	85,46	0,05	97,63	85,46	0,05	97,63	85,46	0,05
	3	70,66	56,58	0,13	53,74	52,34	0,21	**43,69**	**42,31**	**0,21**
	5	119,93	123,16	0,19	177,75	175,59	0,33	85,60	81,14	0,33
	6	174,93	96,29	0,22	99,31	84,92	0,35	51,34	43,42	0,34
	7	65,80	61,80	0,21	1672,58	1675,83	0,37	72,92	70,49	0,30
	Inf	1640,91	1629,97	0,26	77,16	64,61	0,48	63,99	52,09	0,35

abstrações de representação para o horizonte 1 são iguais porque, na realidade, há uma equivalência nas abstrações dos estados nessa situação, e STAs iguais são gerados.

Ao observar os resultados de NF, nota-se que esses valores estão entre: 0,22 e 2,31 para o STA gerado com a representaçao de conjunto; 0,22 e 8,03 para o STA gerado com a abstração multiconjunto; 0,22 e 9,00 para o STA gerado com a abstração multiconjunto. Esses resultados, aliados aos baixos valores dos desvios-padrão (entre 0,05 e 0,48) permitem concluir que a combinação do atributo de controle *incident_state* com os demais atributos de classificação foi capaz de gerar STAs com baixo sobreajuste, confirmando que o direcionamento do especialista é compatível com as informações registradas no log de eventos.

Partindo para análise do MAPE, observa-se que o comportamento difere significativamente ao utilizar as formas de representação. Esse comportamento tem uma variação mais evidente quando observa-se o comportamento com horizontes distintos e função de predição média. Esses valores tem um intervalo mais amplo no caso da representação conjunto (entre 93,22 e 1.146,57) e mais restrito e homogêneo com a representação sequência (entre 72,36 e 126,12). Esses dados confirmam a necessidade de avaliação do comportamento de acordo com a variação dos modelos de representação e do horizonte, pois essa variação permite identificar mais adequadamente o comportamento existente no processo. Mais especificamente no STA da representação conjunto, a variação do horizonte teve um efeito de atuação como filtro para os casos com um número maior de eventos (o valor do horizonte 7 é 93,22 e para o horizonte Inf com a mesma função é 1.146,57) tornando o

STA mais preciso, estável e robusto. Note-se que essa não é uma regra geral, ou seja, para formas diferentes de representação as situações podem ser opostas. Observe-se o caso do resultado para o STA com representação multiconjunto e mesmos horizontes citados (7 e Inf), que tem um valor de MAPE elevado para o horizonte 7 e menor para o horizonte Inf. Nesse caso, o horizonte acabou limitando a representação criada pelo STA, fazendo com que comportamentos distintos fossem tratados de forma similar prejudicando a acurácia.

Quanto ao resultado do MAPE obtido com as duas funções de predição distintas (média e mediana), nota-se que, em todas as situações houve predomínio do melhor resultado em favor da função que utiliza a mediana. Esse também é um comportamento observado, quase que na totalidade, no valor do desvio-padrão.

O melhor resultado de predição obtido usando os atributos selecionados pelo conhecimento do especialista produz valor médio para o MAPE 72,36 considerando a previsão usando a função com a média, e de 63,66 considerando a previsão usando a função com a mediana. O desvio-padrão do melhor valor também é o menor entre todas as avaliações, indicando que essa é a configuração mais homogénea dentre as avaliadas. O valor de NF em 1,38 também indica que o STA obtido é capaz de tratar quase a totalidade de situações existentes no processo. Quanto ao valor geral obtido - 63,66 - não é algo que apresente resultados extremamente relevantes, mas são significativamente melhores que a predição utilizando valores de tempo de execução do incidente com o calculo de heurística simples fazendo o agrupamento pelos campos utilizados (*incident_state*, *category* e *priority*), que são 728,59 e 168,31 com as funções a média e mediana respectivamente.

Além de validar o conhecimento do especialista na seleção dos atributos para construção do STA, esse experimento permite concluir que a variação das representações é um item fundamental para a obtenção de modelos mais precisos, pois os melhores resultados para o MAPE foram obtidos com valores distintos dos horizontes – 1 e Inf – habitualmente utilizados na literatura (AALST; SCHONENBERG; SONGA, 2011).

5.1.2 Experimento #2 – Seleção por filtro com *ranking*

A seleção de atributos foi direcionada por informações do método de filtro utilizando uma estratégia de *ranking*. Esta abordagem segue o que foi apresentado na seção 3.4.2, logo, o *ranking* foi aplicado como uma etapa de pré-processamento para criação de uma

ordenação que serve como ponto de partida na seleção de atributos e é independente do modelo de predição escolhido. O *ranking* foi criado usando análise de variância por meio da correlação das variáveis independentes (i.e., atributos descritivos) e a variável dependente (i.e., o atributo "closed", que é o atributo alvo da predição). Uma vez que a maioria dos atributos descritivos são de natureza categórica, a estatística η^2 foi aplicada, seguindo o exposto em Richardson (2011).

Para execução do método, o número máximo de atributos a ser selecionado foi determinado como sendo os 15 atributos com maior correlação. Esses 15 atributos formam a lista de atributos inicial, a qual representa a composição do *ranking*. A análise de variância foi realizada considerando todos os registros existentes no log de eventos enriquecido, de maneira a ter uma representação real desse conjunto e não apenas uma avaliação amostral. Os resultados obtidos para todos os 15 atributos e seus respectivos valores de correlação estão listados na tabela 11.[1] Ao analisar o conteúdo da tabela 11, observa-se que os atributos descritivos que possuem um valor de correlação mais elevado com a variável dependente são aqueles relacionados à perspectiva de recursos associados ao processo de gerenciamento de incidentes.

Tabela 11 – Os 15 atributos descritivos com o maior valor de correlação com o atributo dependente e seus respectivos valores η.

Ordem	**Atributo**	η	**Ordem**	**Atributo**	η	**Ordem**	**Atributo**	η
1	Caller	0,54	6	Incident state	0,32	11	Created by	0,21
2	Assigned to	0,37	7	Subcategory	0,32	12	Opened by	0,20
3	Assignment group	0,35	8	Category	0,27	13	Location	0,14
4	Symptom	0,33	9	Active	0,25	14	Made SLA	0,14
5	Sys updated by	0,33	10	Priority confirmation	0,24	15	Knowledge	0,12

Partindo dos resultados obtidos com o *ranking* de atributos, o método de filtro foi executado utilizando a combinação dos atributos de maneira sequencial da seguinte forma:

{*Caller (1º)*};

{*Caller (1º), Assigned to (2º)*};

...;

{*Caller (1º), Assigned to (2º), ..., Knowledge (15º)*}.

Nesse cenário, foram gerados e utilizados como preditores de tempo de conclusão 15 conjuntos de 18 STAs - variando os parâmetros de representação do estado e horizonte máximo. Foi considerada a amostra de registros do log de eventos enriquecido com 8.000 incidentes e os resultados dos horizontes com melhor acurácia são apresentados na tabela 12.

[1] O valor η calculado a partir da estatística η^2.

Tabela 12 – Experimento #2 – resultados de predição média. Atributos utilizados: selecionados pelo filtro. Amostra de log: 8.000 incidentes. Métrica: MAPE e DP = Desvio padrão. NF = % dos incidentes não reprodutiveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Atr.**	**Hor. Máx.**	**Conjunto**			**Multiconjunto**			**Sequência**		
			Média	*Mediana*	*NF*	*Média*	*Mediana*	*NF*	*Média*	*Mediana*	*NF*
MAPE	1	Inf	160,22	140,99	20,77	114,62	109,79	30,95	114,62	109,79	30,95
	2	**1**	**110,98**	**90,81**	**59,89**	**110,98**	**90,81**	**59,89**	**110,98**	**90,81**	**59,89**
	3	**1**	**112,27**	**88,99**	**63,92**	**112,27**	**88,99**	**63,92**	**112,27**	**88,99**	**63,92**
	4	6	129,41	98,90	72,22	123,72	96,08	72,72	122,83	95,11	72,73
	5	5	128,71	98,52	72,89	128,36	98,11	73,08	128,49	98,15	73,08
	6	Inf	129,25	100,28	73,39	133,72	102,29	73,51	133,72	102,29	73,51
	7	Inf	146,08	117,20	73,58	129,63	98,36	73,70	129,63	98,36	73,70
	8	Inf	143,84	114,87	73,66	129,42	98,06	73,77	129,42	98,06	73,77
	9	Inf	143,84	114,87	73,66	129,42	98,06	73,77	129,42	98,06	73,77
	10	5	130,46	101,07	73,67	133,72	101,61	73,72	139,35	107,19	73,72
	11	3	135,57	103,93	73,65	133,30	101,25	73,67	134,97	102,96	73,67
	12	Inf	147,31	118,41	73,76	130,57	99,36	73,86	130,57	99,36	73,86
	13	7	127,16	97,58	73,78	128,37	98,20	73,87	128,28	98,16	73,87
	14	Inf	124,96	96,09	73,78	126,14	96,85	73,88	126,14	96,85	73,88
	15	Inf	125,70	96,75	73,78	130,25	98,98	73,88	130,25	98,98	73,88
DP	1	Inf	165,86	160,37	0,76	129,57	136,12	0,70	129,57	136,12	0,70
	2	**1**	**102,28**	**97,83**	**0,57**	**102,28**	**97,83**	**0,57**	**102,28**	**97,83**	**0,57**
	3	**1**	**97,36**	**88,17**	**0,47**	**97,36**	**88,17**	**0,47**	**97,36**	**88,17**	**0,47**
	4	6	105,56	91,19	1,35	105,21	91,58	1,36	105,90	91,91	1,37
	5	4	112,64	94,74	1,40	112,38	95,07	1,39	112,50	95,07	1,39
	6	Inf	107,91	92,56	1,47	111,64	97,34	1,44	111,64	97,34	1,44
	7	Inf	104,31	99,69	1,49	107,39	93,69	1,46	107,39	93,69	1,46
	8	Inf	100,86	97,96	1,48	104,60	91,35	1,45	104,60	91,35	1,45
	9	Inf	100,86	97,96	1,48	104,60	91,35	1,45	104,60	91,35	1,45
	10	4	109,36	93,64	1,43	109,46	94,28	1,45	109,63	94,64	1,45
	11	3	106,05	93,23	1,43	108,02	93,39	1,44	107,18	92,60	1,44
	12	Inf	101,52	97,28	1,44	107,47	93,23	1,45	107,47	93,23	1,45
	13	7	110,60	94,08	1,44	110,83	93,86	1,44	110,84	93,88	1,44
	14	Inf	107,56	91,81	1,44	107,17	91,48	1,44	107,17	91,48	1,44
	15	Inf	107,88	92,29	1,44	105,25	91,78	1,44	105,25	91,78	1,44

Os melhores resultados para o MAPE foram obtidos com horizonte 1 e os subconjuntos de atributos {*Caller, Assigned to*} e {*Caller, Assigned to, Assignment group*}, independentemente da forma de abstração utilizada para representação dos estados. Os resultados obtidos mostram um domínio dos subconjuntos de atributos que representam a perspectiva de recursos. Nota-se, porém, que o valor obtido com o NF é elevado, variando de 20,77 para o conjunto número 1 com horizonte máximo Inf até o valor 73,88 com conjutno número 15, horizonte Inf e representação multiconjunto. Em números absolutos, 73,88% significa que, na validação cruzada, utilizando os dados de testes, de um valor médio de 8.862 eventos, 6.540 não são reconhecidos pelos STAs criados. Esses atributos geram STAs que tem uma tendência ao sobreajuste.

Quanto á avaliação do desvio-padrão, pode-se observar que o valor obtido para o conjunto com o melhor valor de MAPE também é o melhor valor. No caso do desvio-padrão do NF, os valores de ambos os conjuntos (2 e 3) são também os menores em todos os conjuntos e abstrações, seguindo o mesmo comportamento observado no experimento #1.

Tabela 13 – Experimento #2 – resultados de predição média. Atributos utilizados: melhores subconjuntos de atributos selecionados pelo filtro com ranking. Amostra de log: 24.000 incidentes. Métrica: MAPE. NF = % dos incidentes não reprodutiveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Hor. Máx.**	**Conjunto**			**Multiconjunto**			**Sequência**		
		Média	*Mediana*	NF	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF
Subconjunto de atributos: {*caller, assigned_to*}										
MAPE	1	208,61	196,42	30,10	208,61	196,42	30,10	208,61	196,42	30,10
	3	102,09	89,17	32,48	86,41	72,50	33,87	98,69	84,37	33,90
	5	90,73	76,30	33,31	**69,69**	**57,85**	**35,67**	80,97	69,10	35,73
	6	292,51	280,42	33,44	77,53	65,66	36,15	82,78	70,92	36,20
	7	171,55	159,95	33,51	91,22	79,66	36,41	103,14	90,27	36,46
	Inf	249,06	238,05	33,60	96,66	85,85	36,73	78,82	67,97	36,76
DP	1	200,06	190,6	0,61	200,06	190,6	0,61	200,06	190,6	0,61
	3	73,96	67,52	0,46	62,74	53,53	0,52	80,29	67,32	0,53
	5	56,86	44,61	0,33	**50,49**	**39,36**	**0,61**	56,86	44,12	0,59
	6	320,68	311,29	0,36	34,98	25,08	0,66	45,17	33,45	0,65
	7	158,09	146,46	0,36	63,98	52,54	0,67	70,34	59,86	0,65
	Inf	251,58	242,55	0,32	67,27	59,56	0,63	45,91	38,07	0,62
Subconjunto de atributos: {*caller, assigned_to, assignment_group*}										
MAPE	1	80,17	67,87	34,04	80,17	67,87	34,04	80,17	67,87	34,04
	3	93,16	80,65	37,48	102,64	86,15	38,58	131,73	118,08	38,67
	5	91,34	80,96	39,22	**76,21**	**64,98**	**40,67**	86,20	74,89	40,75
	6	85,55	74,76	39,58	94,38	83,01	41,04	78,05	66,67	41,11
	7	96,99	85,00	39,76	102,01	86,35	41,19	105,66	94,33	41,25
	Inf	85,96	74,00	40,03	81,33	70,36	41,33	79,76	68,76	41,36
DP	1	54,73	44,94	0,66	54,73	44,94	0,66	54,73	44,94	0,66
	3	74,77	64,08	0,56	80,28	69,46	0,63	82,72	72,18	0,65
	5	55,50	44,38	0,64	**51,50**	**40,84**	**0,75**	61,07	48,18	0,77
	6	71,19	58,28	0,66	58,87	47,26	0,74	68,03	56,09	0,75
	7	83,75	70,49	0,64	77,95	59,25	0,71	116,69	104,16	0,73
	Inf	71,99	59,17	0,66	73,33	61,76	0,72	68,23	56,84	0,73

Os resultados de predição obtidos com os STAs gerados a partir dos dois melhores subconjuntos de atributos segundo o *ranking* foram comparados com os resultados obtidos no experimento #1. Dois novos conjuntos de STAs foram gerados utilizando os parâmetros dos melhores resultados listados na tabela 12 (linhas 2 e 3), entretanto, com a utilização da amostra de 24.000 incidentes do log de eventos enriquecido. Os resultados são apresentados na tabela 13.

Ao analisar o comportamento dos experimentos apresentados, nota-se a predominância (independentemente do conjunto de atributos utilizado) da representação multiconjunto quanto à melhor acurácia do MAPE quando comparado às demais formas. Vale destacar os valores muito ruins obtidos com a representação conjunto no subconjunto de atributos {*caller, assigned_to*}.

Utilizando agora a avaliação de comportamento do MAPE conforme a variação do horizonte na representação multiconjunto, é possível observar que os melhores valores

são os obtidos com o valor da mediana do número de eventos. Porém, no primeiro subconjunto {*caller, assigned_to*}, o comportamento é de início com um valor elevado e gradativamente reduzido até chegar ao horizonte 5 e depois vai gradativamente piorando (elevando) o valor à medida que o horizonte é incrementado. No subconjunto {*caller, assigned_to assignment_group*} o comportamento é diferente, pois os horizontes 1 e Inf têm valores muito similares e pouco acima do melhor valor obtido com o horizonte 5. Outro comportamento observado no indicador de NF é a tendência ao aumento desse valor com o acréscimo do atributo {*assignment_group*} que tornou o STA mais especializado e sobreajustado.

Os resultados obtidos com os subconjuntos de atributos gerados via *ranking* são ligeiramente melhores, cerca de 9,12%, do que os obtidos a partir do experimento #1. Ao realizar uma análise mais detalhada nesses resultados, observa-se que, de maneira geral, os atributos relacionados à perspectiva dos recursos conseguem obter um desempenho relevante para o modelo de predição gerado. Porém, ao observar o parâmetro de NF dos STAs, é possível identificar que tais modelos não conseguem refletir o comportamento do processo com a mesma fidelidade apresentada por STAs gerados pelos atributos de controle (i.e., incident_state). Uma explicação possível para os resultados piores de NF pode estar relacionada com as alterações frequentes em relação aos recursos humanos (férias, substituições, etc) associados à resolução dos diferentes tipos de incidentes.

5.1.3 Experimento #3 – Invólucro com subida de encosta e com busca pela primeira melhora

A seleção de atributos pelo método de invólucro foi executada utilizando o modelo de seleção incremental (do inglês, forward selection)[2], com os procedimentos de busca da subida da encosta e busca pela primeira melhora, ambos descritos na seção 2.2.2 e, de maneira contextualizada no problema tratado neste trabalho na seção 3.4.3. O espaço de busca a ser explorado é composto por todas as combinações possíveis dos 15 atributos pré-selecionados pelo procedimento de filtro com utilização da estratégia de *ranking*, i.e., são os atributos listados na tabela 11. Uma vez que cada combinação representa um estado em tal espaço, na qual a medida de avaliação da qualidade é calculada como sendo a capacidade

[2] No modo de seleção incremental, o estado inicial da busca é um subconjunto único de atributos que tem um novo atributo adicionado a cada passo do processo de busca.

preditiva atingida pelos STAs gerados com o subconjunto de atributos selecionado nesse modelo[3], um procedimento de busca exaustiva (força bruta) seria impraticável, logo, procedimentos de busca heurística se fazem necessários. Para o procedimentos de busca pela primeira melhora, o número máximo de movimentos de expansão dos estados sem que exista melhoria na acurácia foi configurado para o valor 15.

O método de invólucro foi executado no log de eventos enriquecido com 8.000 e 12.000 amostras de incidentes selecionadas de forma aleatória. Os resultados obtidos pelo melhor estado selecionado pelas buscas nessas condições estão listados na tabela 14. Os dois procedimentos de busca (subida de encosta e busca pela primeira melhora) apresentaram o mesmo resultado para a seleção do melhor subconjunto de atributos, sendo respectivamente, {*incident_state, location*} para o conjunto de dados com 8.000 incidentes e {*u_priority_confirmation, active, location, made_sla*} para o conjunto de dados com 12.000 incidentes.

Apesar dos resultados obtidos nos dois métodos de busca serem idênticos, algumas informações complementares podem ser extraídas de suas execuções, sendo:

1. Experimento de busca com amostra de 8.000 registros
    - **Subida de encosta**: o critério de parada foi atingido após a expansão do terceiro nível de busca; foram explorados 42 estados do espaço de busca; ao aplicar o cálculo da estatística média em todos os STA criados na representação *conjunto*, os valores dos resultados de MAPE foram 146,80 para média e 103,76 para mediana. Os percentuais de NF apresentaram o valor médio de 8,97%.
    - **Primeira melhora**: foram executados 17 movimentos de expansão e explorados 172 estados do espaço de busca; o valor para a média de resultados do MAPE, considerando todos os STAs na representação *conjunto*, foi de 114,96 utilizando a média e 89,68 utilizando a mediana respectivamente. O valor de NF foi 36,27.
2. Experimento de busca com amostra de 12.000 registros
    - **Subida da encosta**: o critério de parada foi atingido após a expansão do quarto nível de busca e tendo sido explorados 65 estados do espaço de busca.
    - **Primeira melhora**: foram executados 19 movimentos de expansão e explorados 197 estados do espaço de busca.

[3] O espaço de busca possui um total de $2^{15} = 32.768$ estados, considerando 18 STAs gerados para cada estado, o intervalo dos horizontes e os modelos de abstração selecionados.

Tabela 14 – Experimento #3 – resultados de predição média. Atributos utilizados: selecionados pelo invólucro. Amostra de log: 8.000 e 12.000 incidentes respectivamente. Métricas: MAPE e DP = Desvio-padrão. NF = % dos incidentes não reprodutiveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Hor.**	**Conjunto**			**Multiconjunto**			**Sequência**		
	Máx.	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF
Subconjunto de atributos: {*incident_state, location*}										
MAPE	1	501,18	450,23	0,88	501,18	450,23	0,88	501,18	450,23	0,88
	3	528,98	522,63	1,92	497,56	475,72	2,70	92,71	64,01	2,96
	5	185,12	66,39	2,51	113,64	84,77	5,71	143,45	72,07	6,60
	6	33,90	19,51	2,58	43,02	23,74	6,91	33,85	22,87	8,19
	7	**17,82**	**10,13**	**2,69**	21,36	15,19	8,07	25,07	15,46	9,74
	Inf	60,69	42,95	2,92	251,79	230,73	14,01	239,53	218,17	15,50
DP	1	876,85	871,88	0,38	876,85	871,88	0,38	876,85	871,88	0,38
	3	368,71	365,74	0,34	356,91	335,9	0,38	82,52	53,89	0,41
	5	227,25	53,43	0,34	99,53	70,75	0,65	147,41	67,36	0,82
	6	27,06	17,99	0,31	33,73	15,28	0,62	23,34	17,81	0,81
	7	**9,64**	**6,12**	**0,29**	10,03	9,08	0,58	16,76	11,02	0,73
	Inf	49,73	31,40	0,44	241,81	220,51	0,72	242,34	220,95	0,75
Subconjunto de atributos: {*u_priority_confirmation, active, location, made_sla*}										
MAPE	1	42,79	26,40	0,59	42,79	26,40	0,59	42,79	26,40	0,59
	3	64,02	60,53	0,83	40,89	37,66	1,06	40,89	37,65	1,06
	5	23,20	17,32	0,85	22,46	13,60	1,59	22,46	13,60	1,59
	6	44,55	22,44	0,83	30,16	23,97	1,81	30,15	23,95	1,82
	7	44,28	23,11	0,82	21,20	19,30	2,00	21,19	19,29	2,01
	Inf	38,48	18,37	0,78	**16,32**	**13,18**	**3,46**	**16,32**	**13,18**	**3,46**
DP	1	32,67	10,25	0,19	32,67	10,25	0,19	32,67	10,25	0,19
	3	107,69	105,57	0,24	53,16	52,33	0,31	53,16	52,32	0,31
	5	15,09	13,11	0,27	14,90	14,02	0,37	14,90	14,03	0,37
	6	44,82	15,37	0,26	18,75	13,41	0,36	18,73	13,38	0,37
	7	51,52	16,45	0,26	9,33	8,01	0,40	9,33	8,00	0,41
	Inf	51,33	18,08	0,18	**13,78**	**11,38**	**0,58**	**13,78**	**11,38**	**0,58**

O esforço de busca adicional, independentemente do número de registros da amostra utilizada, explorou um número muito maior de estados e apresentou valores médios menores para o MAPE, porém, não foi capaz de produzir resultados relevantes no contexto do processo avaliado.

Quanto aos resultados de acurácia, os melhores foram obtidos com a amostra de 8.000 incidentes, horizonte 7 e modelo de abstração para representação de estados *conjunto*. Esse também foi o horizonte que apresentou o menor desvio-padrão; entretanto, os resultados obtidos com os outros modelos de representação de estados para o mesmo horizonte são muito promissores também. A variação do horizonte, partindo do valor 1 até o valor 7 é outro item que apresentou comportamento contínuo de redução do MAPE. A exceção foi o valor Inf que apresentou um valor elevado e portanto foi incapaz de capturar o comportamento adequado do processo. Este é mais um item que confirma a proposta

desse trabalho com relação à utilização de valores distintos para o horizonte durante o processo de construção do STA.

Analisando o resultado obtido pela amostra de 12.000 incidentes, observa-se que os melhores resultados foram obtidos com o horizonte "infinito" e modelo de abstração para representação de estados *multiconjunto*. Da mesma forma que o observado com a amostra de 8.000 incidentes, o resultado para o STA com modelo de representação *conjunto* apresentou bons resultados e o resultado do modelo sequência é idêntico ao do multiconjunto. Observando-se os demais horizontes, nota-se que o horizonte 5 (mediana dos eventos) possui valores muito próximos aos do horizonte infinito, porém, os valores de NF são menores.

Como forma de comparação, pode-se observar que os subconjuntos selecionados em ambas as buscas apresentam resultados próximos, porém, com a utilização de apenas um atributo em comum. É importante destacar que o processo de seleção gerou subconjuntos que destacam a perspectiva de controle do processo de gerenciamento de incidentes aliada à perspectiva organizacional com o atributo *location*. Ao se fazer uma comparação com o experimento #2 é possível observar que o valor obtido para o MAPE – 10,13 contra 88,99 – nota-se que os resultados obtidos são significativamente melhores, independentemente da função de medição utilizada para realizar a predição ser a média ou a mediana. De maneira geral, os baixos índices de não NF e desvios-padrão indicam que os resultados são muito promissores.

A segunda parte do experimento #3 tem por objetivo fazer uma comparação dos resultados de predição obtidos com os STA gerados nos subconjuntos de atributos selecionados pelo invólucro com os resultados obtidos nos experimentos #1 e #2. Novos conjuntos de STAs foram gerados usando os subconjuntos selecionados pelos dois subconjuntos de amostras (8.000 e 12.000), entretanto a amostra do log de eventos enriquecido utilizada possui 24.000 incidentes. Os resultados obtidos estão apresentados na tabela 15, sendo possível observar que os melhores resultados são os do subconjunto de atributos {*incident_state, location*}, com horizonte máximo 5 e STA criado com a representação *conjunto*. As demais formas de representação (sequência e multiconjunto) também apresentaram os segundo e terceiros melhores resultados respectivamente, demonstrando que esse parâmetro de horizonte máximo é o que apresenta a melhor acurácia.

Analisando o comportamento do melhor valor do MAPE para o subconjunto {*incident_state, location*}, é possível verificar que há similaridade com o experimento #2

Tabela 15 – Experimento #3 – resultados de predição média e de desvios-padrão do MAPE da predição média obtida apresentada. Atributos utilizados: melhores subconjuntos de atributos selecionados pelo invólucro. Amostra de log: 24.000 incidentes. Métricas: MAPE e DP = Desvio-padrão. NF = % dos incidentes não reprodutíveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Hor.**	**Conjunto**			**Multiconjunto**			**Sequência**		
	Máx.	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF
Subconjunto de atributos: {*incident_state, location*}										
MAPE	1	138,60	97,59	0,35	138,60	97,59	0,35	138,60	97,59	0,35
	3	107,69	52,48	0,85	69,02	47,17	1,09	65,57	37,25	1,22
	5	**50,45**	**24,49**	**1,11**	41,90	29,35	2,30	35,09	27,28	2,74
	6	69,32	48,98	1,13	59,71	52,16	2,95	57,13	47,21	3,57
	7	132,81	110,51	1,16	153,96	114,83	3,57	68,53	56,39	4,36
	Inf	66,75	46,16	1,24	43,02	35,86	6,51	70,54	38,26	7,43
DP	1	95,45	97,38	0,12	95,45	97,38	0,12	95,45	97,38	0,12
	3	62,47	25,75	0,15	23,18	19,90	0,20	24,28	12,07	0,21
	5	**32,50**	**18,83**	**0,18**	36,18	23,86	0,25	24,02	19,62	0,28
	6	54,87	43,61	0,18	36,76	34,15	0,21	44,99	45,74	0,22
	7	155,63	140,73	0,19	178,37	132,48	0,15	65,98	54,03	0,16
	Inf	52,42	41,70	0,18	26,97	26,66	0,32	56,65	25,58	0,28
Subconjunto de atributos: {*u_priority_confirmation, active, location, made_sla*}										
MAPE	1	54,62	39,55	0,27	54,62	39,55	0,27	54,62	39,55	0,27
	3	51,18	38,42	0,41	**61,46**	**34,86**	**0,52**	**61,49**	**34,88**	**0,52**
	5	57,71	41,64	0,40	65,88	50,90	0,81	65,83	50,85	0,82
	6	55,82	40,54	0,41	75,80	62,53	0,93	75,81	62,54	0,94
	7	62,60	44,74	0,42	51,54	38,40	1,06	51,55	38,41	1,07
	Inf	98,46	84,53	0,37	156,06	148,58	1,85	156,07	148,58	1,87
DP	1	29,46	27,63	0,15	29,46	27,63	0,15	29,46	27,63	0,15
	3	32,67	31,22	0,12	**48,22**	**24,53**	**0,12**	**48,27**	**24,56**	**0,11**
	5	32,34	31,54	0,11	33,44	32,49	0,14	33,38	32,42	0,13
	6	32,92	30,76	0,12	58,16	60,71	0,14	58,18	60,72	0,13
	7	44,24	37,39	0,13	35,02	23,43	0,16	35,03	23,44	0,16
	Inf	78,23	83,01	0,13	145,78	142,11	0,20	145,78	142,11	0,20

em relação ao horizonte, pois apresenta melhoria desse valor de forma contínua entre o horizonte 1 e 5 passando a ter valores piores nos horizontes seguintes. Esse comportamento também é observado no desvio-padrão, demonstrando que os resultados são consistentes. Ao fazer a avaliação do valor de NF pode-se observar uma similaridade com o comportamento do MAPE, ou seja, há um incremento para os modelos com representação multiconjunto e sequência, porém, seus respectivos desvios-padrão são baixos demonstrando consistência nos modelos criados.

No caso do subconjunto de atributos {*u_priority_confirmation, active, location, made_sla*}, destaca-se que os melhores resultados foram obtidos com o horizonte 3 e os modelos com representação multiconjunto e sequência. Apesar de terem um valor superior ao do subconjunto {*incident_state, location*}, os valores de NF são inferiores 0,52% contra

1,11% e portanto indica uma capacidade de representação do processo melhor e capaz de tratar quase que a totalidade dos eventos registrados.

Os resultados obtidos com os dois subconjuntos de atributos selecionados superam os obtidos nos experimentos anteriores, sendo que os resultados para o MAPE (24,49) representam 38,47% do obtido na seleção realizada utilizando o conhecimento do especialista (63,66) e 42,33% daquele obtido com a utilização do filtro (57,85). No caso do parâmetro de NF, pode-se notar que os resultados também são menores (1,11) quando comparados com os obtidos com conhecimento do especialista (1,38) e filtro (35,67) respectivamente.

5.1.4 Experimento #4 – Invólucro com algoritmo genético

A seleção de atributos por invólucro, usando algoritmo genético, foi realizada utilizando a estratégia básica com a variação dos parâmetros taxas de reprodução, cruzamento e mutação, conforme definições apresentadas na seção 3.4.5 e a teoria apresentada na seção 2.3.

A aplicação de algoritmos genéticos como seletor de atributos impôs um alto custo computacional à execução dos experimentos, principalmente porque a avaliação dos cromossomos se dá por meio da criação de STAs. O número de STAs criados durante a execução do algoritmo genético é equivalente ao número de indivíduos multiplicado pelo número de gerações. Assim, foram realizados apenas experimentos com subconjuntos de amostras com a variação de parâmetros apresentada na tabela 16.

Tabela 16 – Experimento #4 – Variação de parâmetros

Tamanho da amostra de traços	{ 1.000}, {8.000 }	seguindo a estratégia de amostragem aleatória
Quantidade de atributos candidatos	15	atributos vindos da seleção por filtro com *ranking*
Abstração de representação de estados	{ conjunto, multiconjunto, sequência }	
Tamanho da população	{48} e {16, 32, 60}	para as amostras de 1.000 e 8.000 respec.
Número de gerações	{40} e {40, 20, 10}	para as amostras de 1.000 e 8.000 respec
Taxa de reprodução	{0,25}	usada apenas para a amostra de 1.000
Taxa de crossover	{0,68} e {0,93, 0,87, 0,75}	para as amostras de 1.000 e 8.000 respec
Taxa de mutação	{0,07} e {0,07, 0,13, 0,25}	para as amostras de 1.000 e 8.000 respec

Importante salientar que não foram realizados experimentos para todas as possíveis combinações desses parâmetros. As taxas de mutação e crossover foram estabelecidas de acordo com o tamanho da população usada, e o numero de gerações diminuiu conforme o

número de indivíduos da população aumentou. O único parâmetro que se manteve variando em todas as possibilidades em todos os experimentos foi a abstração de representação de estados. Assim, por clareza, as combinações de parâmetros referentes à tamanho de população, número de gerações e taxas dos operadores para o uso com amostra de 8.000 traços são:

- **execução 1-8000-A**: 16, 40, 0,93 e 0,07;
- **execução 2-8000-B**: 32, 20, 0,87 e 0,13;
- **execução 3-8000-C**: 64, 10, 0,75 e 0,25.

Os resultados obtidos nas execuções de algoritmos genéticos, bem como os parâmetros utilizados em cada uma delas seguem listados na tabela 17.

Tabela 17 – Experimento #4 – resultados de atributos selecionados, horizonte máximo e erro de predição. Métricas: MAPE com a "estatística média" e NF = % dos incidentes não reprodutíveis pelo STA (non-fitting). Negrito: melhores resultados.

Exec.	**Subconjunto de atributos**	**Hor. Máx.**	**Conjunto**			**Multiconjunto**			**Sequência**		
			Média	Med.	NF	Média	Med.	NF	Média	Med.	NF
1	assigned_to, assignment_group, u_symptom, incident_state, subcategory, category, active, u_priority_confirmation	Inf	**102,69**	**87,99**	**55,64**	105,57	89,87	58,76	105,46	89,69	58,87
2	incident_state, subcategory, active, u_priority_confirmation, sys_created_by	7	**42,24**	**30,77**	**21,12**	52,03	37,57	32,22	55,91	41,34	32,99
3	u_symptom, incident_state, category, active, u_priority_confirmation	7	**80,18**	**39,24**	**16,07**	77,66	42,41	27,45	61,98	39,92	28,43
4	incident_state, category, subcategory, u_symptom, assignment_group, assigned_to	Inf	1,89	1,55	3,74	**1,28**	**0,93**	**6,56**	1,28	0,93	6,56

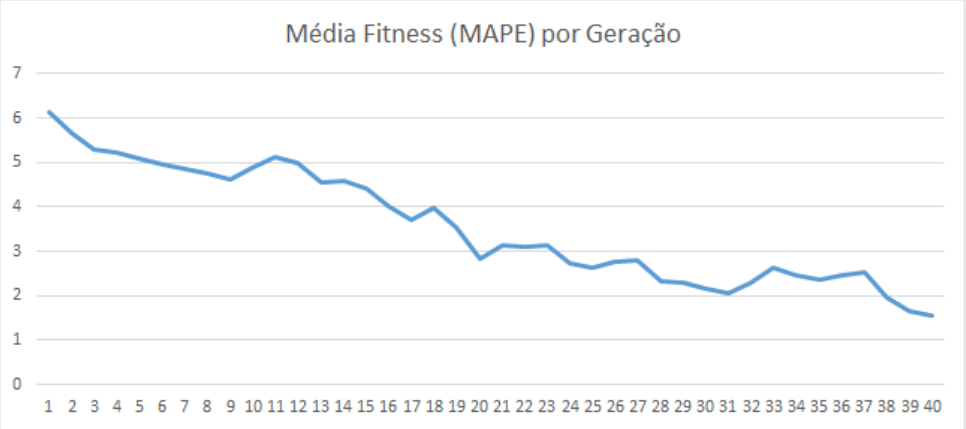
Os experimentos objetivaram avaliar a aplicação da busca com o algoritmo genético e sua viabilidade de aplicação no processo de seleção de atributos. Pode-se observar que o cenário da execução 8000-A, com 8.000 registros e uma população reduzida não conseguiu explorar adequadamente o espaço de estados e produzir um valor de acurácia relevante (comparando-o com os demais experimentos). Entretanto, convergiu para um subconjunto de atributos com oito elementos, o que pareceu ser improvável na exploração realizada nos demais experimentos, dada sua natureza incremental e o número de expansões realizado.

O cenário da execução 8000-B apresentou comportamento que produziu um valor de MAPE mais relevante, selecionando 5 atributos, sendo a combinação de 4 atributos de controle e classificação mais um atributo relacionado ao usuário que efetuou o registro da informação, o qual faz parte da perspectiva de recursos. O cenário da execução 8000-C

produziu um resultado com 5 atributos, todos relacionados à perspectiva de controle e classificação, com valores absolutos do MAPE piores que os do cenário anterior 8000-B.

A execução do #4 do algoritmo genético, com a utilização de uma amostra reduzida de 1000 incidentes, apresentou um resultado relevante para o valor final do MAPE (1,55 com a função mediana) e o valor de NF obtido (3,74). O melhor conjunto, contendo 5 atributos, é composto por atributos da perspectiva de controle, classificação e recursos, ou seja, bem diversificada. Com a configuração utilizada, número menor de instâncias de processo e população maior, foi possível identificar que o algoritmo apresentou um comportamento de decréscimo continuo ao fazer o cálculo da média populacional valor da função de avaliação (fitness). Este valor, calculado em cada uma das 40 gerações é apresentado na figura 11.

Figura 11 – Experimento com 1000 registros - Média do Fitness a cada geração.



Tal como nos demais experimentos, a segunda etapa do experimento #4 tem por objetivo fazer uma comparação dos resultados de MAPE e NF obtidos com os STAs gerados nos subconjuntos de atributos selecionados, com os resultados obtidos nos experimentos #1, #2 e #3. Novos conjuntos de STAs foram criados usando os subconjuntos selecionados, sendo, um conjunto de STAs com o subconjunto de melhor resultado da execução de 8.000 incidentes (Ex. #2, da tabela 17) e outro conjunto com o resultado da execução com 1.000 incidentes (Ex. #4, da tabela 17). Os resultados obtidos estão apresentados na tabela 18, na qual é possível observar que os melhores são os do subconjunto de atributos {*incident_state, category, subcategory, u_symptom, assignment_group, assigned_to*}, utilizando horizonte máximo 5 e representação *sequência*. As demais formas de representação (conjunto e multiconjunto) apresentaram resultados relevantes, porém, equiparados aos STAs criados

com o horizonte máximo 7, demonstrando que há uma similaridade de comportamento para o MAPE entre esses horizontes.

Ao comparar o resultado obtido com o experimento #1, nota-se que o resultado obtido para o MAPE é inferior (34,87 contra 63,66), superando a média obtida na seleção pelo especialista. Porém, o valor de NF é muito superior (32,09 contra 1,38), ou seja, o modelo gerado tem um indicação de sobreajuste e os atributos selecionados tem uma capacidade limitada de representação do processo.

Na comparação com o resultado obtido com o experimento #2, os resultados obtidos com os atributos {*incident_state, category, subcategory, u_symptom, assignment_group, assigned_to*} são muito similares com relação ao MAPE e melhores ao avaliar a questão do NF que tem o valor de 9,98%. Ao fazer a comparação da seleção com o resultado obtido com os atributos {*incident_state, category, subcategory, u_symptom, assignment_group, assigned_to*}, observa-se valores equivalente de NF, mas um valor de MAPE muito inferior. Dessa maneira, considerando a média, os resultados são melhores que os obtidos no experimento #2. Por fim, ao fazer a comparação com o experimento #3, nota-se que os resultados de MAPE são piores, principalmente com relação ao índice de NF.

5.2 Considerações finais

Ao analisar os resultados obtidos nos experimentos executados, foi possível identificar que os resultados do experimento #1 e do experimento #2, quando comparados no indicador referente à acurácia, permitem a construção de STAs com capacidade preditiva muito similar em se tratando do tempo para conclusão dos incidentes. Entretanto, ao realizar uma avaliação do indicador de NF dos STAs gerados, há uma diferença significativa entre os modelos obtidos. Os melhores resultados são de 1,38% para o experimento #1 (conhecimento de especialista) e 35,67% para o experimento #2 (modelo de filtro) respectivamente. Os comportamentos distintos observados nos STAs gerados são causados pelas diferentes perspectivas de processo representadas pelos subconjuntos de atributos utilizados em cada um dos cenários. Nota-se ainda que os testes estatísticos (Tabela 19) realizados apresentaram um valor para o *p-value* de 0,3125 que indica a manutenção da equivalência entre as distribuições obtidas com os valores de MAPE na validação cruzada.

Tabela 18 – Experimento #4 – resultados de predição média e de desvios-padrão do MAPE da predição média obtida apresentada. Atributos utilizados: melhores subconjuntos de atributos selecionados pelo invólucro genético. Amostra de log: 24.000 incidentes. Métricas: MAPE e DP = Desvio-padrão. NF = % dos incidentes não reprodutíveis pelo STA (non-fitting). Negrito: melhores resultados.

Métrica	**Hor.**	**Conjunto**			**Multiconjunto**			**Sequência**		
	Máx.	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF	*Média*	*Mediana*	NF
Subconjunto atributos:{*incident_state,subcategory,active,u_priority_confirmation,sys_created_by*}										
MAPE	1	107,32	101,52	5,13	107,32	101,52	5,13	107,32	101,52	5,13
	3	96,49	85,36	8,31	73,79	62,18	9,41	149,67	133,95	9,65
	5	145,91	129,54	9,62	333,2	311,35	13,49	141,54	121,44	14,01
	6	96,02	78,77	9,85	88,79	69,39	14,95	121,02	104,41	15,52
	7	**74,82**	**60,37**	**9,98**	97,21	81,84	16,10	82,01	65,47	16,64
	Inf	177,02	170,57	9,99	90,38	77,13	18,34	142,38	128,32	18,63
DP	1	96,35	99,15	0,24	96,35	99,15	0,24	96,35	99,15	0,24
	3	78,47	81,46	0,33	40,97	38,92	0,35	118,12	118,95	0,31
	5	115,42	117,28	0,4	394,66	380,46	0,5	117,56	115,24	0,46
	6	78,85	76,38	0,42	68,97	62,42	0,58	102,82	96,86	0,52
	7	**53,76**	**51,18**	**0,41**	75,04	69,25	0,56	62,56	51,56	0,5
	Inf	266,76	271,25	0,51	84,69	78,94	0,6	149,45	143,25	0,58
Subconjunto atributos:{*incident_state,category,subcategory,u_symptom,assignment_group,assigned_to*}										
MAPE	1	75,54	65,27	18,06	75,54	65,27	18,06	75,54	65,27	18,06
	3	116,97	103,52	26,68	68,91	55,15	27,97	85,81	71,08	28,27
	5	56,75	40,91	29,29	63,25	44,82	31,86	**53,34**	**34,87**	**32,09**
	6	88,29	72,48	29,65	75,53	57,92	32,40	89,75	72,10	32,56
	7	55,76	40,07	29,8	57,46	39,26	32,58	58,85	40,59	32,71
	Inf	132,83	116,18	29,97	74,32	55,53	32,69	72,52	54,05	32,77
DP	1	61,09	55,38	0,63	61,09	55,38	0,63	61,09	55,38	0,63
	3	126,65	123,37	0,73	51,22	46,41	0,83	73,47	68,60	0,77
	5	45,34	34,51	0,67	40,38	26,96	0,73	**43,71**	**29,02**	**0,70**
	6	74,13	71,03	0,67	58,09	48,65	0,70	81,95	75,82	0,69
	7	35,16	27,11	0,67	33,62	22,23	0,70	46,47	33,16	0,70
	Inf	129,21	128,07	0,63	51,05	41,10	0,72	46,88	37,82	0,70

Tabela 19 – Resultados para os p-value dos testes estatísticos Wilcoxon pareados comparativos dos valores de MAPE obtidos no experimento #1 contra os obtidos nos experimentos #2, #3 e #4. Amostra de log: 24.000 incidentes.

Experimento #2		Experimento #3		Experimento #4	
2.1	2.2	3.1	3.2	4.1	4.2
0,3125	0,3125	**0,0312**	0,0625	0,2188	0,1562

No primeiro experimento, a geração dos STAs foi conduzida pelos atributos descritivos do processo de gerenciamento de incidentes sugeridos na literatura de melhores práticas do ITIL, aliada à análise de especialistas humanos dessa área de atuação que procuram buscar a melhor forma de agrupamento e roteamento dos incidentes para os respectivos analistas. Dessa forma, o modelo criado foi capaz de representar com precisão o processo de gerenciamento de incidentes. No segundo experimento, o subconjunto de atributos selecionado para geração dos STAs representam a perspectiva organizacional e

de recursos associada ao processo de gerenciamento de incidentes. Neste cenário, os STAs capturaram a forma como as equipes e pessoas estão organizadas para suportar as solicitações dos usuários. Tornaram-se altamente especializados e pouco capazes de generalizar e representar o comportamento real do processo. Essencialmente, o comportamento do modelo foi direcionado pelos atributos selecionados que presumivelmente sofrem alterações com frequência ("solicitante" e "equipe técnica" encarregada de fazer o tratamento do incidente). Esse indicador de sobreajuste, acaba por inviabilizar a utilização dos STAs gerados como preditores, pois 35,67% dos incidentes não serão reconhecidos e portanto não será possível fazer as estimativas de tempo para conclusão de forma assertiva com o modelo deixando de tratar esse número elevado de eventos.

Os experimentos de busca com invólucro – subida da encosta e primeira melhora – no cenário de amostra com 8.000 incidentes, conseguiram obter um valor de MAPE médio utilizando a função predição mediana de 24,49. Esse valor representa apenas 38,47% do valor obtido como o melhor resultado médio de MAPE do experimento #1. Ao aplicar o teste estatístico, obteve-se um valor de 0,0312 (Tabela 19) para o *p-value* que é inferior o valor de significância 0,05 adotado como referência para rejeição de hipótese nula. Dessa forma, pode-se tomar como verdadeira a afirmação que o resultado obtido é melhor do que o processo de escolha manual do experimento #1. Outro ponto relevante é que, ao observar os valores de NF, estes mantiveram-se em um patamar ligeiramente inferior a esse mesmo experimento. Observando-se o subconjunto de atributos selecionado, nota-se que o resultado do processo de busca gerou uma combinação para a construção desse STA de melhor acurácia que é a união de atributos de controle do processo com atributos da perspectiva organizacional. Esse comportamento observado credencia os STAs obtidos no experimento #3 à utilização para geração de estatísticas de predição do tempo restante para conclusão de incidentes.

Ao analisar o resultado obtido pelos mesmos experimentos com invólucro, utilizando a amostra de 12.000 incidentes, os resultados obtidos para o MAPE médio continuam a ser melhores, porém, o teste estatístico comparativo com o experimento #1 apontou um valor de 0,0625, que é insuficiente para rejeição da hipótese nula.

Outro ponto a ser destacado é que os resultados obtidos nos processos de busca subida de encosta e busca pela primeira melhora são idênticos. Este tipo de comportamento também foi observado em experimentos realizados por Kohavi e John (1997), nos quais,

para diferentes tipos de conjuntos de dados o esforço adicional de busca não produziu resultados melhores.

A utilização do algoritmo genético foi proposta como uma forma de avaliar a possibilidade alternativa de aplicação [4]. Os experimentos indicam de que é possível obter valores para a acurácia relevantes e resultados melhores do que aqueles obtidos nos experimentos que utilizaram o conhecimento do especialista e os experimentos com o filtro, mas, os resultados obtidos para o NF limitam sua utilização, pois não refletem adequadamente o comportamento esperado do processo.

Um item a ser destacado é a diversidade da solução obtida, que conta com conjuntos de atributos bem distintos dos que foram obtidos pelos demais experimentos realizados. Esse comportamento demonstra que outros pontos do espaço de hipóteses foram explorados por essa solução. Os parâmetros validados na amostra reduzida permitem que sejam realizados outros experimentos, com amostras maiores, e indicam a possibilidade da obtenção de resultados mais otimizados quando comparados aos demais métodos avaliados.

[4] Pelo número de bits utilizado para a representação (18), outras abordagens seriam possíveis. Porém, para aplicação em um contexto completo no processo de incidentes (37 atributos, 3 abstrações e até 58 horizontes) seriam necessários 45 bits, que justificam a escolha do método

6 Conclusão

O objetivo deste trabalho foi criar um processo de seleção de atributos que pudesse tornar a aplicação de modelos de transição anotados mais assertiva ao realizar uma estimativa do tempo de conclusão. Para isso, uma solução baseada em análise do log de eventos do sistema foi projetada com a utilização de técnicas de seleção por filtro e por invólucro, algoritmos de busca heurística (subida de encosta e busca pela primeira melhora) e meta heurística (algoritmos genéticos). Uma série de experimentos foram executados e organizados em: experimentos exploratórios para confirmar o tipo de modelo de processo real e validar a hipótese em sua forma inicial; experimentos realizados para validação das alternativas propostas na abordagem de seleção de atributos e efetuar a comparação com as recomendações apresentadas na literatura.

Quanto aos objetivos específicos delineados para esse trabalho defende-se que o primeiro, relacionado à criação do ambiente de experimentação, foi atendido, pois foi criado um ambiente de experimentação no qual é possível realizar a exploração de todos os elementos necessários ao estudo de seleção. O segundo objetivo, relacionado ao estabelecimento de uma estratégia de avaliação da seleção de atributos com o STA, foi atendido com a criação das formas de avaliação usando medidas estatísticas conhecidas e amplamente utilizadas na literatura. O terceiro objetivo, relacionado à implementação ampla do processo de seleção de atributos, foi atingido com a implementação do processo de seleção e as variações de representação citadas na literatura.

Os experimentos exploratórios realizados foram utilizados para identificar o modelo de processo real obtido a partir dos registros da plataforma. Esse experimento permitiu identificar que trata-se de um processo semi-estruturado e fazer a validação de comportamento dos STAs ao realizar a modificação dos atributos utilizados para sua construção, direcionando a pré-validação da hipótese. A criação das medidas de referência para avaliação foi realizada com o experimento utilizando o conhecimento do especialista humano apoiado pela teoria do ITIL e apresentou resultados muito significativos relacionados à adequação do modelo ao processo, mas resultados pouco expressivos quanto à utilização na geração de estimativas de conclusão. Os experimentos utilizando a técnica de filtro foram ligeiramente melhores quanto à assertividade de predição mas apresentaram um efeito colateral de uma baixa adequação ao modelo de processo real. Os experimentos conduzidos com as buscas

heurísticas apresentaram uma assertividade muito superior (apenas 38,47% do obtido com o valor de referência) e uma capacidade de adequação ao processo significativa de 98,90%. Os experimentos com os algoritmos genéticos foram capazes de explorar combinações de atributos distintas das anteriores e apresentaram resultados intermediários quando comparados em relação à assertividade e à não adequação, sendo de 34,87% e 32,09% respectivamente, no primeiro resultado e 60,38% e 16,16% no segundo resultado avaliado. Esses valores demonstram que é um método factível para obtenção de resultados robustos, mas carece de uma maior exploração em sua implementação par obtê-los.

Diante dos resultados obtidos, a hipótese delineada nesse trabalho foi confirmada em sua totalidade, pois, a partir dos resultados obtidos com a execução dos experimentos comprovou-se que é possível construir, através dos procedimentos de seleção de atributos utilizando técnicas de filtro e invólucro, uma lista de atributos que permite a criação de sistemas de transição anotados que descrevem adequadamente o processo de gerenciamento de incidentes. Os STAs criados possuem alta capacidade de generalização e produzem estimativas de tempo para conclusão com acurácia superior à obtida com STAs construidos a partir de definições da literatura que orientam a construção de modelos de representação de processos.

6.1 Principais contribuições

A aplicação ao processo de real de tratamento de incidentes, consolidando os dados descritivos obtidos de uma plataforma amplamente utilizada, a $ServiceNow^{TM}$, trouxe uma possibilidade de avaliação pormenorizada desse tipo de processo. A estrutura criada torna possível que outra análises mais aprofundadas possam ser realizadas no futuro utilizando essa mesma plataforma.

O estudo da prática do ITIL, em termos de escolha de atributos por especialistas e sua comparação com os dados obtidos dos processos reais, demonstra que há efetivamente uma diferença entre os modelos de processos teóricos e práticos, ainda que utilizando padrões bem estabelecidos e amplamente difundidos na indústria.

O uso do método de invólucro forneceu uma abordagem capaz de selecionar um subconjunto de atributos que suportou uma melhoria significativa na acurácia do STA usado como modelo de predição do tempo de execução dos incidentes quando comparado ao

método de filtro e ao conhecimento especializado. A utilização dessa abordagem viabilizou a avaliação de variações nos parâmetros de abstração, como o horizonte máximo utilizado na construção do modelo e os diferentes tipos de representações de estados. Foi possível demonstrar que têm uma grande influência nos resultados finais do modelo de predição. Esta abordagem tem potencial para ser usado como um passo útil de pré-processamento antes da aplicação de outros métodos de predição que podem ser complementares ao STA utilizado neste trabalho.

A incorporação do procedimento de buscas com o algoritmo genético representou uma abordagem alternativa promissora e complementar à forma de exploração do espaço de buscas com os algoritmos heurísticos tradicionais.

6.2 Limitações do trabalho

Os dados de log da solução na plataforma $ServiceNow^{TM}$ são apresentados no formato de registro das atualizações realizadas na ferramenta em uma interface gráfica apenas. Poderia ser aprimorado para um processo geração das informações do log de eventos enriquecido mais preciso e de maneira a facilitar o processamento na mineração de processos. Outro ponto identificado foram transições diretamente para o status (campo incident_state) resolvido e nesse caso há uma distorção no tempo de registro, tratamento e conclusão. Essa configuração da plataforma, faz com que parte do tempo utilizado no tratamento do incidente não seja registrado no log de auditoria e portanto não há possibilidade de identificar as atividades realizadas no processo de construção dos STAs.

A estrutura da ProM, não possui uma arquitetura que permita a execução dos aplicativos construidos sob a forma de serviços ou subrotinas, limitando a utilização à construção de uma interface visual ou à criação de código que possa contornar essas limitações. A utilização da interface gráfica trouxe outras consequências, ao realizar a mineração para fazer a descoberta do modelo de transição de estados com mais de 1 atributo, a ProM apresentou um erro indicando numero excessivo de estados. Esse fato fez que houvesse a necessidade de buscar uma alternativa para dar sequência à execução dos experimentos. A decisão foi a construção rotinas específicas para geração dos STAs e dos algoritmos de buscas.

A utilização da linguagem R, embora paralelizada, foi um limitador para execução de alguns dos procedimentos de busca com um número maior de instâncias de processo (a partir de 8.000 registros) e o procedimento dos algoritmos genéticos.

Embora a utilização do log referente a um processo real seja relevante, há necessidade de executar os experimentos em outras implementações reais de modo que seja possível fazer um comparativo do comportamento observado nesse caso com outros processos reais e seus respectivos indicadores, criando então parâmetros de referência para esse processo.

Outra limitação enfrentada foi a falta de um ambiente padronizado para realizar o registro das informações e a comparação com processos similares e seus estudos respectivos. A ProM apresenta um número elevado de plug-ins, alguns com interoperabilidade, mas não possui uma estrutura para compartilhamento dos resultados dos experimentos e seus conjuntos de dados.

Apesar dos resultados obtidos terem atingido os objetivos propostos nesse trabalho, a otimização realizada com a busca utilizando apenas o MAPE e tendo a questão do NF como um item adicional, fez com que alguns dos bons resultados obtidos tivessem sua aplicabilidade limitada.

6.3 Trabalhos futuros

Superar as limitações são possibilidades, mas, à parte delas, há alguns estudos que podem ser interessantes como próximos passos na pesquisa. O estudo da influência de "outliers" ao longo do processo (desempenho de busca, predição e adequação ao processo), uma vez que os resultados obtidos nas experiências apresentaram algum grau de variação e mostrou-se sensível quando comparado com diferentes horizontes. Outro indício são as estatísticas apresentadas na tabela 9 que apontam um último quartil com variações significativas.

Um item a ser estudado diz respeito à utilização de penalidades na função de avaliação de qualidade do modelo. Dessa forma, a busca pode ser direcionada para obtenção de um valor que além de priorizar o MAPE e obtenha um valor de NF otimizado e consequentemente um modelo de processo mais adequado. Essa avaliação, aliada à utilização de modelos de regressão mais sofisticados nas funções de predição (ao invés

de apenas estatísticas simples) podem contribuir para melhorar a performance geral dos STAs.

A expansão do método para utilizar atributos de outras entidades relacionadas ao incidente, bem como a utilização dos atributos não estruturados (textuais) também é um item que deve ser investigado, pois, tal como apresentado na figura 4, os processos do ITIL são inter-relacionados e outras informações podem ser relevantes para a construção dos STAs. Além da seleção de atributos, pode-se avançar na utilização desses atributos para extração de características que possam gerar uma correlação mais precisa e possam produzir modelos de melhor acurária, mantendo o baixo sobreajuste já obtido nos experimentos atuais. Há outros fatores contextuais que influenciam a precisão dos modelos e que podem ser incorporados, como uma característica que aponte o número de casos tratados por um recurso de modo que a utilização de capacidade possa ser considerada na construção do modelo. A avaliação do paralelismo de atividades em cada recurso e o calendário de finais de semana, feriados e férias são outros pontos a serem avaliados.

Dado o cenário do log de eventos enriquecido ter uma ordenação temporal, a exploração de outros algoritmos de indução, tais como redes neurais recorrentes, que tenham a capacidade de manter uma memória referente à essa evolução é um item com potencial para ser explorado e utilizado como alternativa à utilização dos STAs.

Todos esses itens propostos podem se beneficiar da estrutura criada para este trabalho – tanto no código fonte construido em linguagem R quanto ao log de eventos enriquecido – e utilizá-las como fonte de informação para dar sequencia às atividades de maneira a complementar o estudo realizado e evoluir com as novas atividade de pesquisa.

Referências[1]

AALST, W. M. P. van der. Process Mining - Discovery, Conformance and Enhancement of Business Processes. 1. ed. [S.l.]: Springer, 2011. Citado 6 vezes nas páginas 25, 27, 28, 29, 52 e 83.

AALST, W. M. P. van der et al. Process mining: A two-step approach to balance between underfitting and overfitting. Software & Syst. Modeling, v. 9, n. 1, Nov 2008. ISSN 1619-1374. Disponível em: ⟨https://doi.org/10.1007/s10270-008-0106-z⟩. Citado na página 35.

AALST, W. 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 12 vezes nas páginas 20, 21, 22, 23, 30, 38, 39, 54, 56, 88, 94 e 100.

ABBACI, K. et al. A cooperative answering approach to fuzzy preferences queries in service discovery. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v. 7022 LNAI, p. 318–329, 2011. Citado na página 53.

ARMSTRONG, J.; COLLOPY, F. Error measures for generalizing about forecasting methods: Empirical comparisons. Int. J. of Forecasting, v. 8, n. 1, p. 69 – 80, 1992. ISSN 0169-2070. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/016920709290008W⟩. Citado 2 vezes nas páginas 56 e 72.

BAUTISTA, A. et al. Process mining in information technology incident management: A case study at volvo belgium. CEUR Workshop Proceedings, CEUR-WS, v. 1052, p. –, 2013. Citado na página 51.

BEKKERMAN, R. et al. Distributional word clusters vs. words for text categorization. Journal of Machine Learning Research, JMLR.org, v. 3, p. 1183–1208, mar. 2003. ISSN 1532-4435. Citado na página 42.

BERTI, A. Improving process mining prediction results in processes that change over time. In: Data Analytics 2016: 5th Int. Conf. on Data Analytics. [S.l.: s.n.], 2016. p. 37–42. Citado na página 55.

BEVACQUA, A. et al. A data-driven prediction framework for analyzing and monitoring business process performances. Lecture Notes in Business Information Processing, Springer Verlag, v. 190, p. 100–117, 2014. Citado na página 52.

BLUM, A. L.; LANGLEY, P. Selection of relevant features and examples in machine learning. Artificial Intell., v. 97, n. 1, p. 245–271, 1997. ISSN 0004-3702. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/S0004370297000635⟩. Citado 2 vezes nas páginas 41 e 76.

CARUANA, R.; SA, V. R. de. Benefitting from the variables that variable selection discards. Journal of Machine Learning Research, JMLR.org, v. 3, p. 1245–1264, mar. 2003. ISSN 1532-4435. Citado na página 42.

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

CICCIO, C. D.; MARRELLA, A.; RUSSO, A. Knowledge-intensive processes: Characteristics, requirements and analysis of contemporary approaches. J. on D. Sem., v. 4, n. 1, p. 29–57, Mar 2015. ISSN 1861-2040. Citado 2 vezes nas páginas 25 e 26.

COHEN, J. Eta-squared and partial eta-squared in fixed factor anova designs. Sage Publications, Elsevier B.V., v. 33, p. 107–112, 1973. Citado na página 43.

DOMINGOS, P. The role of occam's razor in knowledge discovery. Data Mining and Knowledge Discovery, v. 3, n. 4, p. 409–425, Dec 1999. ISSN 1573-756X. Disponível em: ⟨https://doi.org/10.1023/A:1009868929893⟩. Citado na página 79.

DUDOK, E.; BRAND, P. V. D. Bpic'13: Mining an incident management process. CEUR Workshop Proceedings, CEUR-WS, v. 1052, p. –, 2013. Citado na página 51.

EVERMANN, J.; REHSE, J.-R.; FETTKE, P. Predicting process behaviour using deep learning. Decision Support Systems, v. 100, p. 129 – 140, 2017. ISSN 0167-9236. Smart Business Process Management. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/S0167923617300635⟩. Citado na página 56.

FLUXICON. Fluxicon Disco tool homepage. 2018. Disponível em: ⟨www.fluxicon.com/disco⟩. Citado 2 vezes nas páginas 51 e 83.

FOLINO, F.; GUARASCIO, M.; PONTIERI, L. Discovering context-aware models for predicting business process performances. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v. 7565 LNCS, n. PART 1, p. 287–304, 2012. Citado na página 53.

GEYER-SCHULZ, A. Fuzzy rule-based expert systems and genetic machine learning. 1st. ed. [S.l.]: Springer-Verlag Berlin Heidelberg, 1996. ISBN 9783790818932. Citado 2 vezes nas páginas 46 e 47.

GIBBONS, J. D.; CHAKRABORTI, S. Nonparametric statistical inference. In: _____. International Encyclopedia of Statistical Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 977–979. ISBN 978-3-642-04898-2. Disponível em: ⟨https://doi.org/10.1007/978-3-642-04898-2_420⟩. Citado na página 74.

GOLDBERG, D. E. Genetic algorithms in search, optimization and machine learning. 1st. ed. [S.l.]: Addison-Wesley, 1989. ISBN 0201157675, 9780201157673. Citado 2 vezes nas páginas 46 e 47.

GRACZYK, M. et al. Nonparametric statistical analysis of machine learning algorithms for regression problems. In: SETCHI, R. et al. (Ed.). Knowledge-Based and Intelligent Information and Engineering Systems. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 111–120. ISBN 978-3-642-15387-7. Citado na página 74.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. J. of Machine Learning Research, JMLR.org, v. 3, p. 1157–1182, mar. 2003. ISSN 1532-4435. Disponível em: ⟨http://dl.acm.org/citation.cfm?id=944919.944968⟩. Citado 4 vezes nas páginas 41, 42, 75 e 76.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The Elements of Statistical Learning. 2nd. ed. New York, NJ, USA: Springer-Verlag, 2009. ISBN 9780387848570. Citado na página 42.

HINKKA, M. et al. Structural feature selection for event logs. In: TENIENTE, E.; WEIDLICH, M. (Ed.). Business Process Management Workshops. Cham: Springer International Publishing, 2018. p. 20–35. ISBN 978-3-319-74030-0. Citado na página 55.

HOLLAND, J. Adaptation in natural and artificial systems. 1st. ed. Upper Saddle River, NJ, USA: University of Michigan Press, 1975. ISBN 0262082136. Citado 2 vezes nas páginas 44 e 77.

INTERNATIONAL itSMF. itSMF 2013 Global Survey On IT Service Management. 2013. Disponível em: ⟨www.itil.co.il/wp-content/uploads/2015/02/itSMF-2013-Service-Management-Survey-Report.pdf⟩. Citado na página 18.

ITSMF. Global Survey on IT Service Management. 2013. The IT Service Management Forum. Http://www.itil.co.il. Citado na página 76.

JENSEN, D. D.; COHEN, P. R. Multiple comparisons in induction algorithms. Machine Learning, v. 38, n. 3, p. 309–338, Mar 2000. ISSN 1573-0565. Disponível em: ⟨https://doi.org/10.1023/A:1007631014630⟩. Citado na página 79.

KENNEDY, J. J. The eta coefficient in complex anova designs. Educational and Psychological Measurement, Sage Publications, v. 30, p. 885–889, 1970. Citado na página 43.

KERLINGER, F. N. Foundations of behavioral research. 1. ed. [S.l.]: New York: Holt, Rinehart and Winston, 1964. Citado na página 43.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. Artificial Intell., v. 97, n. 1, p. 273–324, 1997. ISSN 0004-3702. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/S000437029700043X⟩. Citado 8 vezes nas páginas 40, 41, 42, 44, 76, 77, 78 e 114.

KOZA, J. R. Genetic Programming. 1st. ed. [S.l.]: The MIT Press, 1996. ISBN 0262111705. Citado 3 vezes nas páginas 45, 49 e 77.

LAMINE, E. et al. Improving the management of an emergency call service by combining process mining and discrete event simulation approaches. IFIP Advances in Information and Communication Technology, Springer New York LLC, v. 463, p. 535–546, 2015. Citado na página 51.

LIU, X. et al. A novel statistical time-series pattern based interval forecasting strategy for activity durations in workflow systems. Journal of Systems and Software, v. 84, n. 3, p. 354–376, 2011. Citado na página 53.

MARRONE, M. et al. It service management: A cross-national study of itil adoption. Communications of the Association for Information Systems, Association for Information Systems, v. 34, n. 1, p. 865–892, 2014. Citado 3 vezes nas páginas 18, 22 e 50.

MICHALEWICZ, Z. Evolutionary Programming and Genetic Programming. In: Genetic Algorithms + Data Structures = Evolution Programs. 3rd. ed. [S.l.]: Springer-Verlag Berlin Heidelberg, 1996. ISBN 9783540606765. Citado 2 vezes nas páginas 45 e 77.

MITCHELL, M. An Introduction to Genetic Algorithms. 1st. ed. [S.l.]: The MIT Press, 1996. ISBN 9780262133166. Citado na página 45.

MYTTENAERE, A. de et al. Mean absolute percentage error for regression models. Neurocomputing, Elsevier B.V., v. 192, p. 38–48, 2016. ISSN 0925-2312. Disponível em: ⟨http://dx.doi.org/10.1016/j.neucom.2015.12.114⟩. Citado na página 72.

MüLLER, R. et al. Service discovery from observed behavior while guaranteeing deadlock freedom in collaborations. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v. 8274 LNCS, p. 358–373, 2013. Citado na página 52.

NASERI, M.; LUDWIG, S. Automatic service composition using pomdp and provenance data. Proceedings of the 2013 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013, p. 246–253, 2013. Citado na página 52.

POLATO, M. et al. Data-aware remaining time prediction of business process instances. In: 2014 Int. Joint Conf. on Neural Networks. [S.l.: s.n.], 2014. p. 816–823. ISSN 2161-4393. Citado na página 52.

QUINLAN, J. R.; CAMERON-JONES, R. M. Oversearching and layered search in empirical learning. IJCAI95 Proceedings of the 14th international joint conference on Artificial intelligence, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA ©1995, v. 2, n. 1, p. 1019–1024, 1997. Citado na página 79.

RICHARDSON, J. T. Eta squared and partial eta squared as measures of effect size in educational research. Educational Research Review, v. 6, n. 2, p. 135–147, 2011. ISSN 1747-938X. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/S1747938X11000029⟩. Citado 2 vezes nas páginas 76 e 101.

ROGGE-SOLTI, A.; VANA, L.; MENDLING, J. Time series petri net models - enrichment and prediction. In: Proc. of the 5th Int. Symp. on Data-driven Process Discovery and Analysis (SIMPDA 2015). [S.l.: s.n.], 2015. p. 109–123. Citado na página 54.

ROGGE-SOLTI, A.; WESKE, M. Prediction of business process durations using non-markovian stochastic petri nets. Inf. Syst., v. 54, n. Supplement C, p. 1 – 14, 2015. ISSN 0306-4379. Disponível em: ⟨http://www.sciencedirect.com/science/article/pii/S0306437915000642⟩. Citado na página 54.

ROSSO-PELAYO, D. et al. Business process mining and rules detection for unstructured information. Proceedings of Special Session - 9th Mexican International Conference on Artificial Intelligence: Advances in Artificial Intelligence and Applications, MICAI 2010, p. 81–85, 2010. Citado na página 54.

RUSSELL, S.; NORVIG, P. Artificial Intelligence: A Modern Approach. 3rd. ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2009. ISBN 0136042597, 9780136042594. Citado na página 77.

SPIEGEL, P. V. D.; DIELTJENS, L.; BLEVI, L. Bpi challenge 2013 - applied process mining techniques for incident and problem management. CEUR Workshop Proceedings, CEUR-WS, v. 1052, p. –, 2013. Citado na página 51.

TAX, N. et al. Predictive business process monitoring with lstm neural networks. In: DUBOIS, E.; POHL, K. (Ed.). Advanced Information Systems Engineering. Cham:

Springer International Publishing, 2017. p. 477–492. ISBN 978-3-319-59536-8. Citado na página 56.

VERBEEK, H. M. W. et al. Xes, xesame, and prom 6. In: SOFFER, P.; PROPER, E. (Ed.). Information Systems Evolution. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 60–75. ISBN 978-3-642-17722-4. Citado 3 vezes nas páginas 53, 83 e 88.

WEERDT, J. D. et al. Leveraging process discovery with trace clustering and text mining for intelligent analysis of incident management processes. 2012 IEEE Congress on Evolutionary Computation, CEC 2012, p. –, 2012. Citado na página 53.

WESTON, J. et al. Use of the zero-norm with linear models and kernel methods. Journal of Machine Learning Research, JMLR.org, v. 3, p. 1439–1461, mar. 2003. ISSN 1532-4435. Citado na página 42.

Glossário

atributo descritivo Atributo que serve para descrever alguma característica de uma entidade ou processo.

incidente Qualquer situação não prevista que cause impacto (degradação ou indisponibilidade) a um serviço de tecnologia.

instância do processo Uma instância de processo é uma ocorrência específica de um processo criada seguindo a definição formal para aquele processo. Por exemplo, o incidente número INC0001 é uma instância do processo de gerenciamento de incidentes.

log de auditoria Um log de auditoria é um registro cronológico, relevante para a segurança, composto por um conjunto de informações que fornecem evidências da seqüência de atividades que afetaram, a qualquer momento, um registro em uma entidade específica.

log de eventos Um log de eventos é um conjunto de instâncias do processo e seus eventos associados..

Apêndices

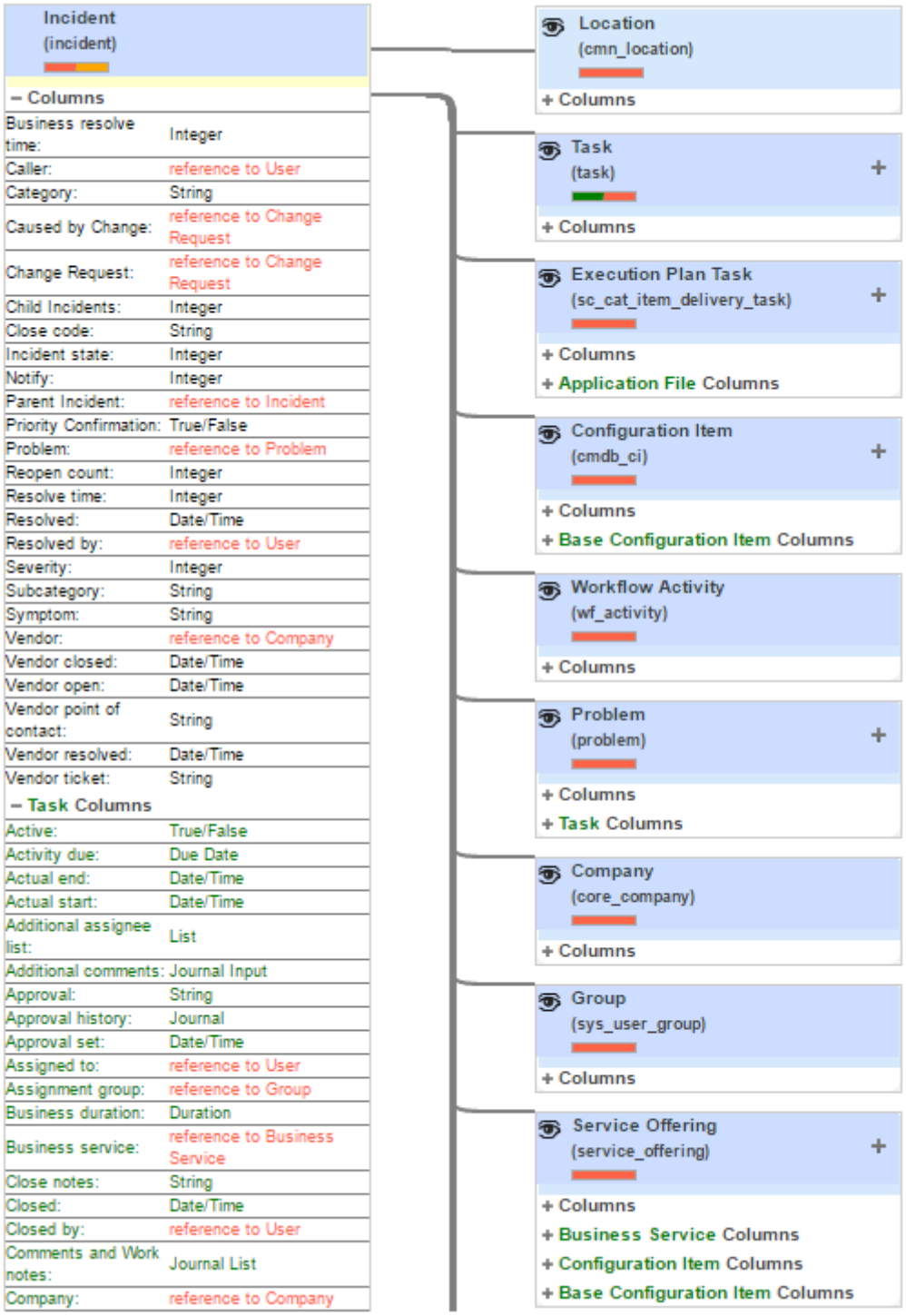
Apêndice A – Modelo de dados da relação *incident*

Nas figuras 12 e 13 apresenta-se o modelo de dados referente à relação *incident*. Nele são observados todos os atributos que descrevem um incidente. Neste conjunto de atributos, parte deles são especificamente do incidente (em cor verde) e parte deles (em cor preta) são herdados de uma outra entidade no sistema (*task*). A figura fornece uma noção da granularidade de informação que são armazenados no sistema de gerenciamento do incidente.

As informações a utilizadas neste trabalho foram escolhidas a partir de uma análise empírica do seu significado em relação ao gerenciamento de incidentes e aos objetivos pretendidos. A lista de atributos que trazem tal conjunto de informações são:

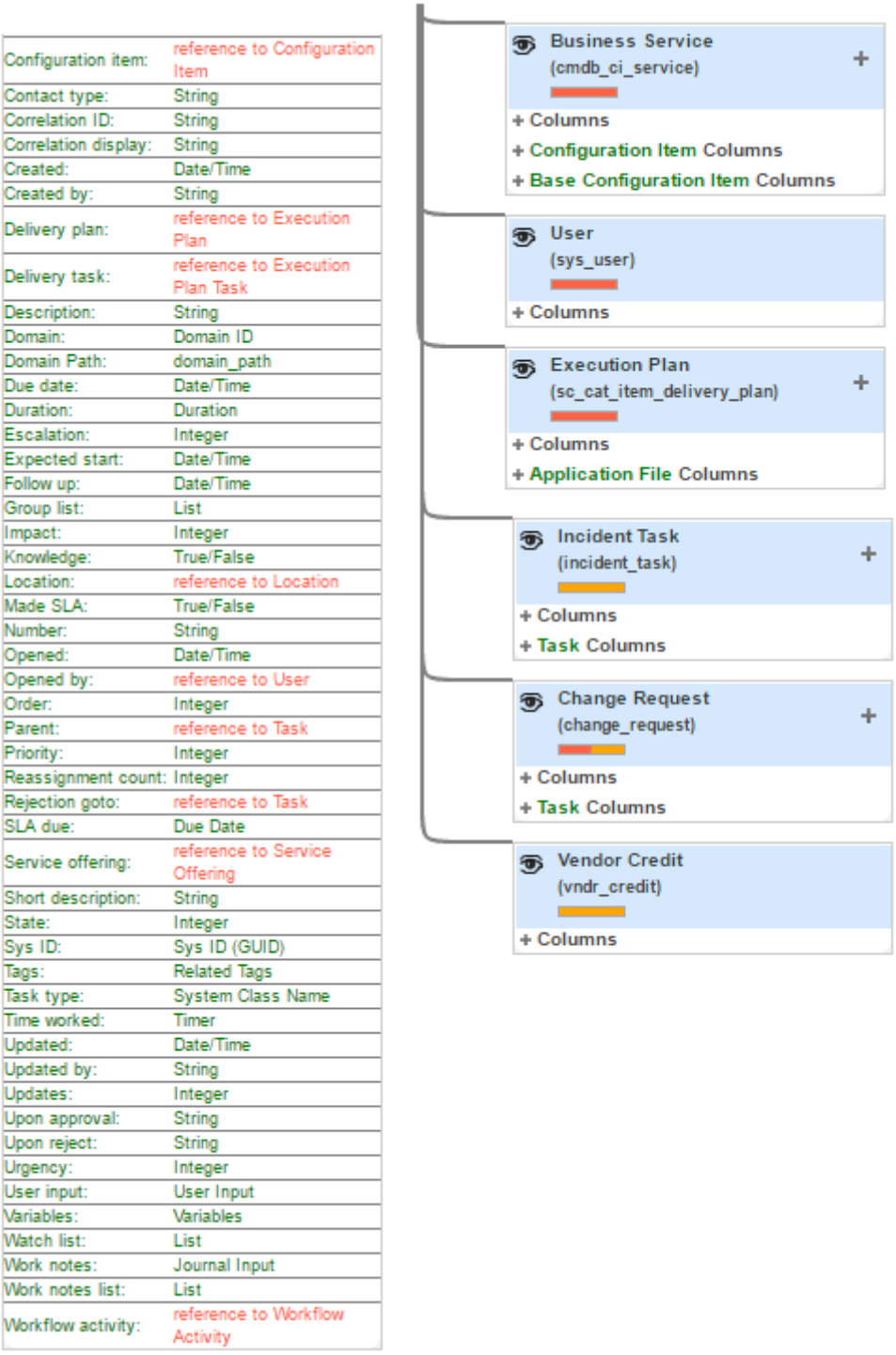
- *Caller*: Trata-se de uma referência para a relação *user*, usada de forma que se tenha a informação sobre quem reportou o incidente.
- *Category*: Atributo cujo domínio é uma lista de opções. É utilizado para fazer a categorização, em primeiro nível, do incidente. Por exemplo, o incidente diz respeito a um serviço de software, um equipamento (hardware), etc. Atualmente a lista possui trinta e duas (32) opções.
- *Subcategory*: Atributo cujo domínio é uma lista de opções. É utilizado para fazer a categorização, em segundo nível, do incidente. Por exemplo, o incidente está relacionado a uma subcategoria de hardware (um servidor, um desktop). Atualmente a lista possui duzentos e quarenta (240) opções.
- *Symptom*: Atributo cujo domínio é uma lista de opções. É utilizado para informar o que está sendo percebido pelos usuários. Por exemplo: o serviço está lento, o hardware está inacessível. Atualmente a lista possui quinhentas e uma (501) opções.
- *Caused by Change*: Se o incidente foi causado por uma requisição de mudanças, esse atributo será preenchido na etapa de investigação e diagnóstico e a mudança relacionada será apontada no conteúdo do atributo.
- *Change Request*: Esse atributo será preenchido caso a correção do incidente tenha gerado a abertura de uma requisição de mudanças. Seu conteúdo diz respeito à indicação de tal requisição.
- *Incident state*: Atributo cujo domínio é uma lista de opções. Diz respeito a um controle sobre a transição de estados do incidente durante o processo de gerenciamento do

Figura 12 – Modelo de dados (Parte 1): relação *incident*



Fonte: *ServiceNow*^{TM}, 2018

Figura 13 – Modelo de dados (Parte 2): relação *incident*



Configuration item:	reference to Configuration Item
Contact type:	String
Correlation ID:	String
Correlation display:	String
Created:	Date/Time
Created by:	String
Delivery plan:	reference to Execution Plan
Delivery task:	reference to Execution Plan Task
Description:	String
Domain:	Domain ID
Domain Path:	domain_path
Due date:	Date/Time
Duration:	Duration
Escalation:	Integer
Expected start:	Date/Time
Follow up:	Date/Time
Group list:	List
Impact:	Integer
Knowledge:	True/False
Location:	reference to Location
Made SLA:	True/False
Number:	String
Opened:	Date/Time
Opened by:	reference to User
Order:	Integer
Parent:	reference to Task
Priority:	Integer
Reassignment count:	Integer
Rejection goto:	reference to Task
SLA due:	Due Date
Service offering:	reference to Service Offering
Short description:	String
State:	Integer
Sys ID:	Sys ID (GUID)
Tags:	Related Tags
Task type:	System Class Name
Time worked:	Timer
Updated:	Date/Time
Updated by:	String
Updates:	Integer
Upon approval:	String
Upon reject:	String
Urgency:	Integer
User input:	User Input
Variables:	Variables
Watch list:	List
Work notes:	Journal Input
Work notes list:	List
Workflow activity:	reference to Workflow Activity

Fonte: *ServiceNow*^{TM}, 2018

mesmo. São valores numéricos que possuem o significado definido de acordo com a tabela 20.

Tabela 20 – Correspondência entre valor e significado do estado no incidente (atributo *Incident state*)

Valor	Significado
1	Novo
2	Ativo
3	Aguardando Problema
4	Aguardando Informações do Usuário
5	Aguardando Evidencias
6	Resolvido
7	Encerrado
-2	Aguardando Fornecedor

Fonte: $ServiceNow^{TM}$, 2018

- *Priority Confirmation*: Valor verdadeiro/falso para confirmação da prioridade em caso de incidentes prioridade 1(mais alta prioridade).
- *Problem*: Esse atributo será preenchido caso a correção do incidente tenha gerado a abertura de um registro de problema para ser tratado. Nesse caso, há a indicação de que a solução será aplicada quando tratada no processo de gestão de problemas.
- *Reopen Count*: Atributo do tipo contador. Indica a quantidade de "reaberturas" e é usado quando o solicitante (*Caller*) reporta que a solução não reestabeleceu corretamente o serviço. Como consequência de seu uso, o valor do campo *estado* passa de 6 (resolvido) para 2 (ativo).
- *Resolved*: Data de resolução do incidente.
- *Resolved By*: Trata-se de uma referência para a relação *user*, usada de forma que se tenha a informação sobre o analista que resolveu do incidente.
- *Vendor*: Trata-se de uma referência para a relação *company*, usada caso haja a necessidade de acionamento de um fornecedor.
- *Vendor ticket*: Número do chamado aberto no fornecedor.
- *Vendor point of contact*: Nome da pessoa de contato no fornecedor.
- *Vendor Open/Resolved*: Atributo que informa as respectivas datas de abertura e encerramento do chamado no fornecedor.
- *Assignment group*: Trata-se de uma referência para a relação *Group*, usada para informar qual o grupo de suporte está responsável pelo incidente.

- *Assigned to*: Trata-se de uma referência para a relação *User*, usada para informar qual o analista responsável pelo tratamento do incidente em um determinado instante.
- *Closed*: Registro com a data completa de quando foi encerrado o incidente.
- *Closed by*: Trata-se de uma referência para a relação *User*, usada para informal qual o usuário responsável pelo encerramento do incidente. Pode ser o caller ou o usuário de sistema caso o encerramento seja feito pelo sistema de forma automática após cinco (5) dias da resolução.
- *Close notes*: Descrição final (textual) da solução e dados diversos sobre o encerramento do tratamento do incidente.
- *Close code*: Lista de opções com os códigos de encerramento do incidente;
- *Comments and Work notes*: Campo em formato de lista que armazena todos os comentários e informações inseridas no transcorrer do ciclo de vida do incidente.
- *Configuration item*: Trata-se de uma referência para a relação *Configuration Item*, usada para informar qual item de configuração foi afetado pelo incidente.
- *Contact type*: Atributo cujo domínio é uma lista de opções. Diz respeito às opções sobre a forma de contato para registro do incidente (telefone, portal, e-mail, monitoração, pessoalmente);
- *Created*: Data completa de criação do incidente.
- *Created by*: Trata-se de uma referência para a relação *User*, usada para informar o usuário que fez o registro do incidente.
- *Description*: Descrição textual informada na abertura do incidente.
- *Impact*: Atributo cujo domínio é uma lista de opções. Representa o impacto causado pelo incidente (1 - alto; 2 - médio; 3 - baixo).
- *Knowledge*: Atributo de valor Verdadeiro/Falso para indicar se foi encontrada informação na base de conhecimento para solucionar o incidente.
- *Location*: Trata-se de uma referência para a relação *Location*. Indica o local afetado pelo incidente. Usualmente é o local do *Caller*;
- *Made SLA*: Atributo de valor Verdadeiro/Falso para indicar se o incidente foi resolvido dentro do tempo alvo de atendimento (SLA).
- *Number*: Atributo identificador único do incidente.
- *Opened*: Data de abertura do incidente.
- *Opened by*: Trata-se de uma referência para a relação a relação *User*, usada para informar o usuário quer fez o registro da abertura do incidente.

- *Priority*: Atributo que indica a prioridade do incidente, com valores de um (1) a cinco (5) sendo que os valores menores representam prioridade mais alta. O tempo alvo de resolução é direcionado pela prioridade. Esse campo é calculado a partir de uma matriz obtida com a definição dos campo *Impact* e *Urgency*;
- *Reassignment count*: Atributo do tipo contador. Indica o número de vezes que o incidente teve seu tratamento transferido de grupo resolvedor e/ou de analista responsável.
- *SLA due*: Data esperada de resolução de acordo com a definição de tempo alvo associado.
- *Short description*: Titulo da descrição informada no momento de abertura do incidente.
- *Updated*: Data completa da última atualização do registro de incidente.
- *Updated by*: Trata-se de uma referência para a relação *User*, usada para informar o usuário que fez a ultima atualização no registro.
- *Updates*: Atributo do tipo contador. Indica o número de atualizações realizadas no registro de incidente.
- *Urgency*: Atributo cujo domínio é uma lista de opções. Representa a urgência para tratamento do incidente (1 - alta; 2 - média; 3 - baixa).

Apêndice B – Log de auditoria

Na figura 14 estão descritos os atributos que compõem a relação *sys_audit* que armazena todos os registros de auditoria da plataforma *ServiceNow*^{TM}. Os atributos que constituem um registro de log são:

Figura 14 – Modelo de dados: relação *sys_audit*.

Sys Audit (sys_audit)	
– Columns	
Created:	Date/Time
Created by:	String
Document Key:	Char
Field Name:	Short Field Name
New value:	String
Old value:	String
Reason:	String
Record internal checkpoint:	String
Sys ID:	Sys ID (GUID)
Table Name:	Short Table Name
Update count:	Integer
User:	String

Fonte: *ServiceNow*^{TM}, 2018

- *Created*: Data completa de criação do registro de log.
- *Created by*: Trata-se de uma referência para a relação a relação *User*, usada par informar qual usuário fez a atualização na relação de origem.
- *Document Key*: Trata-se de uma referência para a relação sob auditoria. No caso deste projeto, o interesse é a auditoria sobre a relação *incident*.
- *Field Name*: Nome no atributo atualizado na relação sob auditoria.
- *New value*: Valor atribuído ao atributo atualizado na relação sob auditoria.
- *Old Value*: Valor anterior do atributo atualizado na relação sob auditoria.
- *Reason*: Campo não utilizado que está sempre vazio.
- *Record internal checkpoint*: Atributo identificador para o conjunto de atualizações.
- *Sys ID*: Atributo identificador para o registro de log.
- *Table name*: Nome da relação sob auditoria (*incident* neste caso).
- *Update count*: Atributo do tipo contador que indica a qual sequência de atualização se refere o registro de log.
- *User*: Referência para a relação *User*, usada para informar qual usuário que fez a atualização na relação sob auditoria.

Apêndice C – Atributos de incidentes agrupados e seus domínios

- **Atributos de controle:** *Number*: identificador único do incidente que tem o mesmo número que o total de casos; *incident state*: atributo 8 níveis distintos que faz o controle das transições do processo de gerenciamento de incidentes da abertura ao encerramento do caso; *Active*: atributo booleano que armazena se o registro está ativo ou inativo (estados fechado/canceledo); *Approval*: atributo booleano que armazena se houve solicitação de aprovação para o registro; *Reassignment count*: número de vezes que o incidente foi transferido entre os grupos e os analistas de suporte; *Reopen count*: contador do número de vezes que a solução apresentada para o caso foi rejeitada pelo solicitante; *Made SLA*: atributo booleano que indica se o incidente excedeu o tempo limite de SLA ou não; *SLA due*: data e hora esperada para resolução do incidente.
- **Atributos de identificação e classificação:** *Caller*: identificador do usuário afetado pela indisponibilidade ou degradação (5642 valores distintos); *Created by*: identificador do usuário que fez o registro do incidente no sistema (234 valores distintos); *Created*: data e hora da criação do incidente; *Opened by*: identificador do usuário que fez a comunicação do incidente (541 valores distintos); *Opened*: data e hora da abertura do incidente; *Contact type*: atributo categórico com 8 valores possíveis que informa qual o meio de contato utilizado para registro do incidente; *Location*: identificador do local afetado pelo incidente (249 valores distintos); *Category*: atributo categórico que faz a descrição do primeiro nível de serviço que está sendo afetado (63 valores distintos); *Subcategory*: atributo categórico que faz a descrição do segundo nível de serviço que está sendo afetado e está relacionado como dependência ao primeiro nível (305 valores distintos); *Symptom*: descrição de qual a percepção do usuário sobre a disponibilidade do serviço (609 valores distintos); *Configuration item*: identificador que faz referência a uma entidade homónima e utilizado para informar o item que está sendo afetado (53 valores distintos). Esta coluna é opcional; *Impact*: descrição do impacto causado pelo incidente. Values are: 1–High; 2–Medium; 3–Low; *Urgency*: descrição da urgência requerida pelo usuário solicitante para resolução do incidente . Valores possíveis: 1–Alta; 2–Média; 3–Baixa; *Priority*: prioridade

calculada pelo sistema baseada nos atributos *Impact* e *Urgency* (5 valores distintos); *Severity*: descrição da severidade do incidente (5 valores distintos).

- **Suporte, Diagnóstico e demais atributos:** *Assignment group*: identificador referenciando a relação *Group*, descrevendo o grupo de suporte encarregado do incidente (82 valores distintos); *Assigned to*: identificador do usuário que está responsável pelo incidente (253 valores distintos); *Updated by*: identificador do usuário que fez a atualização do registro e gerou o log de registro atual (996 valores distintos); *Updated*: data e hora de atualização do registro; *Knowledge*: atributo booleano se foi utilizado algum procedimento da base de conhecimento para fazer a resolução do incidente; *Priority Confirmation*: atributo booleano indicando se o campo *priority* foi revalidado; *Notify*: atributo categórico indicando que foram geradas notificações para o incidente (3 valores distintos); *Problem*: identificador que faz referência a uma entidade homónima descrevendo o registro do processo de gerenciamento de problemas associado a este incidente (273 valores distintos); *Change Request*: identificador que faz referência a uma entidade homónima descrevendo o registro do processo de gerenciamento de mudanças associado a este incidente (190 valores distintos); *Updates*: número de atualizações executadas no registro de incidente até o momento atual; *Vendor*: identificador que faz referência a uma entidade homónima descrevendo qual fornecedor está encarregado do incidente (6 valores distintos); *Resolved*: data e hora da resolução do incidente; *Closed*: data e hora do encerramento do incidente.