Aplicação de técnicas de Mineração de Processos

Alexia C. Scheffer, Ana Paula Mota Lima, Felipe Lobato, Ingrid G. G. Silva, Larissa Marques
Escola de Artes, Ciências e Humanidades - Universidade de São Paulo
Rua Arlindo Béttio, 1000 - Ermelino Matarazzo
03828-000 - São Paulo, SP
{ac.scheffer, ana_lima, felipediegolobatodasilva,
ingrid.gabrielly.silva, larissa4.silva}@usp.br

***Resumo*—Realizou-se uma analise no log de processos do grupo Rabobank utilizando técnicas de mineração de processos, a fim de encontrar a presença de padrões preditivos. Além de uma análise exploratória inicial onde foram aplicados testes com o algoritmo do ATS (Annotated Transition System), também foram realizados alguns estudos sobre o LSTM (Long Short-Term Memory).**

***Index Terms*—Mineração de Processos, Processo de Descoberta, Time Prediction**

I. Introdução

Empresas buscam ser competitivas no mercado, aumentar seus lucros e reconhecimento, ao mesmo tempo em que procuram ser mais eficientes, apresentar maior qualidade e diminuir custos. Para isso, ter eficiência também em seus processos é algo primordial, de maneira a evitar gargalos ou perdas de tempo nos processos de negócios. Para identificarmos problemas como esses, podem ser aplicadas técnicas de mineração de processos, bem como a utilização de ferramentas de mineração de dados e algoritmos para identificarmos padrões preditivos e assim buscar resolver alguns problemas que acontecem. É neste artigo que iremos apresentar as técnicas que utilizamos e os principais algoritmos aplicados para o Log de eventos Rabobank, do BPI Challenge [1]. O log de eventos contém informações relacionadas com processos da central de atendimento da empresa, incluindo o gerenciamento de interação, incidentes e mudanças, que serão úteis para descrever os atributos disponíveis também como o processo que está presente nele. Fizemos uma análise exploratória inicial para auxiliar a identificação de algum padrão no processo. -

II. Fundamentação Teórica

No mercado, na tentativa de as empresas melhorarem seus processos de negócios são apresentadas combinações de técnicas de mineração de dados [2] e algoritmos preditivos para auxilia-las. Devido a alta competitividade e a constante evolução do mercado, torna-se essencial que as empresas tenham um feedback continuo e perspicaz sobre como o processo de negócios está realmente sendo executado e sobre sua efetividade. No entanto, a empresa requer uma boa ferramenta de análise para obter informações precisas e relevantes sobre seus processos de negócios. Sendo assim, um dos principais impulsionadores por trás do desenvolvimento e aumento do uso de abordagens de mineração de processo é exatamente essa busca por saber como realmente os processos acontecem.

A. Information Technology Service Management e Information Technology Infrastructure Library

O Information Technology Service Management(ITSM) descreve o gerenciamento de uma organização com o seu respectivo processo de desenho, entrega, gerenciamento e melhoria dos serviços de tecnologia da informação fornecidos ao usuário final. Ele também pode ser implementado em relação ao service desk a fim de evitar um enorme tempo de espera do cliente no evento de um problema ou prevenir que outros problemas de TI surjam, e se caso ocorrer, saber como resolvê-los rapidamente. Além disso, o ITSM é centrado no cliente, focado na melhoria de ponta a ponta da entrega de serviço através de cinco componentes: Planejamento, Desenho, Entrega, Controle e Operação [3].

Para garantir essa melhoria de ponta em ponta, atualmente algumas empresas de TI introduziram processos ITIL (Biblioteca de Infraestrutura de Tecnologia da Informação) que é uma estrutura para para alinhar o ITSM nas empresas. O Information Technology Infrastructure Library(ITIL) é uma metodologia que cobre todas as fases do ciclo de vida do ITSM, assim pode-se dizer que os cinco componentes do ITSM são representados em 5 estágios.

1) **Estratégia de Serviço**: A meta é criar valor para o cliente transformando o serviço de TI em um ativo estratégico.
2) **Design de Serviço**: erve para mapear a disponibilidade de profissionais do setor e analisar suas habilidades tanto para otimizar serviços já existentes como para desenvolver novos.
3) **Transição de Serviço**: É quando ocorre a implementação do serviço e de sua validação com vários testes incluindo a cultura organizacional da empresa que também é envolvida, visto que essa transição, geralmente, afeta o ambiente corporativo por trazer mudanças.
4) **Operação de Serviço**: É a etapa que tem como meta assegurar que os serviços de TI sejam realizados e entregues com qualidade, e que seguirão as diretrizes determinadas pelo Acordo de Nível de Serviço (SLA).
5) **Melhoria contínua de Serviço**: Essa fase tem como objetivo acompanhar e revisar os serviços de TI. Ela é importante tanto para verificar eventuais falhas e, assim, corrigi-las, como também para definir melhorias.

A medida que o ITSM evoluía, o mesmo acontecia com as várias estruturas disponíveis para o ITSM, e isso inclui o ITIL que passou da versão 3 para a versão 4, que embora isso tenha acontecido não afetou o principal papel dessa estrutura, que é reduzir drasticamente o tempo necessário de implementação do ITSM e garantir com que este seja feito sob medida para o negócio, a fim de mitigar os riscos de negócios decorrentes de interrupções, situação que não acontecia com Rabobank Group ICT, quando o desafio foi feito em 2014.

B. Automatic Process Discovery

Como vimos, é muito importante ter uma visão precisa dos processos de uma empresa para que a elaboração da mineração desses processos e os resultados/analises feitas para melhoria sejam de fato corretos e o próximos da realidade.

Segundo o artigo "Minería de Procesos y Descubrimiento Automático de Procesos" [4] um objetivo de mineração de processos é descobrir os processos reais através da extração de conhecimento dos registros de eventos disponíveis nos sistemas de informação, como um log de eventos, e um dos três tipos de mineração de processos é justamente o de Descobrimento Automático de processos. Isso é feito utilizado um algorítimos desenvolvido para descobrimento automático de processos de negócios, que pode ter diversos enfoques, como o Business Process Model and Notation (BPMN). É usado para melhor visão dos processos que temos e identificação real deles. No desafio proposto o automatic process discovery é aplicado através do Software DISCO [5] com o objetivo de receber um log de eventos e retornar percepções o comportamento do processo.

C. Time Prediction

Tendo como referência teórica o artigo "Time Prediction based on process mining" [6], prevemos o tempo de conclusão de alguma instancia de processo, para isto, nós obtemos a sequencia de eventos executadas até o momento e usamos uma função de representação de estado para mapear essa sequencia em um estado de sistema de transição. Neste trabalho, utilizamos as informações sequenciais de um trace do log de eventos onde as atividades de cada caso são definidas pela abertura do chamado, marcada pela atividade denotada como "open", e seu status final, marcada pela atividade denotada como "closed". Com isso, uma previsão é feita, com base no tempo mínimo, máximo e médio de conclusão para as atividades de cada caso.

III. Trabalhos consultados

Um dos propósitos desse artigo é reproduzir técnicas adotadas em outros trabalhos. Isso pode ser considerado tanto como formas de normalização do conjunto de eventos quanto na aplicação de algoritmos apresentados. Assim, o presente artigo teve como referência teórica a tese "Tratamento do impacto de casos non-fitting em predição de tempo de resolução usando mineração de processos com múltiplos atributos" [7].

A dissertação aborda sobre o uso de técnicas de busca por similaridade para tratamento de casos non-fitting utilizando um método de predição de tempo de resolução de incidentes, baseando-se em mineração de processo. A partir de um método existente e um conjunto de dados de cenário real, houve a avaliação do impacto non-fitting sobre a assertividade do preditor em cenários com múltiplos atributos criados para geração do modelo. Por fim, a dissertação apresenta técnicas de busca por similaridade, normalização do conjunto de dados, geração de atributos de contexto a partir de temporais e adaptação da validação para uso da técnica holdout.

IV. Contextualização do problema e do ambiente

Os logs de eventos da Ferramenta de Gerenciamento de Serviços usado pelo Rabobank para gerenciar os processos ITIL que foram fornecidos no Desafio Business Process Intelligence 2014 é composto por quatro tabelas com eventos desde janeiro de 2013 até dezembro de 2014.

- **Registros de Interações (Detail_Interaction.csv)**
  - Possui 147.004 registros, cada um correspondendo a uma interação.
  - Cada registro contém um código de id, informações sobre a categoria de interação (incidente ou pedido de informação), quer tenha sido resolvido na primeira chamada ou não, impacto,urgência e prioridade da chamada, entre outras informações.
- **Registros de Incidentes(Detail_Incident.csv)**
  - Possui 46.606 registros, cada um correspondendo a um caso de incidente.
  - Todo registro possui um código de id, o código de id para a interação relacionada, se for apenas um, ou um #multivalue indicando que havia mais de uma interação relacionada.
  - O registro também tem informações sobre o item de configuração afetado, a EAP afetada e quais CI e WBS causaram a interrupção.
- **Atividades de Incidentes (Detail_Incident_Activity.csv)**
  - Possui 466.737 registros, vinculados a um id de incidente, com as atividades realizadas em cada caso.
- **Registros de Mudanças (Detail_Change.csv)**
  - Contém 30.275 registros relacionados as atividades realizadas em cada caso de mudança, incluindo informações sobre o item de configuração afetado, serviço de componente afetado, tipo de mudança e avaliação de risco, entre outros.
  - Os registros também inclui informações das seguintes datas: início planejado, término planejado, programado, início do tempo de inatividade, término do tempo de inatividade programado, início real, término real, término solicitado, Data, hora de abertura de registro de mudança, hora de fechamento de registro de mudança.

O processo demonstrado na Figura 1 apresenta:

1) Um cliente (cliente interno) que contata por telefone ou correio para o Service Desk para relatar interrupção em alguns serviços de TI.
2) Um agente do Service Desk recebe o contato do cliente e registra as informações sobre o serviço interrompido e

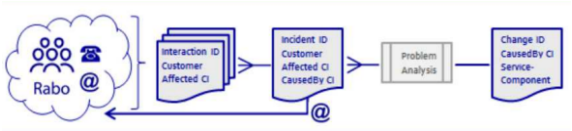

Figura 1. Processo macro

o item de configuração afetados. Essas informações são registradas em um log de eventos, incluindo usuário, carimbo de data / hora e código do agente.

3) As ligações que não são resolvidas diretamente são registradas como incidente e é atribuída a tarefa a uma equipe de especialistas para olvê-la.
4) O Service Desk pode receber várias chamadas relatando a mesma interrupção que será associada a um mesmo caso de incidente.
5) Uma determinada interrupção pode ocorrer novamente com mais frequência do que o normal, portanto, uma investigação de problema é iniciada, e se for o caso, uma solicitação de mudança será criada e o caso de mudança será atribuído a um especialista.

V. Arquitetura da solução

Todos os logs de eventos coletados foram combinados para ser um único arquivo de log de eventos. Após isso, esse log de eventos passou por algumas transformações utilizando a linguagem de programação Python:

- Foi realizado tratamentos distintos em atributos discretos e categóricos para valores nulos ou ausentes.
- Nomes das colunas foram padronizados, inclusive antes da combinação de todos os logs.
- Foi verificado os valores de colunas categóricas a fim de padroniza-las, removendo espaços extras e definindo todos como ”lowercase”.
- Houve formação de atributos de data e hora.

É possível verificar mais informações dessas transformações abaixo. Todos os códigos construídos para o pré processamento do log estão disponíveis no repositório do GitHub [8]. Depois de realizado todo o pré-processamento temos o dataset (log de eventos) contendo 466737 eventos, 46616 casos e 7 atributos:

- `incident_id` - Seria o nosso case id
- `km_number` - Seria o número do chamado no Service Desk
- `interaction_id` - Seria o id da interação com algum recurso
- `datestamp` - Data (%Y/%m/%d/%H:%M:%S) da ocorrência da atividade
- `incidentactivity_number` - Id da atividade
- `incidentactivity_type` - Seria a atividade que está ocorrendo naquele momento
- `assignment_group` - O recurso ocupado naquela atividade

Criamos alguns outros atributos que serão utilizadas para outras análises. Os últimos cinco são necessárias para o algoritmo ATS utilizado em um dos trabalhos consultados:

- `date` - Data (%Y/%m/%d) da ocorrência da atividade
- `first_activity` - Primeira atividade do trace;
- `last_activity` - Última atividade do trace;
- `first_day` - Data da primeira atividade do trace;
- `last_day` - Data da última atividade do trace;
- `case_resolution_days` - Intervalo entre `last_day` e `first_day` (Tempo de resolução em dias);
- `activity_resolution_secs` - Intervalo entre uma atividade e outra dentro do mesmo caso (Tempo de transição em segundos);
- `opened_at_stc` = variavel `first_day` em unix timestamp;
- `updated_at_stc` = variavel datestamp em unix timestamp;
- `closed_at_stc` = variável `last_day` em unix timestamp;
- `elapsed_stc` = tempo decorrido (`updated_at_stc` - `opened_at_stc`);
- `remaining_stc` = tempo restante (`closed_at_stc` - `updated_at_stc`);

Foi mencionado nos trabalhos consultados que, após alguns experimentos, ambos os algoritmos não demonstraram ter bom desempenho com logs com ruídos, loops e repetições de eventos, além disso, o mesmo foi recomendado em discussões de apoio com a ministrante da disciplina e monitores.

Assim, tendo isso em consideração, resolvemos fazer algumas análises do log para realizar cortes que minimizem o impacto no desempenho dos algoritmos, com apoio da ferramenta de mineração DISCO e de técnicas estatísticas utilizando a linguagem de programação R. Os cortes são descritos a seguir.

A. Corte 1 - Data das atividades

Para entender como está distribuída as atividades no tempo foi gerado dois gráficos para determinar um possível corte. No primeiro momento foi possível visualizar alguns períodos onde a massa de atividades é menor em comparação com outros, por exemplo, antes de Setembro/2013 e depois de Abril/2014. Essa análise foi facilmente confirmada no segundo momento com um gráfico de densidade; Realizamos um primeiro corte onde é removido todos os Cases (`incident_id`) que possuem atividades de algum desses períodos, mantendo então os cases que possuem atividades de Outubro/2012 a Março/2014. Encontramos, inclusive, um artigo onde foi mencionado a presença de outliers de 01/08/2013 a 31/03/2014 [9].

O dataset resultante contém 227455 eventos, 27160 cases e 19 variáveis (já mencionadas anteriormente). A comparação de ambos períodos é possível ver abaixo.

É notável que ocorre um padrão no inicio/final de cada mês, vendo somente a ocorrência das datas das atividades;

Após o corte, passamos para a ferramenta DISCO para visualizar o processo, e o resultado está abaixo:

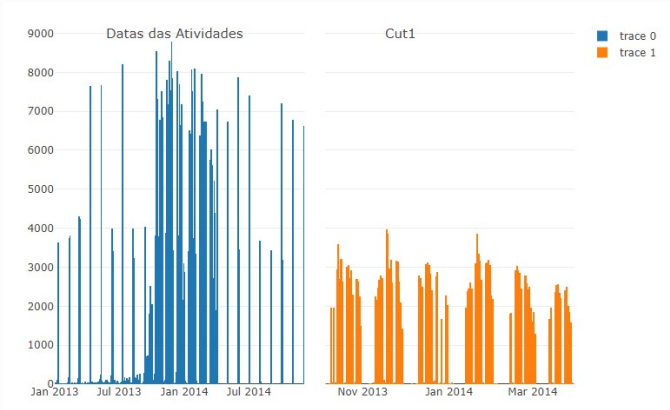

Figura 2. A comparação de ambos períodos.

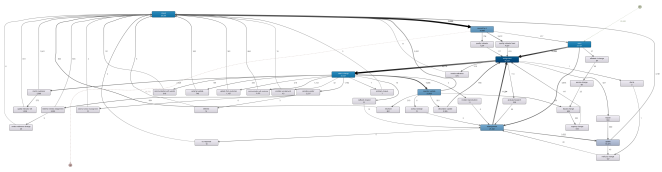

Figura 3. Visão do processo através da ferramenta DISCO para o corte 1.

B. *Corte 2 - Tempo de Resolução do case*

Utilizamos as variáveis first_day e last_day para calcular o tempo de resolução em dias. É possível que haja a necessidade de refazer esse cálculo caso ocorra mais algum "corte" específico do trace no log de eventos. Como por exemplo, se passarmos a considerar um determinado tipo de atividade como a última atividade do trace;

Os cases onde demoram ceca de 0 dias para serem resolvidos estão bastante presentes nos dados. Por ser um processo de atendimento ITIL não conseguimos obter muitas informações desses cases visto que podem ter sido chamados para tirar dúvidas ou até mesmo cases que foram "abertos" somente para manter alguma rastreabilidades/mapeamento.

A mesma ideia é de cases onde o tempo de resolução excede os 10 dias, olhando a questão de "dias úteis", cases que extrapolam esse tempo não são cases que queremos "atingir". Por exemplo, olhando na visão de cliente não gostaríamos que um problema demorasse mais que isso para ser resolvido.

Realizamos um segundo corte onde é removido todos os cases(`incident_id`) que possuem tempo de resolução menor que 1 dia e maior que 10 dias.

O log resultante contém 85081 eventos, 7171 cases e 19 variáveis (já mencionadas anteriormente). A comparação de todos os cortes realizados, olhando a visão tempo de resolução, pode ser analisada abaixo.

Curiosamente também podemos observar que pode ser que haja um padrão no tempo de resolução do case.

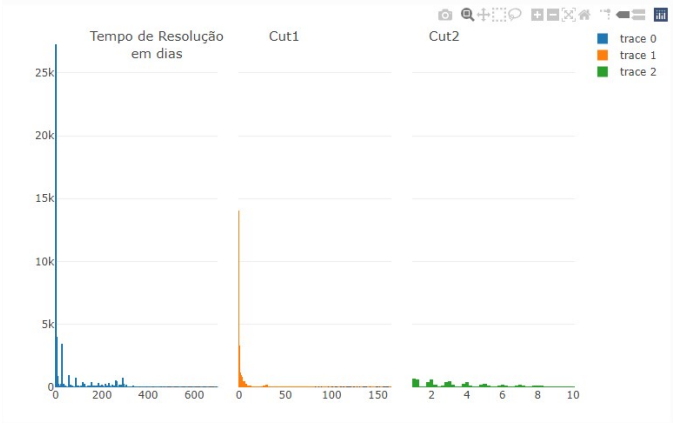

Figura 4. A comparação de todos os cortes realizados, olhando a visão tempo de resolução.

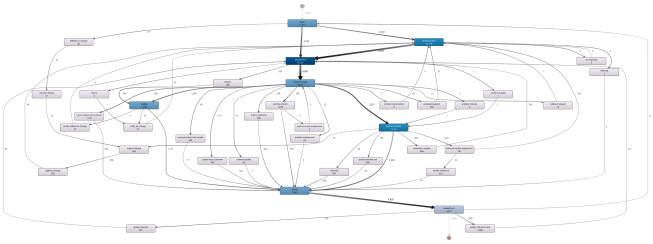

Figura 5. Visão do processo utilizando a DISCO para o corte 2.

C. *Corte 3 - Atividades em um case id*

Utilizando o boxplot é possível fazer um corte inicial no 3º quartil (Q3 75% dos dados presentes) indicando que teremos em média 15 eventos por case.

O log resultante contém 47643 eventos, 5540 cases e 19 variáveis (já mencionadas anteriormente). A comparação com o último o corte realizado, olhando a visão da quantidade eventos, pode ser analisada abaixo.

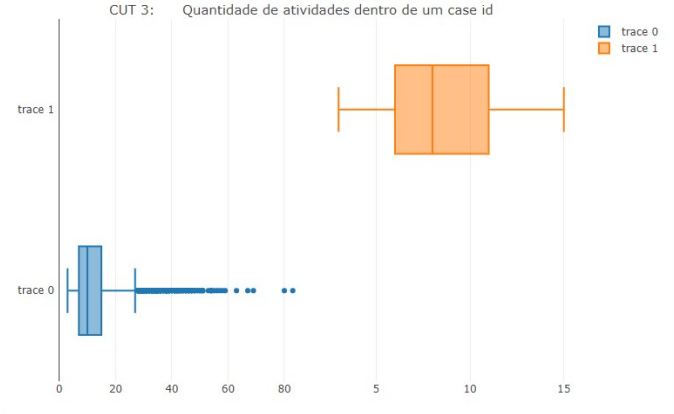

Figura 6. A comparação de todos os cortes realizados, olhando a visão tempo de resolução.

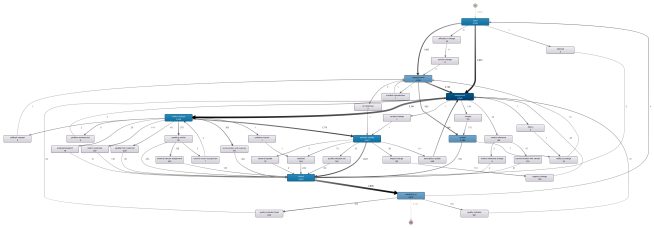

Figura 7. Visão do processo utilizando a DISCO para o corte 3.

D. Corte 4 - Repetição de Atividades dentro de um case id

Utilizando o conceito de sistemas ITIL, foi possível correlacionar o sistema utilizado nesse log de eventos com outro sistema também ITIL, visto por um membro do grupo.

- Muitas das variaveis que se repetem em um case id podem provocar loops e, dentro do log de eventos, tais atividades que se destacam são mais "qualitativas", ou seja, relacionadas a integridade do log e também na questão da satisfação do cliente, outras são bastante especificas relacionadas à alguma alteração de recursos ou até geradas por ação do cliente.
- Foi retirado então todas as atividades repetidas de cada case id.

O log resultante contém 36142 eventos, 5540 cases e 19 variáveis (já mencionadas anteriormente). A análise dos eventos removidos, olhando a visão da repetição de atividades em um case, pode ser analisada abaixo.

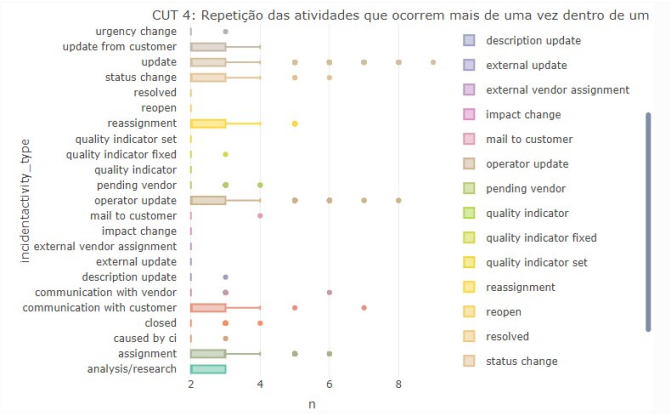

Figura 8. A análise dos eventos removidos, olhando a visão da repetição de atividades em um case.

Com esse corte conseguimos melhorar o desempenho de algoritmos que não lidam bem com repetição de eventos.

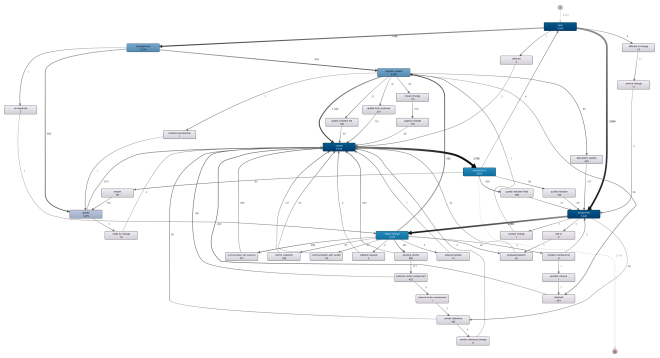

Figura 9. Visão do processo utilizando a DISCO para o corte 4.

E. Corte 5 - Atividades que não possuem uma média razoável de duração

Para estudar o comportamento ideal do processo começamos a olhar quais atividades consideramos iniciais e finais. A ideia inicial seria excluir todos os cases que não atenderem determinados parâmetros. Aparentemente, por enquanto não precisamos excluir todos os cases que não começam com 'open' e não terminam com 'closed', pois todas as atividades que começam ou terminam diferente disso são "qualitativas", podendo atrapalhar no fluxo do processo. Algumas opções:

- Excluirmos os eventos que aparecerem depois de closed, se houver closed e eventos antes de open, se houver open.
- Ver em média a duração de cada atividade em horas, atividades consideradas curtas poderiam ser ignoradas no case visto que demandariam baixo "custo" para um possível atendimento.

A segunda opção, inicialmente, parece ser mais razoável além de que assim o log de eventos poderá ter menos ruídos, visto que atividades que durarem em média 0 segundos não parece ser atividades que dependem necessariamente de um recurso. Podem, inclusive, serem atividades automatizadas pelo sistema. Olhando as atividades que possuem media 0 segundos a mesma questao levantada no corte anterior aparece, exceto pela atividade 'open' que pode estar ocorrendo durante um case, esta atividade manteremos.

O log resultante contém 20463 eventos, 5540 cases e 19 variáveis (já mencionadas anteriormente). A análise dos eventos removidos, olhando a visão tempo de resolução da atividade, pode ser analisada abaixo.

Poderemos olhar o 3º Quartil, caso seja necessário, em outro momento.

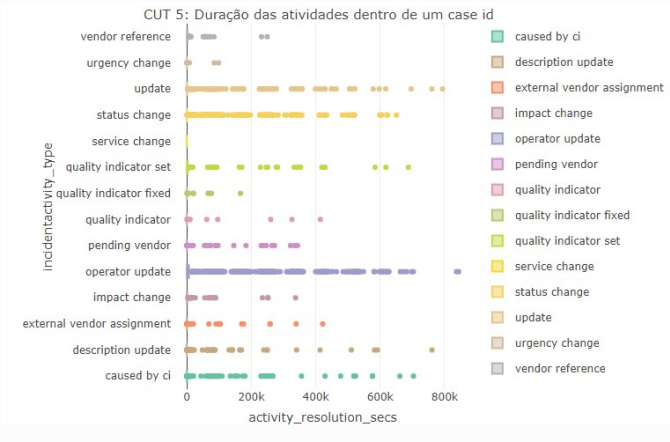

Figura 10. A análise dos eventos removidos, olhando a visão tempo de resolução da atividade.

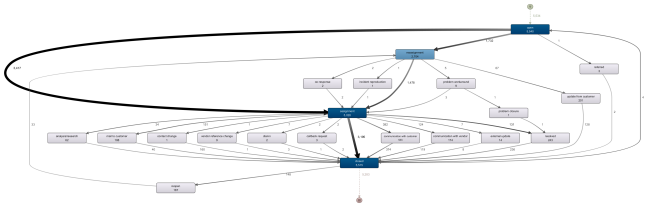

Figura 11. Visão do processo utilizando a DISCO para o corte 5.

F. Corte 6 - Atividade inicial e final de um Case Id

Após o último corte, agora, parece razoável excluirmos os cases que não possuem a atividade inicial 'open' ou atividade final 'closed' pois são em cases que possuem em média de 1 a 4 dias de duração mesmo ocorrendo alguns cases com duração maior que 8 dias.

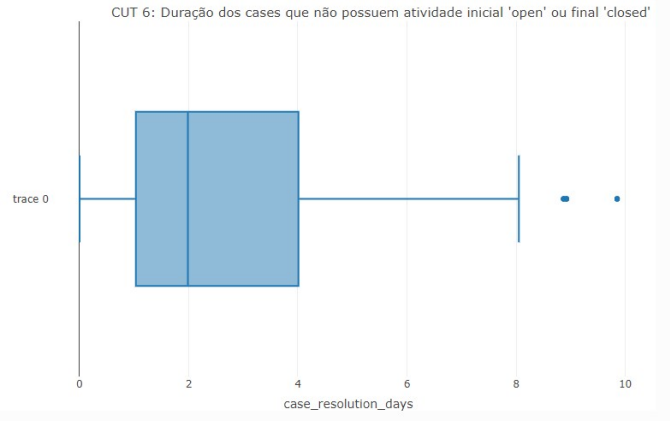

Figura 12. A análise dos cases removidos, olhando a visão tempo de resolução.

O log resultante contém 19300 eventos, 5278 cases e 19 variáveis (já mencionadas anteriormente). A análise dos cases removidos, olhando a visão tempo de resolução, pode ser analisada abaixo.

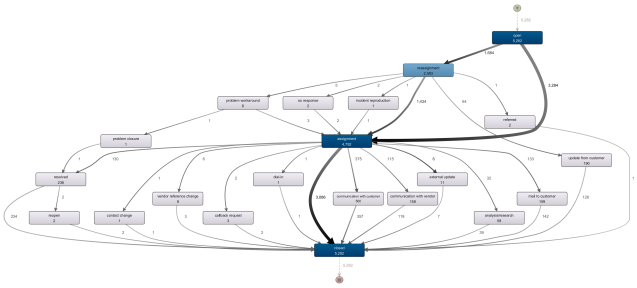

Figura 13. Visão do processo utilizando a DISCO para o corte 6.

VI. Conceitos levantados para aplicação do algoritmo ATS

Após realizado os cortes, foi possível enxergar um fluxo do processo de forma mais macro e mais compreensível para aplicarmos estudos e experimentos nos algoritmos de trabalhos consultados, para assim, mencionar sobre os testes realizados no algoritmo ATS.

Algumas definições da dissertação [7] foram necessárias para compreender e se familiarizar com a configuração do algoritmo aplicado ao nosso log:

A. Função Prefixo

O prefixo corresponde aos k primeiros eventos de um caso. [7]

B. Função Horizonte

Já a função horizonte funciona de outra forma: seja o último evento `ek` de um determinado caso `c1` (que pode ou não ser um um evento final). Essa função corresponde aos h eventos anteriores a `ek`. Para a situação em que h >= k, a função horizonte é análoga a função prefixo, ou colocado de outra forma, assume-se h=k. Para determinar que sempre sejam usados todos os eventos anteriores ao eventos atual, basta definir h = `Inf`. Isso significa dizer que sempre será usado todo o histórico do evento. [7]

C. Abstrações de conjunto

O trabalho utiliza três abstrações importantes para representar um evento: [7]

- Representação por **sequência** - SEQ, onde um evento é representado pela sequência que ocorre as atividades;
- Representação de **multiconjunto** - MSET, onde não importa a ordem da ocorrência das atividades e sim a quantidade de vezes;
- Representação de **conjunto** - SET, onde nem a ordem e nem a quantidade de ocorrências da atividade são consideradas.

D. Sistema de Transição

Um sistema de transição é representado por uma tupla TS = (S,E,T) onde S é o conjunto de estados, E é o conjunto de eventos e $T \subseteq S \times T \times S$ é o conjunto de transições que descreve como o sistema se move de um estado a outro. Para formar a

representação de estado, definem-se funções específicas com base nos atributos selecionados e na abstração de conjunto a ser usada. [7]

E. Cálculo da Predição

Foi utilizado funções de similaridade distintas para que os requisitos de cada abstração de conjunto seja atendido. Foi utilizado então a **similaridade de Jaccard** para as abstrações SET e MSET e para a abstração SEQ foi utilizada a **distância de Damerau-Levenshtein**. [7]

F. Métricas para Validação

Foram selecionadas duas métricas para a avaliar a assertividade da predição:

1) **Mean Absolute Percentage Error** (MAPE) - Erro médio absoluto percentual;
2) **Root Mean Squared Percentage Error** (RMSPE) - Erro quadrático médio percentual;

Ambos escolhidos por utilizarem porcentagem, e assim facilitar a comparação, além disso, como os tempos de resolução dos casos em segundos pode ser valores altos, com o pré-processamento atual também é evitado que o MAPE lide com valores muito baixos. A última métrica evita o viés do MAPE pois seleciona predições mais otimista (evita predições menores). [7]

VII. Configuração de Experimentos

Configuramos o algoritmo para nossos experimentos com os seguintes artefatos:

- Log de eventos após tratamento: O log de eventos utilizado foi o dataset resultado após o corte 6.
- Atributos: Decidimos utilizar alguns atributos que consideramos relevantes como **atividade, recurso e número do chamado**. Este último atributo foi considerado pois um chamado pode estar relacionado a mais de um incidente.
- Conjunto de dados de treinamento e validação: Separamos o log de eventos após o tratamento e teste em alguns subconjuntos como **60%/40%, 70%/30% e 80%/20%**.
- Horizontes: Foi utilizado os mesmos horizontes definidos **(1,3,5,6,7,Inf)**, por mais que o ideal fosse que a lista de horizontes variasse de 1 até o total de eventos de acordo com o autor após análise e testes realizados com horizontes maiores não foi encontrada diferença significativa, além disso o horizonte Inf contemplará todos os eventos do caso.

VIII. Resultados Obtidos

Métricas dos testes para os diferentes conjuntos de treino e validação juntamente com a porcentagem de casos Non-Fitting (casos que aparecem no conjunto de validação e porém não no conjunto de treino). Abaixo está representado em formas de gráficos os resultados obtidos para cada parâmetro de Horizonte definido.

A. Horizonte 1

Figura 14. H1 RMSPE.

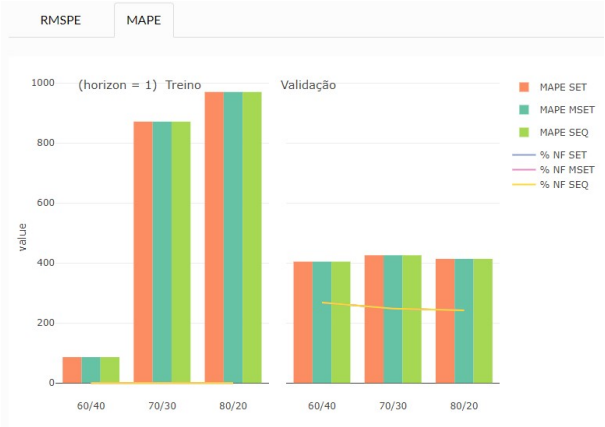

Figura 15. H1 MAPE.

B. Horizonte 3

Figura 16. H3 RMSPE.

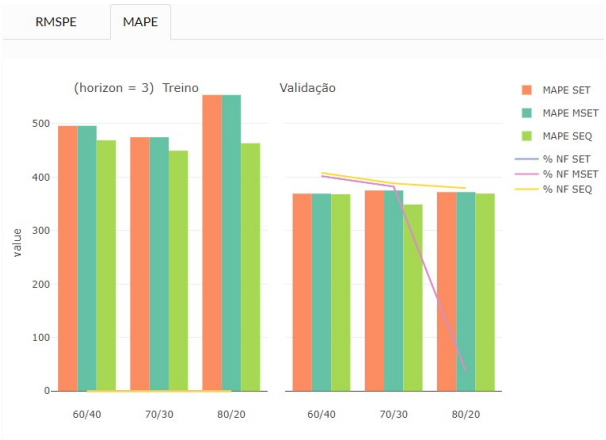

Figura 17. H3 MAPE.

C. *Para os Horizontes: 5,6,7 e Inf*

Figura 18. RMSPE.

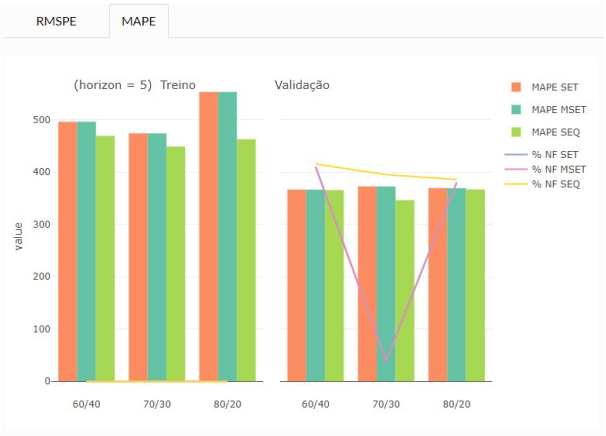

Figura 19. MAPE.

IX. Análise dos resultados

Os gráficos das figuras da sessão 6 são execuções do algoritmo ATS para cada subconjunto e horizonte que foram utilizados na configuração e mostram os resultados em termos de erro de predição medido com RMSPE e MAPE.

Decidimos por separar as visões entre os subconjuntos de treino e validação por realizarmos um pré-processamento para fins de estudo, utilizando somente de artefatos relacionados ao log de eventos e experiência com processos de sistemas ITIL para levantar questões que poderiam gerar ou não um impacto para o cliente. Foi instruído que cada manipulação do log de eventos em uma situação real deve ser acompanhada por um agente que conhece o processo de negócio e com isso, alterações drásticas no log são evitadas.

Após o corte 3, mencionado anteriormente na seção V.C, são realizados filtros com o objetivo de entender quais são as atividades que geram algum impacto para com quem realizou a abertura do incidente, evitando assim atividades que não tiveram mudanças significativas durante um case. Com isso, é esperado que apareçam erros com valores maiores nos resultados dos conjuntos de validação, bem como uma taxa de eventos non-fitting.

Os gráficos das figuras 14 e 15 é a execução para o horizonte 1 (A), ou seja, utilizando apenas um evento do caso, os modelos são os mesmos para as três abstrações o que faz sentido pois nessa configuração o evento atual para uma abstração será o mesmo para as demais, além disso a taxa non-fitting se manteve independente do erro. O conjunto utilizado com a configuração de treino e validação 60% e 40%, respectivamente, obteve um melhor desempenho em termos do erro.

Os gráficos seguintes, das figuras 16 e 17, mostram mudanças nos resultados quando utilizado o horizonte 3, as abstrações SET e MSET continuam muito parecidas, apesar disso, em termos de casos non-fitting a abstração MSET chega a uma taxa muito inferior em comparação com a abstração SET para a configuração utilizando o conjunto de treino e validação 80% e 20% respectivamente, ao contrário do que é analisado para o conjunto 70% e 30% onde o desempenho do conjunto de validação está superior em comparação ao conjunto de treino em termos do erro RMSPE. A abstração SEQ somente possuí desempenho melhor ou parecido com as demais abstrações ao analisar os resultados em termos do erro MAPE, que possuí uma restrição em lidar com valores muito próximos de zero e por consequência um desempenho inferior aos resultados em temos de RMSPE.

Para os outros horizontes presentes na configuração, ou seja, horizontes 5, 6, 7 e Inf, os resultados para o erro MAPE são muito próximos do que foi analisado para o horizonte 3, porém, a baixa taxa de casos non-fitting foi afetada para a configuração que utiliza o conjunto de treino e validação 80% e 20% respectivamente onde a taxa inverteu em comparação com o horizonte 3 deixando assim a configuração que utiliza o conjunto de treino e validação 70% e 30% com melhor desempenho em termos de resultado do erro RMSPE, inclusive, a abstração SEQ também mostrou um bom resultado em comparação com as demais abstrações nesse horizonte.

Os cortes realizados após a definição da quantidade máxima de eventos dentro de um case fez com que o case chegasse a

ter em média 5 eventos, explicando resultados muito próximos, até iguais, para os horizontes 5, 6, 7 e Inf.

X. Estudo LSTM

A arquitetura da Rede de Memória Curto Prazo Longo (LSTM) é um tipo de especial de RNN, capaz de aprender dependências de longo prazo. Utilizando esse conceito nós redirecionamos esforços em extrair ideias para treinar modelos precisos de logs de eventos de processos de negócio através de dois artigos. O primeiro artigo consultado foi o "Learning Accurate LSTM Models of Business Process" [10] que propõe uma abordagem para aprender modelos que geram traces, consistindo de triples(tipo de evento, função e timestamp. Já o segundo artigo é o "Predictive Business Process Monitoring with LSTM Neural Networks" [11] que tem como teor mostrar que tal arquitetura pode resolver problemas de predição da próxima atividade do que outras aplicações. A ideia de utilizar o LSTM para o nosso desafio seria a confirmação de que ao encontrar a predição da próxima atividade e seu respectivo registro de timestamp, haveria um rendimento de precisão maior do que predize-los usando modelos separados, no nosso caso o ATS. Portanto, o nosso intuito seria de reforçar que a técnica de predizer a próxima atividade de um caso em execução e seu respectivo timestamp usando redes neurais superaria as baselines existentes no que se refere à datasets da vida real.

XI. Conclusão

Foi possível realizar estudos estatísticos com apoio de linguagens de programação e também da ferramenta de mineração de dados Disco para entender melhor o log e aplicar o algoritmo ATS, buscando entender e replicar algumas abordagens escolhidas pelo autor para *predição de tempo de resolução de incidentes e realizando uma análise do impacto de casos non-fitting nos modelos de predição gerados a partir de múltiplos atributos descritivos* [7]. Como mencionado, a ideia era estudar o log de eventos e os tratamentos desse log, que gerou a possibilidade de analisar o impacto ao acabarmos executando filtros ao não possuírmos a análise de um agente que entende do fluxo do processo presente no log para evitar que os casos que estejam presentes no conjunto de validação não estejam presentes no conjunto de treino.

Em algumas das análises dos dados foi notada a variação que ocorre durante o passar do tempo, indicando períodos que possuem maiores aberturas de incidentes ou uso intenso do sistema seguidos de períodos com baixa ocorrência. Isso pode indicar algumas mudanças que podem ter ocorrido na qualidade ou uma campanha de um serviço ofertado, uma mudança de estrutura, mudança de processo e até mesmo mudança drástica de recursos e/ou clientes. Por conta disso,poderíamos ter enriquecido o trabalho abordando conceitos de *concept-drift*.

Ao olhar o trabalho desenvolvido pelo autor [7], poderíamos ter feito mais experimentos baseado em alguns dos experimentos apresentados em sua dissertação, gerando modelos com cenários diferentes ao executar o algoritmo atributo a atributo, de forma que houvesse combinações entre elas ou até mesmo da forma como o autor [7] fez com técnica de ranking de atributos. Poderíamos também ter executado o algoritmo para cada etapa de corte que realizamos e analisar os erros para comparar com os resultados utilizados na configuração apresentada neste trabalho.

O autor também menciona que a literatura utilizada por ele concluí que a técnica que apresentou melhor resultado em termos de assertividade na predição do tempo de resolução de incidentes foi a que usa *long-short term memory networks* (LSTM) [11], porém, essa técnica não é vantajosa pela dificuldade em interpretar seus resultados e os fatores determinantes para a decisão final, além de que o ganho na assertividade nem sempre é benéfico comparando a complexidade da implementação [7] de um algoritmo desse nível.

Referências

[1] A. Rozinat. BPI Challenge 2014 — Get Started Now! [Online]. Available: https://fluxicon.com/blog/2014/04/bpi-challenge-2014/

[2] S. Aleem, L. F. Capretz, and F. Ahmed, "Business process mining approaches: A relative comparison," 2015.

[3] G. Marketing. What is the difference between ITSM and ITIL? [Online]. Available: https://www.globallogic.com/insights/blogs/itsm-vs-itil/

[4] M. R. Peña and S. Bayona-Oré, "Process mining and automatic process discovery," in *2018 7th International Conference On Software Process Improvement (CIMPS)*, 2018, pp. 41–46.

[5] fluxicon. BPI Challenge 2014 — Get Started Now! [Online]. Available: Discover your processes

[6] W. Van der Aalst, M. Schonenberg, and M. Song, "Time prediction based on process mining," *Information Systems*, vol. 36, pp. 450–475, 2011.

[7] A. G. L. Fernandes, "Tratamento do impacto de casos non-fitting em predição de tempo de resolução usando mineração de processos com múltiplos atributos," 2019.

[8] L. Marques, "Repositório com códigos desenvolvidos e algoritmos utilizados para a disciplina de Mineração de Dados (ACH2187)," Available at https://github.com/marqueslarissa/data-mining-2021 (2021-2022).

[9] G. Cacciola, R. Conforti, and N. Hoang, "Rabobank: A process mining case study bpi challenge 2014 report," *Università della Calabria*, 2014.

[10] M. Camargo, M. Dumas, and O. G. Rojas, "Learning Accurate LSTM Models of Business Processes," *Universidad de los Andes*, pp. 286–302, 2019.

[11] N. Tax, I. Verenich, M. La Rosa, and M. Dumas, "Predictive business process monitoring with LSTM neural networks," in *Proceedings of the 29th International Conference on Advanced Information Systems Engineering*. Springer, 2017.