2020 46ª Conferência Euromicro sobre Engenharia de Software e Aplicações Avançadas (SEAA)

Modelando pipelines de dados

Aiswarya Raj, Jan Bosch
Chalmers University of Technology
Goteborg, Suécia
{aiswarya,jan.bosch}@chalmers.se

Helena Holmstrom Olsson
Malmo University
Malmo, Suécia
helena.holmstrom.olsson@mau.se

Tian J. Wang
Ericsson
Gotemburgo, Suécia
tian.j.wang@ericsson.com

Resumo—Os dados são a nova moeda e a chave para o sucesso. No entanto, coletar dados de alta qualidade de várias fontes distribuídas requer muito esforço. Além disso, existem vários outros desafios envolvidos no transporte de dados de sua origem para o destino. Os pipelines de dados são implementados para aumentar a eficiência geral do fluxo de dados da origem ao destino, uma vez que é automatizado e reduz o envolvimento humano que é necessário de outra forma.

Apesar da pesquisa existente sobre pipelines ETL (Extract-Transform-Load) e ELT (Extract-Load-Transform), a pesquisa sobre esse tópico é limitada. Os pipelines ETL/ELT são representações abstratas dos pipelines de dados de ponta a ponta. Para utilizar todo o potencial do pipeline de dados, devemos entender as atividades nele e como elas estão conectadas em um pipeline de dados de ponta a ponta.
Este estudo fornece uma visão geral de como projetar um modelo conceitual de pipeline de dados que pode ser usado posteriormente como uma linguagem de comunicação entre diferentes equipes de dados. Além disso, pode ser usado para automação de monitoramento, detecção de falhas, mitigação e alarme em diferentes etapas do pipeline de dados.

Termos do índice—Pipelines de dados; modelo conceitual; trabalho de dados fluxo; linguagem específica do domínio; metodologia ágil

I. INTRODUÇÃO

Os dados estão se tornando cada vez mais populares no setor devido à importância dos produtos de dados, como APIs, painéis, benchmarks e criação de relatórios. O papel que os dados desempenham no processo de tomada de decisão e o desenvolvimento de modelos ML e DL o torna ainda mais importante. Portanto, todos os processos associados aos dados, desde a geração até a recepção dos dados, precisam ser monitorados. Detecção de falhas, relatórios e mitigação do efeito de falhas são complexos, mas inevitáveis durante a construção de produtos de dados eficientes.

Os pipelines de dados são cadeias complexas de atividades que manipulam dados em que a saída de um componente se torna a entrada do outro [1], permitindo assim um fluxo de dados suave e automatizado da origem ao destino. Um pipeline de dados começa com uma fonte de dados que gera dados e termina em um destino que recebe os dados processados. O destino final de um pipeline de dados não precisa ser o armazenamento de dados. Em vez disso, pode ser qualquer aplicativo, como uma ferramenta de visualização [2] [3], modelos de Machine Learning (ML) [4] [5] ou modelos de Deep Learning (DL) [6] [7]. Os componentes no pipeline de dados são capazes de automatizar os processos envolvidos na extração, transformação, combinação, validação e carregamento de dados [8]. Os pipelines de dados podem processar diferentes tipos de dados, como dados contínuos, intermitentes e em lote [9]. Além disso, os pipelines de dados eliminam erros e aceleram os processos de dados de ponta a ponta, o que, por sua vez, reduz a latência no desenvolvimento de produtos de dados. Portanto, o uso de pipelines de dados é uma necessidade absoluta para todas as empresas orientadas a dados.

Embora os pipelines de dados tenham o potencial de superar os desafios de gerenciamento de dados por meio de automação, monitoramento, detecção de falhas, etc., modelar pipelines de dados para um caso de uso exige uma quantidade de tempo e esforço irracional. Precisamos identificar as atividades que consomem dados, a saída de cada atividade, ordem de execução, métodos de monitoramento, armazenamentos intermediários, onde colocar o armazenamento no pipeline, método de coleta de dados, etc e variam entre as empresas. Portanto, modelar um pipeline de dados é importante e demorado.

Este estudo aborda os problemas acima mencionados propondo um modelo conceitual que é desenvolvido com base em um estudo de caso múltiplo realizado em uma empresa de telecomunicações de grande porte. A contribuição deste trabalho é tríplice. Primeiro, ele descreve os desafios associados ao gerenciamento de dados e ao uso de pipelines de dados existentes. Em segundo lugar, é apresentado um modelo conceitual de um pipeline de dados de ponta a ponta que pode ser usado como referência durante a construção de pipelines de dados para aplicativos como modelos ML/DL para incorporar monitoramento automático, detecção de falhas, mitigação e técnicas de alarme. O modelo conceitual é validado por meio de um estudo de caso com três empresas líderes nos setores de manufatura, telecomunicações e automobilístico. Além disso, o artigo mapeia os desafios que podem ser potencialmente resolvidos pelo uso do modelo de pipeline de dados proposto.

O restante deste artigo está organizado da seguinte forma. Na próxima seção, apresentamos os antecedentes do estudo. A Seção III discute a metodologia de pesquisa adotada para a condução do estudo. A Seção IV apresenta os casos de uso e a Seção V descreve os desafios usando os pipelines existentes. A Seção VI detalha o metamodelo do pipeline de dados. A Seção VII descreve o modelo conceitual que utilizamos como base para nossa análise.
Um estudo de validação é detalhado na seção VIII e a seção IX descreve as ameaças à validade. A Seção X resume nosso estudo e as conclusões.

II. FUNDO

O desenvolvimento baseado em dados foi adotado pelas empresas percebendo os benefícios que podem ser gerados a partir de dados [10]. As tecnologias para desenvolvimento orientado a dados precisam de uma enorme quantidade de dados para seu processamento. Consequentemente, a coleta, armazenamento e processamento de grandes quantidades de dados tornou-se uma necessidade [11]. Com o aumento da quantidade de dados, os dados

978-1-7281-9532-2/20/$31,00 ©2020 IEEE
DOI 10.1109/SEAA51224.2020.00014

os desafios de gestão também aumentaram [12]. Os pipelines de dados podem ser uma solução potencial para superar pelo menos alguns desses desafios.

Os pipelines de dados são amplamente classificados em duas categorias, a saber, ETL e ELT. ETL significa Extract, Transform and Load, enquanto ELT significa Extract Load and Trans form. P. Vassiliadis apresentou uma pesquisa sobre os processos e ferramentas Extraction Transformation-Loading (ETL). O estudo descreve uma abordagem padronizada para a construção de ferramentas de modelagem conceitual e lógica para processos ETL [13].

Além disso, cada componente do trio ETL é analisado separadamente para as atividades que acontecem dentro de cada componente, identificados os desafios em tempo real associados a cada um deles e as soluções para superar esses desafios são explicadas em detalhes.

Em [14], os autores propuseram uma abordagem conceitual baseada em UML para modelar processos ETL. Um grupo de conceitos UML é utilizado para representar os processos ETL, como integração de fontes de dados, transformação, geração de chaves e assim por diante. Os autores afirmam que uma abordagem baseada em UML torna o modelo simples de entender e poderoso. Tilmann e Hans descreveram um pipeline de análise de big data com estágios abstratos [15]. Eles também discutem a necessidade de processos semelhantes ao ETL em benchmarking e propuseram e implementaram uma estrutura semelhante ao ETL.

Um pipeline de aprendizado de máquina é proposto por Amershi et. al em [16] com base em um estudo de caso da Microsoft. Ele discute os nove estágios do fluxo de trabalho de ML junto com as práticas recomendadas seguidas na Microsoft. Os autores mencionam a importância dos dados em aplicativos de IA, ilustrando os três estágios relacionados a dados no fluxo de trabalho de ML.

Embora esses estudos estabeleçam uma base sólida, os profissionais enfrentam vários desafios de qualidade de dados e problemas relacionados à governança e segurança de dados ao lidar com dados em tempo real. Nosso estudo visa projetar um modelo conceitual que possa atuar como uma linguagem específica de domínio para pipelines de dados tolerantes a falhas.

III. METODOLOGIA DE PESQUISA

O objetivo deste estudo é entender o pipeline de dados existente, bem como os desafios vivenciados na empresa de caso e desenvolver um modelo conceitual do pipeline de dados robusto. Com base nos objetivos do estudo, formulamos as seguintes questões de pesquisa: • **RQ1:** Quais são os desafios relacionados ao gerenciamento de dados

e pipeline de dados que os profissionais da empresa de caso vivenciam? • **RQ2:** Quais são os elementos essenciais de um pipeline de dados de ponta a ponta,

automatizado e rastreável?

A metodologia de pesquisa adotada para o estudo é ilustrada na fig. 1.

A. Estudo de caso exploratório

Uma abordagem qualitativa foi escolhida para o estudo de caso, pois permite que os pesquisadores explorem, estudem e compreendam os casos do mundo real em seu contexto com mais profundidade [17]. Como o conceito de pipeline de dados é um tópico menos explorado em pesquisas,

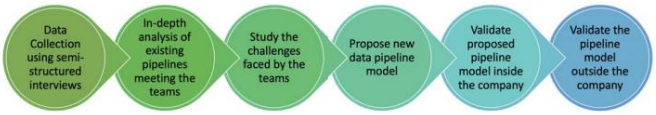

Fig. 1. Metodologia de pesquisa

TABELA I
DESCRIÇÃO DE CASOS DE USO E PAPÉIS DOS ENTREVISTADOS

ID do caso	Use casos na empresa do caso	Especialistas Entrevistados	
		EU IA	Papel
A	Pipeline de coleta de dados para análise de dados	R1	Cientista de Dados Sênior
		Cientista de Dados R2	
B	Construindo pipelines de dados para governança de dados	R3	cientista de dados
		R4	Sistema Analítico Arquiteto
		Desenvolvedor de Software R5	
C	Pipeline de aprendizado de máquina	R6	cientista de dados
		R7	Cientista de Dados Sênior
		Desenvolvedor de Software R8	
		R9	Cientista de Dados Sênior

adotamos uma abordagem de estudo de caso [18]. Cada caso no estudo pertence a um caso de uso que faz uso de dados. Embora os casos em nosso estudo sejam casos de uso diferentes da mesma empresa, eles utilizam dados para atividades diferentes e podem se beneficiar do pipeline de dados que desenvolvemos. Três casos de uso selecionados na empresa são apresentados na tabela 1.

B. Coleta de dados

Os dados qualitativos foram coletados por meio de entrevistas e reuniões [19]. Com base no objetivo da pesquisa, explorar e estudar os aplicativos que consomem dados na empresa, foi formulado um roteiro de entrevista com 45 questões categorizadas em seis seções. A primeira e a segunda seções

focado no histórico do entrevistado. A terceira e a quarta seções focaram na coleta e processamento de dados em vários casos de uso e a última seção perguntou sobre testes de dados e práticas de monitoramento e os impedimentos enfrentados durante cada etapa do pipeline de dados. O roteiro de entrevista foi elaborado pela primeira autora e revisado pela segunda e terceira autoras. De acordo com as recomendações, foram adicionadas perguntas extras, retiradas algumas perguntas semelhantes e irrelevantes e algumas perguntas modificadas. Por fim, foi desenvolvido um protocolo de entrevista com 30 perguntas em seis categorias diferentes. Todas as entrevistas, exceto três, foram realizadas por videoconferência. Cada entrevista durou de 50 a 100 minutos. As entrevistas foram gravadas com a autorização dos entrevistados e posteriormente transcritas para análise.

Um dos autores trabalha na empresa do caso e o primeiro autor é um consultor que participa de reuniões semanais com cientistas e analistas de dados. Os dados coletados nas reuniões e discussões também são incorporados.

C. Análise dos dados

As gravações em áudio das entrevistas foram transcritas e um resumo foi elaborado pela primeira autora. As transcrições foram investigadas quanto a relações, semelhanças e diferenças.
As transcrições das entrevistas foram codificadas de forma aberta seguindo o

orientações em [20]. O primeiro autor preparou notas durante as reuniões com a equipe e as analisou posteriormente. O principal ponto de contato, que também é autor, ajudou na análise das partes do gasoduto, bem como da infraestrutura usada para construí-lo. Essas notas, juntamente com os códigos das transcrições, foram analisadas posteriormente para obter uma visão de ponta a ponta de diferentes casos de uso. Também ajudou a entender as partes comuns a todos os casos de uso. Após análise cuidadosa dos dados coletados e com base nas contribuições dos outros dois autores, o primeiro autor desenvolveu o primeiro modelo conceitual que foi refinado por meio de iterações.

D. Estudo de Validação

O estudo de validação é realizado interna e externamente por meio de entrevistas qualitativas seguidas de sessões de feedback. Primeiro, o modelo conceitual do pipeline de dados foi apresentado pelo primeiro autor às equipes da empresa de caso. As reflexões sobre o pipeline de dados, opinião geral, acordos e sugestões de melhoria foram coletadas de cada profissional. Estas reflexões são consideradas como validação interna.

Para validação externa de nossas descobertas, duas empresas de manufatura foram selecionadas. Um guia de entrevista foi preparado pelo primeiro autor para validar o modelo conceitual de pipeline de dados. O primeiro autor apresentou o modelo. O segundo e terceiro autores conduziram uma entrevista seguida de uma discussão para coletar dados. A sessão inteira foi gravada para o primeiro caso e para o segundo, o primeiro autor fez anotações. Assim, o feedback de 20 praticantes foi coletado e registrado.

O modelo conceitual foi então modificado para abordar algumas das questões levantadas durante as discussões. Além disso, as contribuições dos profissionais são incorporadas ao modelo conceitual. As demais questões serão abordadas em trabalhos futuros.

4. CASOS DE USO

Esta seção apresenta os pipelines de dados existentes usados na empresa de telecomunicações. Cada um desses casos de uso é separado e não há interação entre esses pipelines.

A. Processo de coleta de dados

A empresa coleta dados de várias fontes distribuídas em todo o mundo, o que é uma atividade desafiadora. Quando os dados são coletados de um dispositivo localizado em outro país ou da rede do cliente, eles devem estar de acordo com o acordo legal. Além disso, informações confidenciais nos dados devem ser tratadas com responsabilidade. Além disso, a coleta de dados deve considerar o fato de que diferentes fontes de dados geram dados em diferentes frequências e formatos. Por exemplo, os dados podem ser coletados de forma contínua, intermitente ou em lotes. Além disso, o próprio mecanismo de coleta de dados deve ser capaz de se ajustar a diferentes intensidades de fluxo de dados.

Quando a coleta de dados é automatizada, esses desafios devem ser abordados adequadamente. A Fig. 2 mostra a coleta automática de dados dos dispositivos. Nesse cenário, o dispositivo é colocado dentro de um equipamento de propriedade do cliente. No entanto, os dados do dispositivo são extraídos excluindo os dados confidenciais do cliente

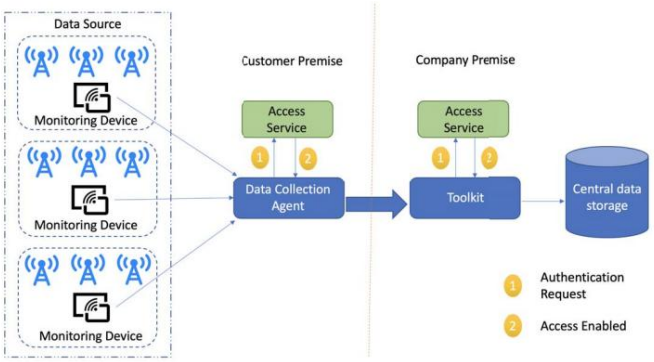

Fig. 2. Processo de coleta de dados

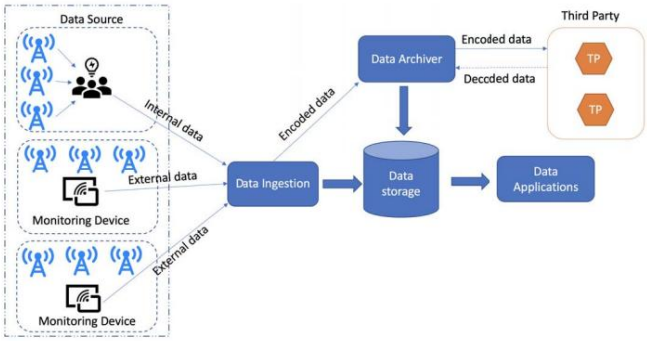

Fig. 3. Pipeline de dados para governança de dados

Informação. As estações base possuem nós, bem como um dispositivo para monitorar e gerenciar os nós. Os agentes de coleta de dados são equipamentos localizados nas instalações do cliente (localização física) que podem interagir diretamente com os nós ou com o dispositivo para coletar os dados. No entanto, para garantir que o agente de coleta de dados tenha o direito de coletar dados, ele é autenticado com a ajuda do serviço de acesso. Os dados assim recolhidos são transmitidos através de um túnel seguro para o toolkit localizado nas instalações da empresa. Este agente de coleta de dados também precisa da ajuda do serviço de acesso para autenticação. Depois que o agente nas instalações do cliente obtém os dados, eles são armazenados no armazenamento de dados central. As equipes podem obter os dados do armazenamento de dados central.

B. Pipeline para Governança de Dados O

pipeline de dados mostrado na fig. 3 é desenvolvido para atender as equipes da empresa que estão trabalhando com dados sempre que precisam (com o termo 'dados', queremos dizer o link de onde os dados originais podem ser baixados). Esse pipeline de dados obtém dois tipos de despejos de dados: interno e externo. O despejo de dados interno são os dados que são ingeridos pelas equipes dentro da empresa e o despejo de dados externo são os dados coletados diretamente dos dispositivos nos campos. O método de ingestão de dados é diferente para diferentes fontes e os dados ingeridos são armazenados no armazenamento de dados para uso posterior. Os dados podem ter links criptografados que precisam ser descriptografados antes de armazená-los.

Sempre que um despejo de dados criptografados é encontrado, o arquivador de dados

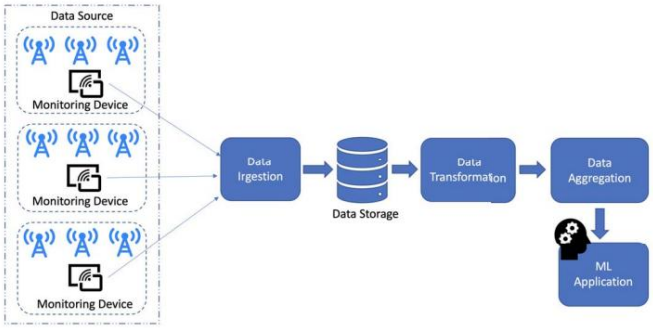

Fig. 4. Pipeline para sistemas de aprendizado de máquina

módulo envia para serviços de terceiros para descriptografia. Links decodificados de terceiros são armazenados. Assim, dados de diferentes fontes são disponibilizados em um local central. As equipes podem solicitar dados de qualquer estágio do pipeline. A tubulação é monitorada manualmente pelo 'guardião de fluxo' que é responsável por identificar as falhas e resolvê-las.

C. Pipeline para Sistemas de Aprendizado de Máquina O

pipeline de dados tem quatro etapas principais, a saber, ingerir, armazenar, transformar e agregar. Os dados são gerados pela fonte que é coletada por uma zona especial no campo. O módulo de ingestão de dados é conectado a essas zonas no campo e os dados coletados são inseridos no pipeline como lotes.

Quando novos arquivos compactados são encontrados durante as verificações periódicas, a transação é registrada e baixada. Esses novos arquivos são carregados no diretório de arquivo do cluster de dados. Os dados armazenados no cluster não podem ser usados diretamente pelos aplicativos de ML. Além disso, os registros de dados coletados de diferentes dispositivos podem ser de diferentes formatos. Eles precisam ser convertidos para um formato adequado. Essa conversão é realizada pelo módulo de transformação de dados. A transformação de dados verifica o

novos arquivos no diretório de arquivamento do cluster de dados e, quando encontrados, são buscados, descompactados e processados para convertê-los em um formato apropriado. Os dados convertidos são então fornecidos como entrada para o módulo de agregação de dados, onde os dados são agregados e resumidos para formar dados estruturados que são posteriormente fornecidos como entrada para os modelos de ML.

V. DESAFIOS COM GERENCIAMENTO DE DADOS

Nesta seção, são apresentados insights sobre os desafios de gerenciamento de dados com base nas descobertas de nossa análise de casos cruzados.
Identificamos dez grandes desafios com gerenciamento de dados e pipelines de dados existentes usados na empresa.

A. Disponibilidade de dados

A disponibilidade dos dados certos no formato certo no momento certo é um requisito básico para o desenvolvimento bem-sucedido de produtos de dados. A coleta de dados é uma tarefa difícil e às vezes falha devido a falha de autenticação, fatores ambientais ou falha do dispositivo de coleta. Mesmo depois de coletar uma enorme quantidade de dados dos dispositivos, eles podem não chegar ao destino pretendido. Os dados coletados podem estar incompletos. ou seja

nem todas as informações estarão disponíveis no data warehouse.

Por exemplo, devido a falhas de software, partes dos dados podem ser perdidas. A menos que tenhamos um mecanismo de rastreamento, essa perda é difícil de identificar. Além disso, se dados bem definidos forem fornecidos como entrada para o modelo, ele não poderá fornecer o mesmo desempenho quando dados invisíveis do mundo real forem fornecidos, levando ao subajuste.

B. Qualidade dos

dados Para sistemas com fome de dados, como ML e DL, a qualidade dos dados é crucial. Quando dados de baixa qualidade são alimentados aos algoritmos, a saída de baixa qualidade será produzida. Por exemplo, ao coletar registros de falhas dos dispositivos no campo, deve haver uma distinção clara entre falhas devido a problemas ambientais

fatores e falhas devido à falha do dispositivo. O desafio é quando os dados são transformados para caber em uma estrutura predefinida, partes "desnecessárias" são removidas. Portanto, precisaríamos de um método para salvar o arquivo original. Quando os dados são transformados em tempo real e armazenados, isso não seria possível. É sempre bom ter o arquivo original de dados brutos armazenado para que possa ser acessado sempre que os dados estruturados se tornarem insuficientes para atender aos requisitos.

C. Instabilidade do fluxo de dados

O fluxo de dados para o armazenamento de dados da empresa nem sempre é estável. O dispositivo do lado da empresa deve estar preparado para receber os dados dos dispositivos distribuídos. O pipeline, se existente, deve ser capaz o suficiente para lidar com o fluxo de dados através dele. Se várias equipes solicitarem dados simultaneamente, o pipeline poderá atendê-los. Além disso, os elementos no pipeline devem ser monitorados adequadamente quanto a falhas.

A falha dos elementos do pipeline leva à instabilidade do fluxo de dados.
O upload oportuno dos dados processados é obrigatório, especialmente no caso de dependências. Se o pipeline de dados estiver aceitando dados contínuos, intermitentes e em lote, em alguns pontos, a entrada de dados será pesada e em outros momentos, ele terá que lidar apenas com dados contínuos. Isso também leva à instabilidade do fluxo de dados se o pipeline não for capaz de ajustar sua capacidade.

D. Silos de Dados

Um silo de dados é a lacuna entre os dados e os consumidores que precisam dos dados. É resultado de arquitetura pobre, sistemas operacionais legados e cultura corporativa ultrapassada. O principal problema é que os dados ficam isolados e presos sem chegar aos consumidores. Quando equipes individuais desenvolvem seu próprio pipeline de dados, isso também pode levar a silos de dados. Há uma alta probabilidade de que várias equipes executem as mesmas atividades para diferentes casos de uso.

E. Dependências de dados A

dependência de dados ocorre quando uma equipe ou dispositivo depende do resultado de alguma outra equipe ou dispositivo para iniciar sua atividade enquanto desenvolve qualquer produto de dados. Pode haver situações em que a equipe tenha que esperar muito tempo para obter os dados solicitados. Quando o dependente for um dispositivo de software que falhou, o dependente não será notificado.

Normalmente, nesses casos, após o tempo previsto de entrega,

ações necessárias são tomadas para verificar a existência de dependee. Dependências de dados geralmente levam a atrasos na produção.

F. Latência e sobrecarga do pipeline de dados

Os pipelines de dados podem criar latência adicional para todo o fluxo de trabalho de dados. A latência é definida como o tempo que os dados levam para percorrer todo o pipeline. Quando várias equipes estão solicitando dados ou quando o fluxo de dados aumenta, o pipeline de dados pode produzir saídas atrasadas. Além disso, os dados devem passar por todos os componentes e verificações no pipeline para chegar ao destino. A falha ou lentidão de qualquer um desses componentes no trânsito pode levar à latência. O pipeline de dados se torna uma sobrecarga quando os dados são usados para casos em que a qualidade dos dados não é importante.

G. Proprietário do pipeline de dados sobrecarregado

O proprietário do pipeline de dados é uma pessoa responsável por monitorar e gerenciar os dados que fluem pelo pipeline de dados. Durante os horários de pico, essa pessoa fica sobrecarregada.
Além disso, é sempre bom ter o máximo de automação no pipeline de dados.

H. Pipelines de dados não confiáveis

A confiabilidade de um pipeline de dados depende da confiabilidade de seus elementos. Portanto, os elementos em um pipeline de dados devem ser tolerantes a falhas. Um pipeline de dados sem validação integrada, os mecanismos de mitigação não podem garantir a qualidade dos dados.

I. Baixa capacidade de armazenamento

Quando cada equipe está construindo seu próprio pipeline de dados, todos armazenam os mesmos dados em diferentes formas no armazenamento de dados, levando à falta de espaço de armazenamento. Bancos de dados, data warehouses, data lakes são para eliminar o armazenamento redundante de dados. No entanto, um número maior de pipelines de dados acabará levando a uma capacidade de armazenamento reduzida. Outro motivo é a divisão do espaço de armazenamento entre as equipes. Quando o armazenamento é dividido entre equipes, cada um receberá apenas uma pequena parte do espaço de armazenamento real disponível.

VI. META-MODELO DE PIPELINE DE DADOS

A seção acima descreveu os desafios com o gerenciamento de dados encontrados pelos profissionais do setor. A partir das descobertas empíricas e por meio da análise dos pipelines existentes, desenvolvemos um modelo conceitual de pipeline de dados que pode potencialmente superar as limitações dos pipelines de dados existentes.

Metamodelo é um conjunto de conceitos usados para construir pipelines de dados. Nós e conectores são os dois principais componentes básicos usados para construir um pipeline de dados. Os nós são interconectados uns com os outros usando conectores. Ambos os componentes têm certas capacidades. Por exemplo, a capacidade de conectar nós diferentes é a capacidade do conector. A Fig. 5 mostra os componentes, recursos de nós e conectores, notações usadas para representar nós e conectores, etc. Codificação de cores, ícones e diferenças de estilo são usados para representar as variações nos nós e nos dados que fluem pelos nós. Capacidade é a capacidade de um nó de executar uma determinada atividade. Por exemplo,

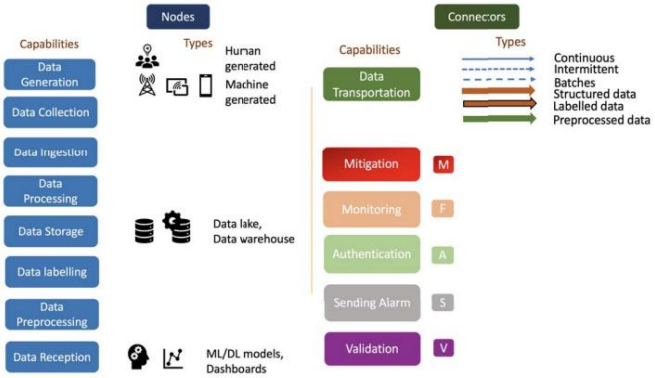

Fig. 5. Metamodelo para construção de pipeline de dados

as fontes de dados no pipeline têm a capacidade de gerar dados.

Capacidades dos nós: geração de dados, coleta de dados, ingestão de dados, armazenamento de dados, processamento de dados, rotulagem de dados, pré-processamento de dados, recepção de dados.

Capacidades dos conectores: Transmissão de dados, autenticação cação, Validação, Monitoramento, Mitigação, Envio de alarme.

A capacidade dos conectores de realizar determinada atividade é denominada capacidade dos conectores. Os conectores têm capacidades diferentes. Os conectores são os portadores de dados. ou seja, eles transmitem dados de um nó para o próximo. Alguns conectores carregam dados brutos e alguns carregam dados processados. Alguns conectores carregam dados rotulados. Cada um desses conectores recebe notação separada e codificação de cores na fig. 5. Todos os conectores têm a capacidade de transmitir dados. Além disso, possuem recursos adicionais como autenticação, validação, monitoramento, mitigação e envio de alarme. A capacidade de autenticação de um conector é indicada pela colocação de um quadrado de cor verde claro com 'A' em cima dele. Da mesma forma, a validação é representada por um quadrado amarelo com 'V' nele. A mitigação é representada por um quadrado vermelho com 'M'. Monitoramento pelo quadrado de cor bege com 'F' nele. Como a letra 'M' já é usada para mitigação, usamos F(Detecção de falha). O quadrado de cor cinza com 'S' denota a capacidade de enviar o alarme.

VII. MODELO CONCEITUAL DE PIPELINES DE DADOS

O pipeline de dados é uma série complexa de componentes interconectados entre si, onde a saída de um componente é alimentada como entrada para o outro. O ponto inicial do pipeline de dados é chamado de origem e o destino final é chamado de coletor. Todos os outros nós são nós intermediários. Cada componente no pipeline de dados manipula dados executando atividades. O modelo de pipeline de dados apresentado nesta seção é um modelo conceitual. De acordo com o requisito, pode ser usado por qualquer organização para qualquer aplicativo de dados, criando instâncias.

A Fig. 6 ilustra o modelo conceitual de um pipeline de dados.

Geração de dados: os pipelines de dados começam a partir de uma fonte e, na maioria dos casos da vida real, a fonte será múltipla e distribuída. Portanto, nosso pipeline também possui várias fontes distribuídas em todo o mundo. As fontes de dados podem ser de

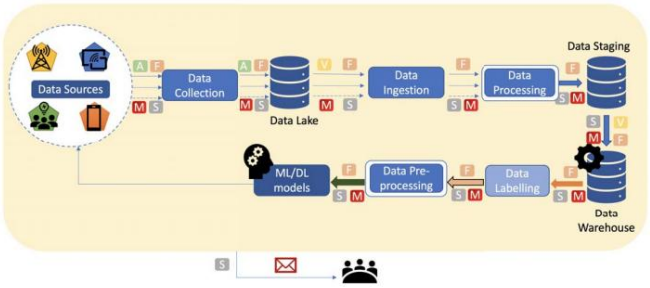

Fig. 6. Modelo conceitual de pipeline de dados

tipos diferentes. Qualquer dispositivo que tenha a capacidade de gerar dados é chamado de fonte de dados. As fontes de dados consideradas aqui são classificadas em duas categorias: geradas por humanos e geradas por máquina. Os dados produzidos por meio de manipulações por membros da equipe, outros membros da equipe ou outras organizações são dados gerados por humanos. Os dados gerados por máquinas são produzidos pelos vários dispositivos empregados em diferentes equipamentos, veículos e assim por diante. Por exemplo, dados gerados por um dispositivo embutido no carro, dados produzidos por aplicativos móveis, etc.

Coleta de dados: O fluxo de dados da fonte pode ser em lote, intermitente ou contínuo. Três estilos diferentes de setas a partir de fontes de dados indicam que o conector entre a fonte de dados e a ingestão de dados pode transportar todas as três variações de fluxo de dados. Embora o nó de coleta de dados possa coletar dados das origens, ele deve mostrar a permissão para coletar dados das origens.

Armazenamento de dados brutos - Data Lake: os dados coletados da fonte serão brutos e devem ser armazenados para que os arquivos de dados originais possam ser recuperados no futuro. No entanto, o nó de coleta de dados deve mostrar seu direito de ingerir dados no pipeline de dados. Essa autenticação é realizada por conectores entre a coleta de dados e o data lake.

Ingestão de dados: arquivos de dados brutos podem ser retirados do data lake e ingeridos no componente de processamento de dados. Esse método de ingestão de dados será diferente para diferentes tipos de dados. Os dados podem chegar em todas as formas e tamanhos. Os dados de fluxo em tempo real serão processados sequencialmente. Os dados contínuos serão validados imediatamente após extraí-los do data lake.

Processamento de dados: o processamento de dados em si é um nó composto no qual pode haver vários componentes individuais, como agregação de dados, análise de dados, transformação de dados, etc. A agregação de dados é o processo pelo qual os dados brutos são expressos em uma forma adequada para análise estatística. Com a transformação de dados, os dados agregados não estruturados são convertidos em um formato estruturado ou formato semiestruturado. Assim, a etapa de processamento de dados converte todos os diferentes tipos de dados em um único formato e os armazena na área de preparação de dados. Isso é representado simbolicamente na fig. 6 com três setas de entrada diferentes para processamento de dados indicando dados contínuos, intermitentes e em lote. A saída da etapa de processamento de dados é um único grosso seta.

Data Staging e Data Warehouse: O data staging é uma área de armazenamento temporário onde os dados podem ser armazenados para validação. Depois de validados os dados estruturados ou semiestruturados, eles são armazenados em um data warehouse. Este data warehouse funciona como um ponto a partir do qual os dados podem ser retirados para vários aplicativos de dados, como criação de relatórios, aplicativos de ML, criação de painéis, etc.

Rotulagem de dados: nosso estudo é focado principalmente em aplicativos de ML. Portanto, o pipeline de dados mostra as etapas necessárias para automatizar o pipeline de dados para aplicativos de ML. Os algoritmos de ML podem ser supervisionados, não supervisionados ou reforçados. Para algoritmos não supervisionados, a etapa de rotulagem de dados pode ser ignorada. Essa é a razão pela qual a etapa de rotulagem de dados é mostrada em uma cor azul clara diferente dos outros nós. Como a maioria das empresas está usando uma abordagem supervisionada para seus aplicativos de ML, o foco está mais no mesmo.

Pré-processamento de dados: para obter um melhor desempenho dos algoritmos de ML, os dados precisam ser pré-processados antes do treinamento. O pré-processamento depende dos profissionais que desenvolvem o aplicativo e também da natureza do problema. No entanto, etapas populares de pré-processamento de dados são avaliação de qualidade de dados, imputação de dados, codificação de dados, amostragem de dados, redução de dimensionalidade, etc. Assim, o pré-processamento de dados pode incluir qualquer processo que transforme dados de forma que possam ser alimentados um algoritmo de aprendizado de máquina.

Recepção de dados: a saída do pré-processamento de dados é fornecida como entrada para os modelos de ML que os utilizam para treinamento, retreinamento, teste e validação. O modelo ML atua como um coletor de dados na fig. 6. Como a maioria das empresas segue a metodologia ágil, os dados serão coletados dos produtos de dados para futuras iterações. Portanto, os dados produzidos por esses aplicativos de ML são novamente coletados pelo nó de coleta de dados e passam pelo pipeline continuamente.

Capacidades dos conectores: Cada conector entre os nós tem a capacidade de enviar dados para qualquer outro nó, monitorar o fluxo de dados, detecção de falhas, verificar estratégias de mitigação associadas quando uma falha é detectada. Caso não haja uma estratégia de mitigação definida, ele tem a capacidade de enviar alarmes para a equipe responsável. As falhas que podem acontecer em cada estágio serão diferentes. Portanto, as estratégias de mitigação também serão diferentes. Da mesma forma, a equipe/pessoa responsável que pode lidar com uma falha específica também será diferente entre si.

Para resumir, o modelo conceitual do pipeline de dados é totalmente automatizado, no qual o monitoramento é realizado em todo o pipeline. O pipeline de dados é tolerante a falhas até certo ponto porque as estratégias de mitigação existem para melhorar os efeitos das falhas. Além disso, as equipes podem solicitar dados de qualquer ponto do pipeline de acordo com suas necessidades.

VIII. ESTUDO DE VALIDAÇÃO

O modelo conceitual de pipeline de dados foi validado por meio de entrevistas com 20 profissionais da indústria de três empresas de diferentes domínios e diferentes níveis de maturidade. Dois dos autores, juntamente com um autor online conduziram o estudo de entrevista para validação. O objetivo da pesquisa foi desenvolver um modelo conceitual de um sistema totalmente automatizado,

pipeline de dados tolerante a falhas e rastreável. A Tabela 2 ilustra os desafios com gerenciamento de dados que podem ser parcialmente ou totalmente resolvidos pelo uso de pipelines de dados.

A disponibilidade de dados pode ser resolvida até certo ponto com o pipeline de dados proposto. No entanto, quando a fonte falha ao gerar dados, os dados não estarão disponíveis. A falha será detectada automaticamente e as estratégias de mitigação correspondentes serão adotadas. Além disso, quando o cliente está relutante em compartilhar dados, é claro que os dados não chegarão ao pipeline. Nesse cenário, o gerente correspondente receberá um alarme e poderá tomar as ações necessárias para melhorar a situação. Os desafios de qualidade de dados não podem ser completamente resolvidos. Se os dados produzidos pela fonte forem de baixa qualidade, não há nenhum mecanismo inerente no pipeline para torná-los de alta qualidade. No entanto, dados de alta qualidade não perderão sua qualidade durante sua transmissão pelo pipeline. A instabilidade do fluxo de dados também pode ser resolvida usando o pipeline de dados proposto com seus mecanismos integrados para controlar o fluxo de dados. Os silos de dados são um fenômeno complicado que não pode ser resolvido pela mera introdução de um pipeline de dados.
Precisa de reorganizações, uma mudança cultural na empresa, etc.
As dependências de dados podem ser completamente resolvidas, pois o próprio pipeline foi projetado para ser totalmente automatizado. A dependência entre as equipes será eliminada e todas as equipes terão uma dependência no pipeline de dados. A latência do pipeline de dados estará lá. Como os componentes no pipeline são mais e todos esses componentes têm inteligência na forma de recursos, a latência surge como um efeito colateral. A sobrecarga do proprietário do pipeline de dados pode ser reduzida, pois a responsabilidade será distribuída entre várias pessoas que entenderam melhor uma parte específica do pipeline de dados. A confiabilidade do pipeline de dados pode ser garantida com as estratégias de mitigação no nível do conector. A falha de um determinado componente afetará o fluxo de dados que será detectado pelo mecanismo de monitoramento e a falha será tratada pelas estratégias de mitigação ou pelo responsável correspondente. A capacidade de armazenamento não pode ser aumentada implementando o pipeline de dados. Conforme discutido anteriormente, ele pode eliminar o armazenamento redundante de dados. No entanto, nenhuma provisão no pipeline de dados pode aumentar a capacidade de armazenamento.

TABELA II

ANÁLISE DE DESAFIOS DE DADOS QUE PODEM SER RESOLVIDOS COM O PIPELINE DE DADOS PROPOSTO

Desafios com o pipeline de dados proposto para gerenciamento de dados	
Disponibilidade de Dados	resolver parcialmente
Qualidade de Dados	resolver parcialmente
Instabilidade do fluxo de dados	resolver completamente
Silos de dados	não pode resolver
Dependências de dados	resolver completamente
Latência do pipeline de dados	não pode resolver
Proprietário do pipeline de dados sobrecarregado	resolver completamente
Pipeline de dados não confiáveis	resolver completamente
Baixa capacidade de armazenamento	não pode resolver

O modelo desenvolvido pelo primeiro autor foi apresentado aos especialistas industriais e seus consensos e discordâncias foram registrados. A seção de validação será estruturada em termos de acordos e sugestões de melhoria.
Os acordos referem-se a situações em que os praticantes concordam e confirmam enquanto as sugestões de melhoria referem-se a situações em que os entrevistados tiveram uma opinião diferente. Algumas pequenas correções foram feitas no modelo e as outras preocupações levantadas são guardadas para o trabalho estendido do modelo de pipeline de dados.

Caso A: Empresa de Manufatura O

modelo conceitual foi apresentado pelo segundo autor e o terceiro autor coletou o feedback do profissional da indústria.

Acordos: O modelo conceitual de pipeline de dados foi identificado como um conceito padrão que pode ser usado por equipes localizadas em diferentes partes do mundo durante a construção de seu pipeline de dados. Com uma arquitetura padrão, pode haver um entendimento comum dos processos. Automação de monitoramento e mitigação são sugestões interessantes e importantes **para**

aprimoramentos adicionais: O monitoramento tem três variações, como monitoramento de desempenho, monitoramento de perfil de dados e monitoramento de condição. O monitoramento de desempenho pode verificar continuamente o desempenho do software no pipeline de dados. O monitoramento de condições pode garantir a integridade do pipeline de dados e o monitoramento do perfil de dados pode garantir que os dados que fluem pelo pipeline de dados atendam aos requisitos de qualidade de dados, detectem falhas e permitam a investigação de problemas de dados, tornando-os rastreáveis.

Caso B: Companhia Automobilística

O modelo conceitual para pipeline de dados foi apresentado pelo primeiro autor e coletamos as concordâncias e discordâncias de todos os profissionais da indústria.

Acordos: os profissionais reconheceram esse modelo conceitual como uma linguagem para comunicação entre profissionais de dados. Quando há uma linguagem comum, é fácil evitar interpretações errôneas. Além disso, todos concordaram sobre o monitoramento espalhado pelo pipeline e a necessidade de diferentes estágios de armazenamento.

Sugestões para Melhorias: A maior discordância foi sobre a nomenclatura dos nós no pipeline de dados. O data warehouse é um termo para representar os dados agregados e bem transformados que são usados para um caso de uso específico. No modelo conceitual, o data warehouse é capaz de armazenar dados para vários aplicativos. A sugestão foi reformular todos esses nomes ambíguos, como preparação de dados e processamento de dados. Outro problema é que o modelo não dá atenção ao algoritmo de reforço. O pipeline de dados não explica quem é o responsável por cada etapa do pipeline. Para aumentar a capacidade de leitura e compreensão para todos na empresa, foi sugerido ter diferentes visualizações de modelo. Por exemplo, um modelo sem termos técnicos, um segundo com muito menos abstração e assim por diante.

Caso C: Empresa de Telecomunicações

A validação interna do pipeline de dados é realizada na empresa de telecomunicações. O primeiro autor apresentou o trabalho e registrou as respostas da equipe.

Acordos: a equipe gostou do modelo conceitual do pipeline de dados, pois todos estavam desenvolvendo seu pipeline para um caso de uso específico. Eles perceberam a necessidade de um modelo de pipeline padrão que pudesse ser seguido por todos na organização. Além disso, alguns deles ficaram satisfeitos com o armazenamento de dados originais em um data lake, pois estavam enfrentando problemas com a disponibilidade de arquivos de dados brutos.

Sugestões para melhorias adicionais: A própria etapa de processamento de dados pode incluir rotulagem e pré-processamento de dados, que são mostrados como etapas separadas no modelo de pipeline de dados proposto. Eles sugeriram que é bom ter armazenamento separado em cada etapa do pipeline de dados. Deve haver uma provisão para parar de enviar alarmes continuamente para a equipe sempre que o problema persistir por mais tempo.

IX. AMEAÇAS À VALIDADE

Este estudo foi baseado em pipelines de dados existentes desenvolvidos por diferentes equipes da mesma empresa localizadas em várias partes do mundo para reduzir o viés de operar com uma única equipe dentro da mesma organização. Para abordar a ameaça de validade interna, um dos autores que possui conhecimento profundo sobre o processamento de dados na empresa foi solicitado a validar os achados. Além disso, os resultados foram validados com outras equipes da empresa que não estavam envolvidas no estudo. Além disso, o estudo foi novamente validado por empresas externas de diferentes domínios.

X. CONCLUSÃO

No futuro imediato, será inexorável que a análise diária não consiga acompanhar o fluxo diário de dados. Junto com o aumento da popularidade dos dados e seus produtos, os desafios associados a eles também aumentaram. Os cientistas de dados e outros profissionais que trabalham com dados gastam uma quantidade considerável de tempo lutando contra esses desafios.

O modelo de pipeline de dados proposto pode resolver ou aliviar vários desafios de gerenciamento de dados com intervenção humana limitada. No entanto, os pipelines de dados precisam ser cuidadosamente projetados para que o fluxo de dados possa ser monitorado, gerenciado e mantido. Portanto, pipelines de dados totalmente automatizados, tolerantes a falhas e rastreáveis estão ganhando importância. O modelo conceitual de pipeline de dados proposto neste artigo possui nós e conectores que executam as atividades no workflow de dados. O modelo conceptual é validado através de um estudo de caso exploratório onde participaram um total de 20 praticantes de três empresas diferentes. Todos concordaram com a necessidade de pipelines de dados na organização, gostaram da estruturação do modelo conceitual e da automatização do monitoramento, alarme e mitigação. Eles também deram algumas sugestões para a melhoria do modelo. Como trabalho futuro, será feita a descrição das estratégias de mitigação em cada etapa, a realização física do modelo conceitual e os resultados serão analisados.

AGRADECIMENTOS

Este trabalho é parcialmente suportado pela Vinnova e pelo Software Center. Os autores também gostariam de expressar sua gratidão por todo o suporte fornecido pela Ericsson.

REFERÊNCIAS

[1] MW Van Alstyne, GG Parker e SP Choudary, “Pipelines, plataformas e as novas regras de estratégia”, Harvard Business Review, vol. 94, nº. 4, pp. 54–62, 2016.

[2] R. Matheus, M. Janssen e D. Maheshwari, “Data science empowering the public: Data-driven dashboards for transparent and accountable decision-making in smart cities,” Government Information Quarterly, p. 101284, 2018.

[3] JG Stadler, K. Donlon, JD Siewert, T. Franken e NE Lewis, “Melhorando a eficiência e a facilidade da análise de saúde por meio do uso de painéis de visualização de dados,” Big Data, vol. 4, não. 2, pp. 129–135, 2016.

[4] G. Gautam e D. Yadav, “Análise de sentimento de dados do Twitter usando abordagens de aprendizado de máquina e análise semântica”, em 2014 Sétima Conferência Internacional sobre Computação Contemporânea (IC3). IEEE, 2014, pp. 437–442.

[5] SMS Tanzil, W. Hoiles e V. Krishnamurthy, “Esquema adaptativo para cache de conteúdo do YouTube em uma rede celular: abordagem de aprendizado de máquina”, IEEE Access, vol. 5, pp. 5870–5881, 2017.

[6] P. Covington, J. Adams e E. Sargin, "Redes neurais profundas para recomendações do youtube", em Anais da 10ª conferência ACM sobre sistemas de recomendação, 2016, pp. 191–198.

[7] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams et al., “Recentes avanços em aprendizado profundo para pesquisa de fala na microsoft”, em 2013 IEEE International Conferência sobre Acústica, Fala e Processamento de Sinais. IEEE, 2013, pp. 8604–8608.

[8] H. Sun, S. Hu, S. McIntosh e Y. Cao, “Classificação de viagem de big data na rede de sensores de táxi e uber da cidade de Nova York,” Journal of Internet Technology, vol. 19, não. 2, pp. 591–598, 2018.

[9] K. Goodhope, J. Koshy, J. Kreps, N. Narkhede, R. Park, J. Rao e VY Ye, "Construindo o pipeline de dados de atividades em tempo real do LinkedIn." Eng. de Dados IEEE Boi., vol. 35, não. 2, pp. 33–45, 2012.

[10] HH Olsson e J. Bosch, “Das opiniões à pesquisa e desenvolvimento de software orientado a dados: um estudo de vários casos sobre como fechar o problema do 'circuito aberto'”, em 2014 40ª Conferência EUROMICRO sobre Engenharia de Software e Aplicativos Avançados. IEEE, 2014, pp. 9–16.

[11] M. Banko e E. Brill, “Scaling to very very large corpora for natural language desambiguation,” in Proceedings of the 39th Annual Meeting on Association for Computation Languages. Association for Computational Linguistics, 2001, pp. 26–33.

[12] E. Deelman e A. Chervenak, "Desafios de gerenciamento de dados de fluxos de trabalho científicos intensivos em dados", em 2008 Oitavo Simpósio Internacional IEEE sobre Cluster Computing and the Grid (CCGRID). IEEE, 2008, pp. 687–692.

[13] P. Vassiliadis, “Uma pesquisa sobre a tecnologia de extração-transformação-carregamento,” Jornal Internacional de Data Warehousing e Mineração (IJDWM), vol. 5, não. 3, pp. 1–27, 2009.

[14] J. Trujillo e S. Lujan-Mora, “Uma abordagem baseada em uml para modelagem de processos etl em data warehouses,” na Conferência Internacional sobre Modelagem Conceitual. Springer, 2003, pp. 307–320.

[15] T. Rabl e H.-A. Jacobsen, “Geração de big data”, em Especificando grandes Benchmarks de Dados. Springer, 2012, pp. 20–27.

[16] S. Amershi, A. Begel, C. Bird, R. DeLine, H. Gall, E. Kamar, N. Na gappan, B. Nushi e T. Zimmermann, “Engenharia de software para aprendizado de máquina: um caso estudo,” em 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). IEEE, 2019, pp. 291–300.

[17] JM Verner, J. Sampson, V. Tosic, NA Bakar e BA Kitchenham, "Diretrizes para estudos de casos múltiplos de base industrial em engenharia de software", em 2009 Terceira Conferência Internacional sobre Desafios de Pesquisa em Ciência da Informação. IEEE, 2009, pp. 313–324.

[18] P. Runeson e M. Host, “Diretrizes para conduzir e relatar pesquisas de estudo de caso em engenharia de software,” Engenharia de software empírica, vol. 14, não. 2, pág. 131, 2009.

[19] J. Singer, SE Sim e TC Lethbridge, “Coleta de dados de engenharia de software para estudos de campo”, no Guia para engenharia de software empírica avançada. Springer, 2008, pp. 9–34.

[20] SH Khandkar, "Codificação aberta", Universidade de Calgary, vol. 23, pág. 2009, 2009.