14º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2023

Contribuições ao compartilhamento de serviços e pipelines em uma plataforma de ciência de dados aplicada à saúde

NOME A. SOBRENOME[1], AUTOR[2], AUTOR[3], AUTOR[4], AUTOR[n]
(Times New Roman, 11, Centralizado)

[1] Graduando em Tecnologia de Análise e Desenvolvimento de Sistemas, Bolsista PIBIFSP, IFSP, Câmpus Capivari, emailautor@ifsp.edu.br. (Times New Roman, 9, Justificado)
[2]
[3]
[n]

Área de conhecimento (Tabela CNPq): 1.03.03.04-9 Sistemas de Informação

RESUMO: Mais de 300 mil pessoas morreram de câncer no Brasil só em 2020. Nesse contexto, a Ciência de Dados têm combinado competências de profissionais de áreas como Computação, Estatística e Saúde no estudo do genoma, suas características e correlações com o surgimento de neoplasias malignas. Atualmente, há algoritmos de ciência de dados disponíveis na literatura que poderiam ser reutilizados e aprimorados para diferentes tipos de estudos sobre câncer. Entretanto, o entendimento e a configuração do ambiente para a execução desses algoritmos é desafiador, em especial, para pesquisadores de áreas fora da computação. Por isso, foi iniciada a construção de uma plataforma para execução de algoritmos de ciência de dados aplicados à saúde na forma de serviços. Essa plataforma viabiliza a criação de *pipelines* de ciência de dados aplicados à saúde de forma livre de configuração e amigável a pesquisadores em geral. Este artigo visa promover as novas funcionalidades que permitem a gestão do compartilhamento dos serviços e pipelines entre diferentes usuários, além da otimização das funcionalidades já existentes.

PALAVRAS-CHAVE: Ciência de Dados; Saúde; Câncer; Serviços; Microsserviços

ABSTRACT: Over 300 thousand people died of cancer in Brazil in 2020 alone. In this context, Data Science has been combining competences of professionals from fields such as Computing, Statistics and Health in the study of the genome, its characteristics and its correlations with the emergence of malignant neoplasms. Currently, there are data science algorithms available in literature that could be reutilized and enhanced for different types of studies on cancer. However, understanding and configuring the environment for executing these algorithms is challenging, especially, for researchers from fields outside computing. Hence, the construction of a platform for executing health applied data science algorithms in the form of services was started. This platform enables a configuration-free and friendly way of creating health applied data science pipelines for researchers in general. This article aims to promote the new functionalities that enable the management of the sharing of services and pipelines between users, in addition to optimizing existing functionalities.

KEYWORDS: Data Science; Health; Cancer; Services; Microservices

INTRODUÇÃO

Câncer é um termo que engloba um conjunto de mais de uma centena de doenças malignas que têm em comum o crescimento desordenado das células. Essas doenças surgem a partir de mutações genéticas, nas quais o DNA da célula passa a receber instruções erradas para suas atividades,

transformando células normais em células cancerosas[1]. Na literatura, já foram identificadas uma grande variedade de genes e características dos indivíduos que, isoladamente ou combinadas, podem indicar uma maior propensão ao desenvolvimento de determinados tipos de câncer. Identificar e compreender essas informações relevantes em grandes volumes de dados clínicos existentes tem sido uma tarefa desafiadora nessa área de pesquisa.

Nesse contexto, a Ciência de Dados é uma área multidisciplinar que cresceu substancialmente em interesse nos últimos anos, por permitir extrair conhecimento a partir de grandes volumes de dados de forma ágil e eficiente (DHAR, 2013). Na Ciência de Dados, são estudados diferentes princípios, métodos e técnicas primordialmente fundamentadas na Computação, na Matemática e na Estatística com o objetivo de encontrar padrões a partir de conjuntos de dados provenientes de diferentes domínios, como a Saúde. O processo de obtenção de conhecimento em Ciência de Dados envolve diferentes etapas, que vão desde a preparação dos dados, limpeza, exploração, criações de modelos até a interpretação dos resultados (GRUS, 2015).

Muitos dos algoritmos de ciência de dados são construídos a partir de um amplo conjunto de bibliotecas, que possuem requisitos de instalação, configuração e compatibilidade entre si. Dentre essas bibliotecas é possível citar a SciPy[2], o Pandas[3] e a Matplotlib[4]. Embora esteja disponível na literatura uma grande quantidade de estudos envolvendo algoritmos de Ciência de Dados para a área da Saúde, muitas vezes a tarefa de reprodução ou reúso de tais estudos por outros pesquisadores é complexa e envolve diferentes problemas de configuração. Isso se torna um fator complicador para a condução de novas pesquisas, principalmente por investigadores de fora da área da Computação, como biomédicos, médicos e biólogos e estatísticos.

Uma forma de se contornar esse problema é a disponibilização de rotinas de ciência de dados na forma de serviços coesos e passíveis de composição. O conceito de software como serviço distribuído teve início com a Arquitetura Orientada a Serviços (do inglês, *Service-Oriented Architecture*), que possibilitou a integração de sistemas de software heterogêneos a partir do uso de interfaces independentes de linguagem de programação, oferecidas a partir de protocolos padrão (JOSUTTIS, 2007). Mais recentemente, tem sido amplamente adotado outro estilo arquitetural apoiado sobre o conceito de serviços, chamado REST (do inglês, *Representational State Transfer*) (FIELDING, R., 2000), que faz uso de protocolos HTTP amplamente conhecidos para fornecer recursos de software distribuídos, executados em diferentes plataformas, de forma simples e produtiva. Por se tratar de uma forma mais "leve" de fornecer software como serviço, o REST vem viabilizando a criação de sistemas altamente modulares construídos na forma de microsserviços. Um microsserviço é uma pequena porção coesa de software executada de forma independente, que pode ser reutilizada na composição de processos de negócio maiores e mais complexos, formando sistemas flexíveis, fáceis de escalar e de se evoluir (PAUTASSO, 2017).

Para investigar a hipótese de que serviços podem contribuir para simplificar a execução de algoritmos de ciência de dados aplicados à saúde, foi proposta a PipeGene, uma plataforma para o cadastramento, compartilhamento, composição e execução de algoritmos de ciência de dados como serviços. Nessa plataforma, algoritmos já configurados em suas máquinas de execução remotas, disponibilizados como serviços, são inseridos para serem reutilizados em diferentes contextos da ciência de dados aplicada à saúde. Por se tratar de um projeto complexo, o desenvolvimento da plataforma foi dividido em diversas iterações, sendo que na primeira delas foi disponibilizada como uma prova de conceito. Embora funcional, a primeira versão possuía limitações que precisavam ser aperfeiçoadas no sentido de fornecer a plataforma para teste junto a pesquisadores de ciência de dados aplicada à saúde. Dessa forma, o objetivo do presente trabalho é contribuir para a consolidação de

[1] Instituto Nacional do Câncer (INCA). Ministério da Saúde. https://www.inca.gov.br/como-surge-o-cancer. Acesso em 15 de agosto de 2023.
[2] SciPy: https://scipy.org. Acesso em 15 de agosto de 2023.
[3] Pandas: https://pandas.pydata.org. Acesso em 15 de agosto de 2023.
[4] Matplotlib: https://matplotlib.org. Acesso em 15 de agosto de 2023.

funcionalidades já existentes e a implementação de novas funcionalidades na PipeGene. permitindo a gestão de projetos e o compartilhamento de recursos (serviços e processos de execução) entre diferentes projetos e usuários.

MATERIAL E MÉTODOS

A PipeGene é disponibilizada como uma plataforma Web, sendo a interface com o usuário (*front-end*) desenvolvida com o *framework* Angular e a parte do servidor (*back-end*) implementada em Java, usando o *framework* Spring Boot[5]. A Figura 1 ilustra a organização geral da ferramenta. No desenvolvimento do projeto foi adotado um design baseado na arquitetura hexagonal e também em princípios SOLID, de modo a prover uma melhor modularidade e flexibilidade a partir do isolamento das regras de domínio da aplicação, isolando dependências externas como persistência e chamadas a APIs de terceiros. Ao longo do projeto foram desenvolvidos diversos testes usando Postman para emular requisições feitas pelo *front-end* ao *back-end*.

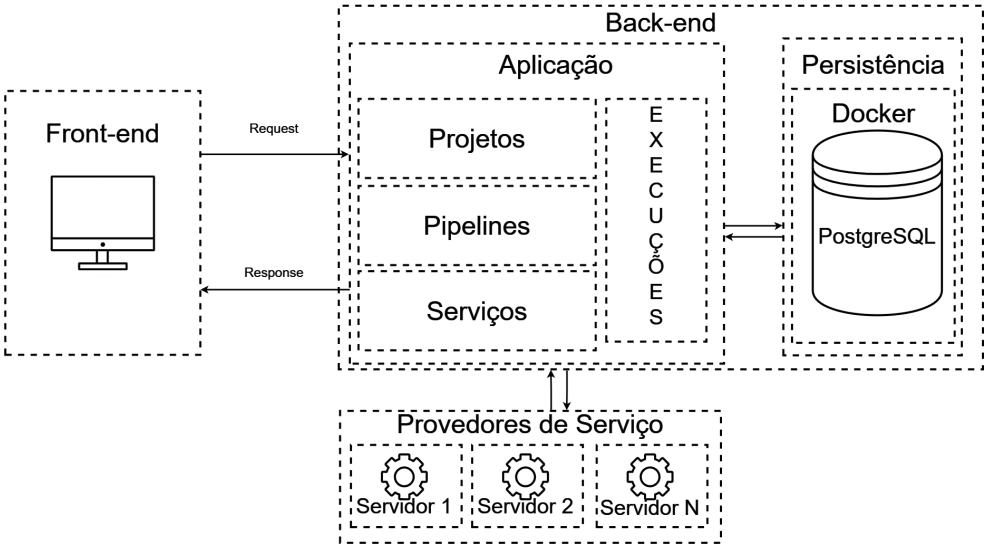

Figura 1. Diagrama de Fluxo da Plataforma PipeGene. Fonte: Autores

No *front-end*, usuários da ferramenta autenticados podem criar projetos de ciência de dados, cadastrar serviços de ciência de dados, montar *pipelines* de execução com os serviços já cadastrados e executar tais *pipelines* a partir de conjuntos de dados (*datasets*) de câncer. As funcionalidades selecionadas por usuários no *front-end* são enviadas ao *back-end* e processadas por *endpoints* fornecidos com Spring Boot, para então serem gerenciadas e armazenadas em um banco de dados PostgreSQL[6] executado em um container Docker[7]. Sempre que o usuário realiza uma solicitação de execução de um *pipeline* a partir de um *dataset*, a PipeGene orquestra o fluxo de execução com chamadas aos microsserviços disponibilizados em servidores externos e, ao final da execução, disponibiliza um arquivo para *download* contendo o resultado. Um exemplo de uso da plataforma PipeGene é ilustrado na captura de tela da Figura 2, na qual usuários podem realizar execuções de *datasets* e observar o andamento do fluxo do *pipeline*, serviço a serviço. Os três estados possíveis de uma execução são sucesso (no qual o arquivo do resultado é disponibilizado para *download*), falha e em execução.

[5] Spring Boot. https://spring.io/. Acesso em 17 de agosto de 2023.
[6] PostgreSQL. https://www.postgresql.org/. Acesso em 17 de agosto de 2023.
[7] Docker. https://www.docker.com/. Acesso em 17 de agosto de 2023.

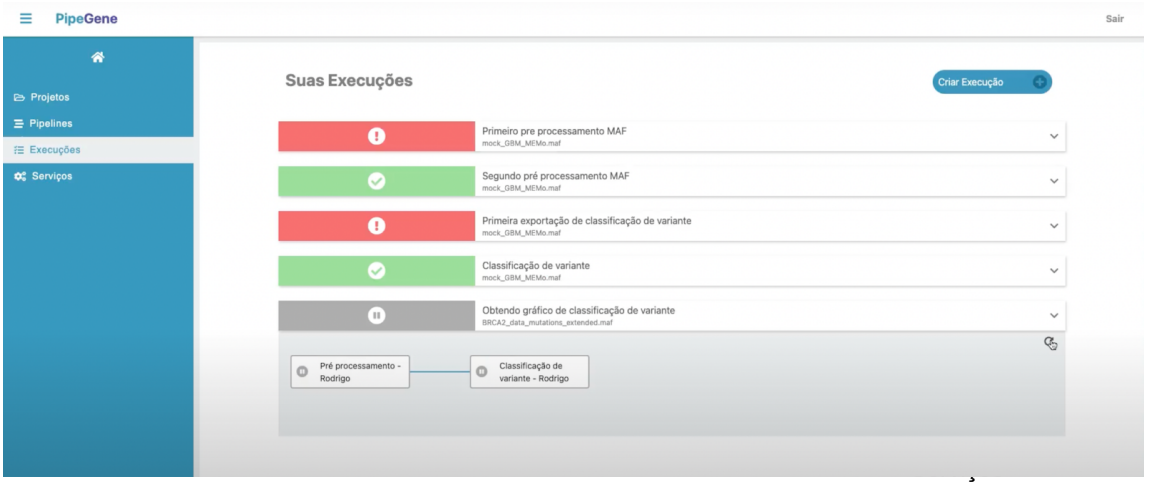

Figura 2. Captura da tela de execução de pipelines de serviços. Fonte: Autores

RESULTADOS E DISCUSSÃO

As funcionalidades projetadas e desenvolvidas durante a etapa do projeto descrita neste trabalho envolvem melhorias na gestão do ciclo de vida de projetos, *pipelines* e serviços, bem como a inclusão de funcionalidades para trabalho multi-usuário em projetos e a gestão do compartilhamento de serviços e *pipelines* com diferentes níveis de acesso. Até o momento, todas as funcionalidades foram desenvolvidas no *back-end*, sendo que as interfaces gráficas ainda serão elaboradas na segunda metade do projeto.

Com relação às melhorias das funcionalidades já existentes, foram adicionadas funcionalidades para visualização, edição e remoção de serviços e *pipelines*. Na versão anterior, só era possível criar serviços e *pipelines*, não sendo permitido editar informações preenchidas de maneira equivocada ou atualizar entradas do sistema com informações mais atuais. Também não era permitido remover informações, o que inviabilizava o uso da plataforma em situações reais, envolvendo pesquisadores de ciência de dados.

Adicionalmente, foi criada uma funcionalidade para importação de *pipelines* em um mesmo projeto, o que permite a criação de variações de fluxos de execução de maneira mais produtiva. Por exemplo, ao importar uma *pipeline* com dezenas de serviços já organizados, um pesquisador pode substituir uma funcionalidade para seleção de um determinado gene, selecionando outro, e gerar gráficos para comparação do resultado das duas execuções com um mesmo *dataset*. Anteriormente, para chegar a mesma comparação, um pesquisador deveria reproduzir o processo de criação em uma nova *pipeline*, mas incluindo novamente todos os serviços que não precisaria alterar.

Além da possibilidade de importação de *pipelines*, foram criadas funcionalidades para a gestão de grupos de pesquisadores em um mesmo projeto, permitindo o compartilhamento de *pipelines*, serviços e seus resultados de forma integrada, sendo possível o criador do projeto enviar solicitações a outros pesquisadores da plataforma ou removê-los do grupo, reduzindo a dependência de serviços externos e promovendo a reutilização de serviços já existentes, economizando recursos e tempo. Outrora, na plataforma não era possível que pesquisadores estivessem em um mesmo projeto, sendo necessário que cada pesquisador criasse seu projeto mesmo que seus serviços e *pipelines* sejam os mesmos que os de outros projetos da plataforma.

Com a adição de grupos e compartilhamento, foi necessário a criação de níveis de acesso para os diferentes elementos do sistema, que possibilita o compartilhamento de informações apenas com projetos e usuários desejados, sendo possível evitar um custo por processamento em nuvem causado

pela execução de serviços por terceiros. Na versão passada da plataforma, um serviço poderia ser público a todos ou somente ao usuário que o inseriu.

Exemplificando as novas funcionalidades: temos um pesquisador que é especialista em genética e desenvolveu um serviço para a seleção de genes de câncer de mama. Um outro pesquisador especialista em estatística trabalhou na criação de correções e na plotagem de estatística descritiva de câncer de mama. Com isso, ambos os pesquisadores podem utilizar seus serviços de forma complementar em um mesmo projeto, compartilhando resultados, serviços e *pipelines*. Anteriormente estes procedimentos não eram possíveis.

CONCLUSÕES

Este trabalho aborda a mais recente etapa de desenvolvimento da PipeGene, uma plataforma que visa facilitar a execução de algoritmos de ciência de dados aplicados à saúde como serviços passíveis de composição, sem necessidade de configuração e amigável a pesquisadores das diversas áreas de conhecimento envolvidas no estudo do câncer. Foram implementadas novas funcionalidades para a gestão de grupos, compartilhamento de projetos e serviços, bem como trabalho colaborativo. Também foram aprimoradas funcionalidades já existentes, em especial, para permitir a visualização, edição e remoção de *pipelines* e serviços. No atual estado desta etapa, foi dada ênfase ao desenvolvimento das APIs, regras de negócio e regras de persistência no *back-end*, além do teste de tais funcionalidades. O trabalho continuará com ênfase na integração do *framework* Angular, buscando a implementação das novas funcionalidades à plataforma. Também são esperados ajustes nas funcionalidades já executadas, para melhorar ainda mais o desempenho do sistema. Por fim, serão realizadas reuniões para avaliação e validação do sistema junto a potenciais usuários. As melhorias desenvolvidas na plataforma visam permitir interações mais eficazes entre pesquisadores, facilitando o compartilhamento de recursos e proporcionando uma melhor experiência de uso, contribuindo, assim, para o desenvolvimento de pesquisas envolvendo ciência de dados aplicada à saúde.

CONTRIBUIÇÕES DOS AUTORES

Apresente de forma simplificada as contribuições de cada autor. Esta seção é baseada na Taxonomia CRediT e visa descrever as contribuições dos autores no trabalho. Como sugestão, utilize o Guia para Marcação e Publicação de contribuição de autores: Taxonomia CRediT da Scielo.

Exemplo: M.F.C, E.B.M.S e K.L.G. (podem ser utilizadas as iniciais do nome) contribuíram com a curadoria e análise dos dados. H.F.S e M.F.C procederam com a metodologia e experimentos. M.S.L, K.L.G. e E.B.M.S atuaram na redação do trabalho.

Todos os autores contribuíram com a revisão do trabalho e aprovaram a versão submetida.

REFERÊNCIAS

DHAR, V.. Data science and prediction. **Communications of the ACM**, v. 56, n. 12, p. 64-73, 2013.
GRUS, J. **Data Science from Scratch:** First Principles with Python. Sebastopol, CA: O'Reilly,
JOSUTTIS, N. M. **SOA in practice:** The art of distributed systems design. Sebastopol, CA: O'Reilly, 2007.
FIELDING, R. **Architectural styles and the design of network-based software architectures.** 2000. Tese (Doutorado). University of California, Irvine, California, USA, 2000.
PAUTASSO, Cesare et al. Microservices in practice, part 1: Reality check and service design. **IEEE software**, v. 34, n. 01, p. 91-98, 2017.