Análise Comparativa de Repositórios de Software Open-Source Pré e Pós-Revivência

Gabriel Ramos Ferreira, João Pedro Silva Braga

30 de outubro de 2025

1 Introdução

O ecossistema de software de código aberto (Open-Source Software - OSS) representa um pilar fundamental no desenvolvimento de tecnologia moderna. Projetos OSS impulsionam a inovação, permitem a colaboração em escala global e fornecem a base para inúmeras aplicações comerciais e acadêmicas. No entanto, a sustentabilidade desses projetos é um desafio constante. Muitos repositórios passam por períodos de baixa ou nenhuma atividade, um fenômeno frequentemente referido como a "morte"de um projeto. Curiosamente, alguns desses projetos conseguem ser "revividos"ou "ressuscitados", retomando o desenvolvimento ativo após um período de dormência.

Este fenômeno de revivência é de grande interesse para a engenharia de software, pois compreendê-lo pode fornecer insights valiosos sobre a saúde, a resiliência e as melhores práticas de gestão de comunidades OSS. O que diferencia um projeto que consegue se recuperar de um que permanece inativo permanentemente? Quais são as mudanças nas práticas de desenvolvimento e colaboração que caracterizam essa transição?

1.1 Problema de Pesquisa

A transição de um estado de inatividade para um de desenvolvimento ativo em repositórios OSS não é um processo bem compreendido. Faltam estudos empíricos que caracterizem as mudanças nas atividades técnicas e colaborativas que ocorrem durante esse processo de revivência. A análise de métricas relacionadas a Pull Requests (PRs), Issues e Commits pode revelar padrões de comportamento que estão associados a uma recuperação bem-sucedida, oferecendo um guia para comunidades que buscam revitalizar projetos estagnados.

Este trabalho busca preencher essa lacuna, investigando as diferenças quantitativas e qualitativas nas práticas de desenvolvimento de software em um conjunto de repositórios OSS antes do período de inatividade e após sua revivência.

1.2 Objetivo Geral

Analisar e comparar as características das atividades técnicas e colaborativas em repositórios de software de código aberto, contrastando o período que antecede a sua inatividade ("pré-morte") com o período que sucede a sua retomada de atividades ("pós-revivência").

1.3 Objetivos Específicos

- Caracterizar e comparar a qualidade dos artefatos técnicos, como Pull Requests, Issues e Commits, entre os períodos pré e pós-revivência.

- Investigar as mudanças na dinâmica de contribuição da comunidade, analisando a atividade dos contribuidores e a distribuição do trabalho após a revivência do repositório.
- Avaliar se o comportamento relacionado à revisão de código e ao feedback técnico se altera significativamente após a retomada das atividades do projeto.

1.4 Questões de Pesquisa e Hipóteses

Para guiar nossa investigação, formulamos as seguintes questões de pesquisa e suas respectivas hipóteses nulas (H0) e alternativas (H1).

RQ1: Quais características nas atividades de Pull Requests, Issues e Commits estão associadas à revivência e manutenção de repositórios OSS?

- **Hipótese Nula ($H0_1$):** Não há diferença estatisticamente significativa na qualidade das métricas de Pull Requests (taxa de aceitação, tempo de revisão), Issues (tempo de fechamento) e Commits (taxa de adesão a convenções) entre os períodos pré e pós-revivência.
- **Hipótese Alternativa ($H1_1$):** O período pós-revivência apresenta melhorias estatisticamente significativas na qualidade das métricas de Pull Requests, Issues e Commits, como maior taxa de aceitação de PRs, menor tempo para fechamento de Issues e maior adesão a conventional commits, quando comparado ao período pré-revivência.

RQ2: Como a dinâmica de contribuição muda após a revivência do repositório?

- **Hipótese Nula ($H0_2$):** A dinâmica de contribuição, incluindo o número de contribuidores ativos, a frequência de commits por desenvolvedor e a distribuição de autoria, não apresenta mudança estatisticamente significativa entre os períodos pré e pós-revivência.
- **Hipótese Alternativa ($H1_2$):** A dinâmica de contribuição se torna mais ativa e distribuída no período pós-revivência, com um aumento estatisticamente significativo no número de contribuidores, na frequência de commits e uma colaboração menos centralizada.

RQ3: O comportamento de revisão e feedback técnico se altera após a revivência?

- **Hipótese Nula ($H0_3$):** Não há diferença estatisticamente significativa nos indicadores de revisão e feedback (média de comentários por PR, tempo até o primeiro feedback, percentual de revisões formais) entre os períodos pré e pós-revivência.
- **Hipótese Alternativa ($H1_3$):** O processo de revisão e feedback se torna mais robusto e responsivo no período pós-revivência, apresentando um aumento estatisticamente significativo na quantidade de comentários por PR, uma redução no tempo para o primeiro feedback e um maior percentual de revisões formais.

2 Metodologia

Para responder às questões de pesquisa propostas, este estudo adota uma abordagem de pesquisa empírica quantitativa, baseada na mineração de repositórios de software (Mining Software Repositories - MSR). A metodologia segue os passos de um estudo observacional, onde analisamos dados históricos para identificar padrões e correlações sem manipular diretamente as variáveis, conforme o ciclo da experimentação empírica. O fluxograma da nossa metodologia de pesquisa é apresentado na Figura 1.

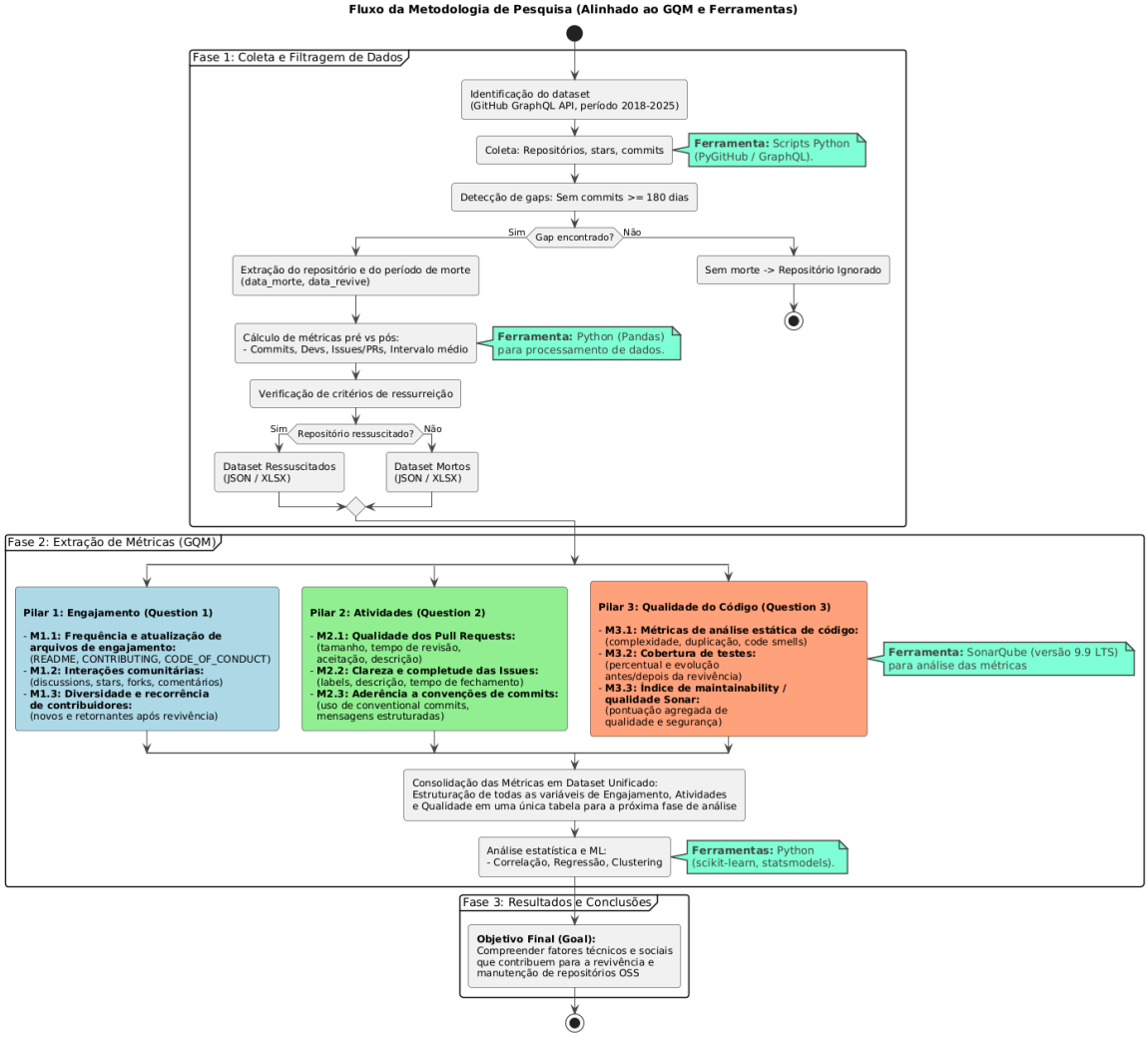

Figura 1: Fluxograma da metodologia de pesquisa adotada.

2.1 Seleção do Dataset e Coleta de Dados

O conjunto de dados utilizado nesta pesquisa foi curado a partir de repositórios de software de código aberto hospedados no GitHub. Os repositórios foram selecionados por apresentarem um padrão claro de "morte"(período de inatividade significativa) seguido por uma "revivência"(retomada do desenvolvimento ativo). A coleta de dados foi automatizada por meio de scripts que acessaram a API do GitHub para extrair métricas relacionadas a Pull Requests, Issues e Commits.

As variáveis coletadas foram divididas em dois períodos para cada repositório, permitindo uma análise comparativa. As métricas centrais para este estudo estão descritas na Tabela 1.

Tabela 1: Descrição das variáveis coletadas no dataset.

Variável	Descrição
`repo_name`	Nome do repositório no GitHub.
`period`	Período da análise ('pre_death' ou 'post_revive'). Variável independente.
`pr_count`	Número total de Pull Requests abertos no período.
`pr_merged_rate`	Taxa de Pull Requests que foram aceitos (merged).
`pr_avg_time_to_merge_days`	Tempo médio, em dias, para um Pull Request ser aceito.
`issue_count`	Número total de Issues abertas no período.
`issue_avg_time_to_close_days`	Tempo médio, em dias, para uma Issue ser fechada.
`commit_count`	Número total de commits no período.
`conventional_commit_rate`	Taxa de commits que seguem a especificação de *Conventional Commits*.

2.2 Procedimentos de Análise

A análise dos dados será conduzida em duas frentes, conforme especificado no roteiro do laboratório (LAB04): visualização de dados e análise estatística.

1. Visualização de Dados: Utilizaremos uma ferramenta de Business Intelligence (BI), como Microsoft Power BI, Tableau ou Google Data Studio, para criar dashboards interativos. Esses dashboards terão duas seções principais:

- **Caracterização do Dataset:** Apresentação das características gerais dos repositórios analisados, utilizando gráficos para sumarizar métricas como contagem total de PRs, Issues e Commits.

- **Análise das Questões de Pesquisa:** Para cada RQ, serão criadas visualizações comparativas (e.g., gráficos de barras, box plots) que contrastam as métricas do período "pré-morte"com o "pós-revivência". O objetivo é "contar a história"dos dados de forma clara e objetiva.

2. Análise Estatística: Para testar formalmente nossas hipóteses, aplicaremos testes de significância estatística. Dado que estamos comparando duas medições do mesmo grupo de repositórios (pré vs. pós), o teste mais apropriado é o teste de Wilcoxon para amostras pareadas, que não assume uma distribuição normal dos dados, uma característica comum em métricas de software. A análise buscará identificar se as diferenças observadas nas visualizações são estatisticamente significativas.

A Tabela 2 mapeia cada questão de pesquisa às métricas utilizadas para respondê-la.

2.3 Ameaças à Validade

Conforme discutido nos conceitos de experimentação, é crucial reconhecer as possíveis ameaças à validade deste estudo.

- **Validade de Construto:** As métricas utilizadas são proxies para conceitos mais abstratos como "qualidade"ou "dinâmica de contribuição". Por exemplo, uma alta taxa de *conventional commits* sugere disciplina, mas não garante a qualidade intrínseca do código.

Tabela 2: Mapeamento das Questões de Pesquisa para as Métricas do Dataset.

RQ	**Métricas Associadas**
RQ1: Características de PRs, Issues e Commits	`pr_merged_rate`, `pr_avg_time_to_merge_days`, `issue_avg_time_to_close_days`, `conventional_commit_rate`.
RQ2: Dinâmica de contribuição	`pr_count`, `issue_count`, `commit_count`. (Métricas proxy para atividade geral).
RQ3: Comportamento de revisão	`pr_avg_time_to_merge_days`. (Métrica proxy para responsividade e eficiência do processo de revisão).

- **Validade Interna:** A causalidade não pode ser garantida. Mudanças observadas após a revivência podem ser devidas a fatores externos não controlados, como a entrada de um novo mantenedor influente ou uma mudança na popularidade da tecnologia subjacente.
- **Validade Externa:** O conjunto de dados, embora representativo do fenômeno, pode não ser grande o suficiente para generalizar os resultados para todo o universo de projetos OSS. Os achados podem ser específicos para o perfil dos repositórios analisados.

A abordagem de análise e a discussão dos resultados levarão em conta essas limitações para garantir uma interpretação cautelosa e robusta dos achados.

3 Resultados

Nesta seção, apresentamos os resultados da análise comparativa dos 25 repositórios de software, contrastando os períodos pré e pós-revivência. Os achados são derivados das visualizações e métricas agregadas geradas no dashboard de Business Intelligence, que sumariza os dados coletados. Os resultados estão organizados de acordo com as questões de pesquisa estabelecidas.

Uma visão geral dos Key Performance Indicators (KPIs) é apresentada na Figura 2, que destaca a magnitude das mudanças entre os dois períodos analisados.

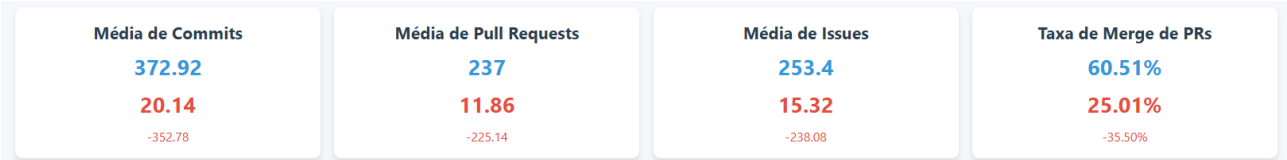

Figura 2: Resumo dos KPIs: comparação das médias de atividade entre os períodos.

3.1 RQ1: Características de PRs, Issues e Commits

A primeira questão de pesquisa investiga se as características de qualidade dos artefatos de desenvolvimento (Pull Requests, Issues e Commits) se alteram após a revivência.

Primeiramente, analisamos a aderência à especificação de *Conventional Commits*. A Figura 3 mostra que houve um aumento substancial na adoção dessa prática. A taxa média de commits convencionais mais do que dobrou, passando de **10,38%** no período pré-morte para **23,58%** no período pós-revivência. Isso sugere uma maior preocupação com a padronização e a clareza do histórico de contribuições após a retomada das atividades.

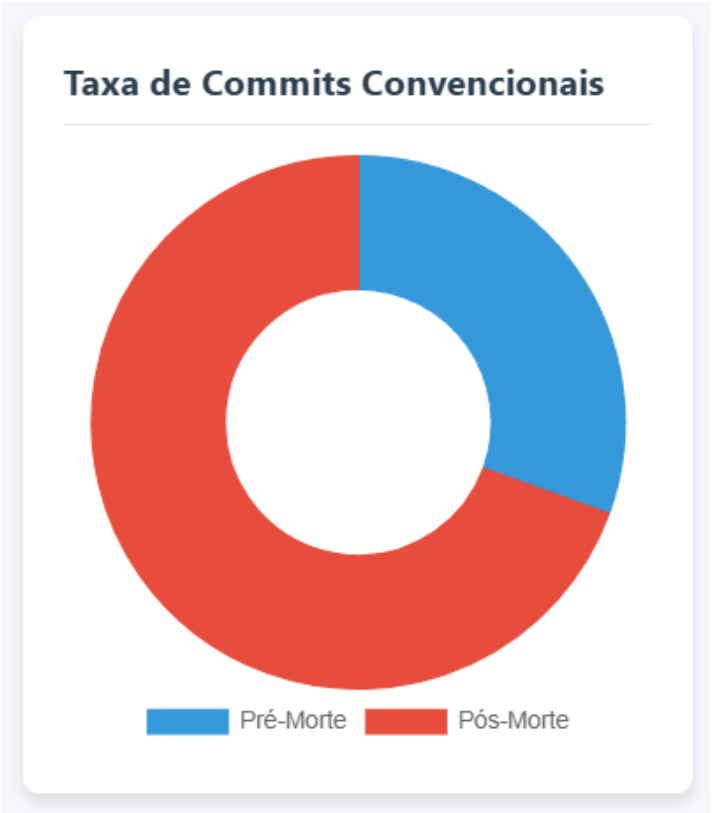

Figura 3: Comparação da taxa média de Commits Convencionais.

Em seguida, avaliamos a eficiência na gestão de Issues e Pull Requests. A Figura 4 compara o tempo médio, em dias, para o fechamento de Issues e para o merge de PRs. Observa-se uma melhoria no tempo de fechamento de Issues, que caiu de uma média de 86 dias para 68,8 dias. No entanto, o tempo médio para merge de PRs aumentou drasticamente, passando de **12,68 dias** para **44,01 dias**.

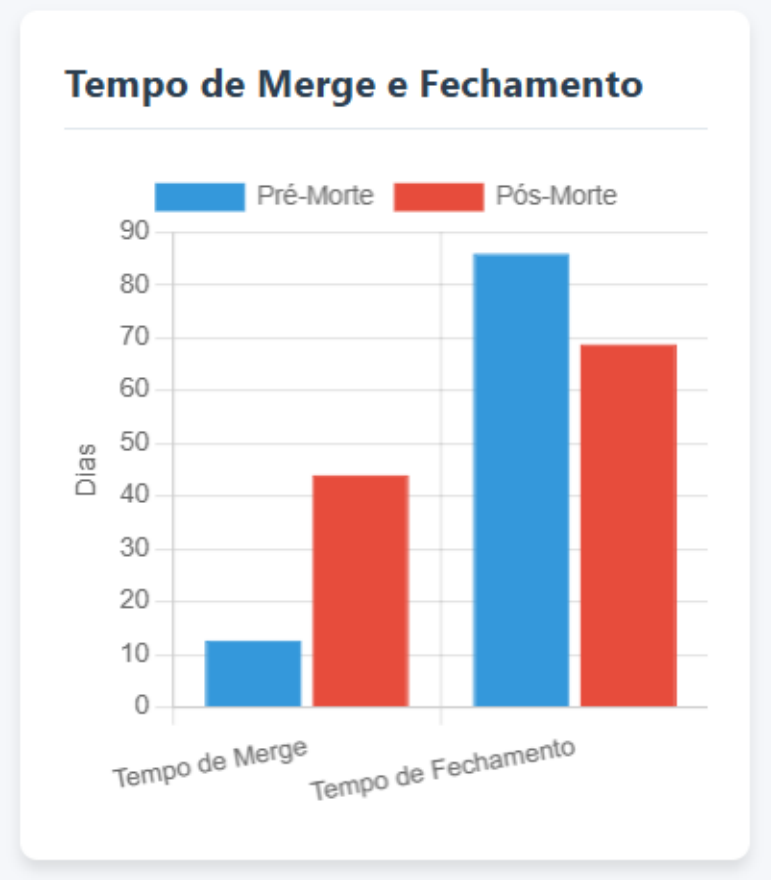

Figura 4: Tempo médio (em dias) para Merge de PRs e Fechamento de Issues.

Complementarmente, os dados da Tabela Detalhada (Figura 7) revelam que a taxa média de merge de PRs também diminuiu significativamente, caindo de **60,51%** para **25,01%**. Juntos, esses resultados indicam um processo de revisão de PRs mais lento e com menor taxa de aceitação no período pós-revivência.

3.2 RQ2: Dinâmica de Contribuição

A segunda questão de pesquisa busca entender como a dinâmica de contribuição muda após a revivência do repositório. As métricas de volume de atividade, como a contagem de commits, PRs e issues, servem como um proxy para a intensidade da colaboração.

Os resultados, sumarizados nas Figuras 2 e 5, apontam para uma redução drástica no volume de atividade em todas as frentes. O número médio de commits por repositório despencou de **372,92** no período pré-morte para apenas **20,14** no período pós-revivência, representando uma queda de mais de 94%.

Figura 5: Queda acentuada na média de commits por repositório.

A mesma tendência é observada para Pull Requests e Issues, conforme ilustrado na Figura 6. A média de PRs abertos caiu de **237** para **11,86**, e a média de Issues de **253,4** para **15,32**. Esses dados refutam a hipótese alternativa de que a dinâmica de contribuição se tornaria mais ativa. Pelo contrário, indicam que a "revivência"ocorre em uma escala de atividade muito menor, possivelmente impulsionada por um núcleo reduzido de mantenedores e contribuidores.

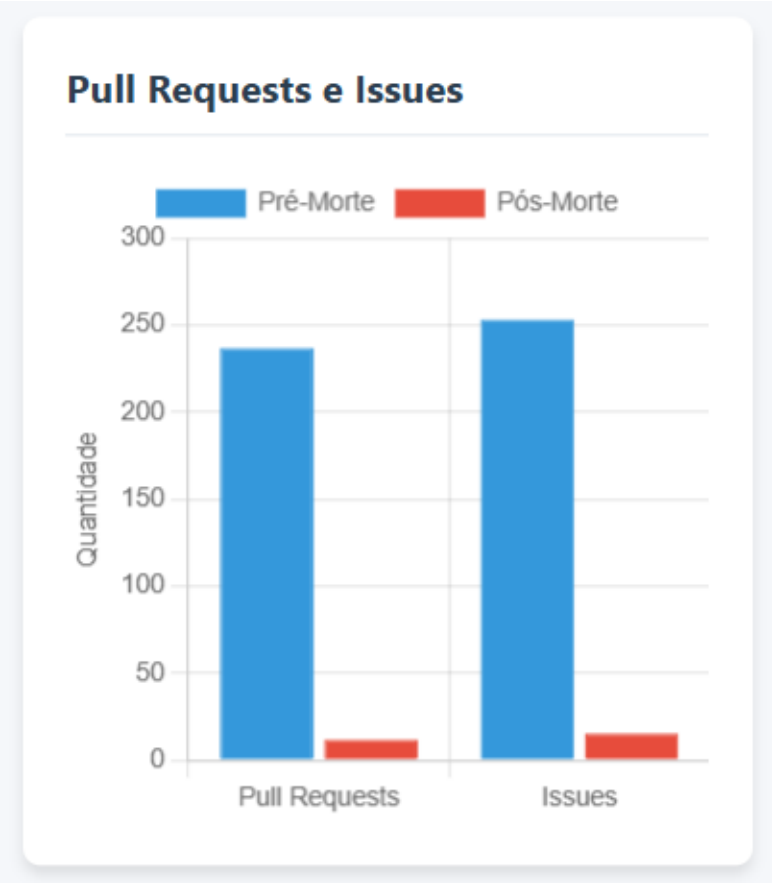

Figura 6: Comparativo do volume médio de Pull Requests e Issues.

3.3 RQ3: Comportamento de Revisão e Feedback Técnico

A terceira questão de pesquisa avalia se o comportamento de revisão e feedback técnico se altera após a revivência. Utilizando o tempo médio para merge de PRs como um indicador da responsividade e eficiência do processo de revisão, os resultados indicam uma mudança negativa.

Conforme já demonstrado na Figura 4 e detalhado na Figura 7, o tempo médio para integrar uma contribuição via Pull Request aumentou em mais de **31 dias** (de 12,68 para 44,01 dias). Este aumento substancial sugere que, embora o volume de PRs seja muito menor, o processo para revisá-los e aceitá-los tornou-se consideravelmente mais lento.

Este achado, combinado com a queda na taxa de merge (de 60,51% para 25,01%), contraria a hipótese de que o processo de revisão se tornaria mais robusto e responsivo. As evidências apontam para a existência de gargalos no processo de revisão pós-revivência, possivelmente devido a um número menor de revisores ativos ou a um processo de avaliação mais rigoroso e demorado para as poucas contribuições recebidas.

Comparativo Detalhado de Métricas

Métrica	Pré-Morte	Pós-Morte	Diferença (Δ)
Média de Commits	372.92	20.14	-352.78
Média de Pull Requests	237	11.86	-225.14
Média de Issues	253.4	15.32	-238.08
Taxa Média de Merge de PRs	60.51%	25.01%	-35.50%
Tempo Médio de Merge (dias)	12.68	44.01	+31.34
Tempo Médio de Fechamento de Issues (dias)	85.99	68.80	-17.19
Taxa Média de Commits Convencionais	10.38%	23.58%	+13.19%

Figura 7: Tabela comparativa detalhada das métricas agregadas.

4 Conclusão

Nesta seção, interpretamos os resultados apresentados, discutimos suas implicações em relação às hipóteses formuladas e ao contexto da literatura existente. Adicionalmente, reconhecemos as limitações do estudo antes de concluir e apontar direções para trabalhos futuros.

4.1 Discussão dos Resultados

O achado mais proeminente deste estudo é que a "revivência"de um repositório de software não representa um retorno ao seu estado anterior de atividade, mas sim uma transformação para um novo modo de operação, caracterizado por um volume drasticamente reduzido. A queda superior a 90% em commits, Pull Requests e Issues sugere que a retomada é frequentemente liderada por um núcleo muito menor de desenvolvedores, talvez um único mantenedor dedicado, em vez de uma reativação completa da comunidade.

4.1.1 RQ1 (Qualidade dos Artefatos)

Os resultados foram ambíguos e refutaram parcialmente a hipótese alternativa ($H1_1$). Por um lado, o aumento significativo na taxa de *Conventional Commits* (de 10,38% para 23,58%) indica uma maior disciplina e preocupação com a clareza do histórico de desenvolvimento. Este achado se conecta a trabalhos como o de Ma et al. [2], que discutem o impacto da qualidade na aceitação de PRs, sugerindo que a comunidade pós-revivência pode estar tentando sinalizar maior qualidade em suas contribuições. Por outro lado, a deterioração das métricas de PRs (aumento do tempo de merge e queda na taxa de aceitação) aponta para o oposto, indicando um processo de integração de código menos eficiente.

4.1.2 RQ2 (Dinâmica de Contribuição)

Os dados refutam veementemente a hipótese alternativa ($H1_2$) de que a atividade aumentaria. A realidade observada é uma atividade residual. Este resultado contribui para a área de "Abandono e Sobrevivência"de projetos OSS, explorada por Avelino et al. [1]. Enquanto o trabalho deles se concentra no ciclo de vida geral, nosso estudo quantifica a magnitude da queda de atividade em projetos que conseguem "sobreviver", mostrando que a sobrevivência não implica em prosperidade. A revivência parece ser mais uma fase de manutenção de baixa intensidade do que um renascimento.

4.1.3 RQ3 (Comportamento de Revisão)

Os resultados também refutaram a hipótese alternativa ($H1_3$). O processo de revisão, longe de se tornar mais responsivo, tornou-se um gargalo significativo. O aumento de 12 para 44 dias no tempo médio de merge, apesar do volume muito menor de PRs, é um forte indicativo de que a capacidade de revisão da equipe mantenedora está sobrecarregada. Isso pode ser explicado por uma perda de contribuidores experientes, um ponto relacionado ao engajamento comunitário estudado por Calefato et al. [3]. Uma equipe menor ou menos experiente pode levar mais tempo para avaliar cada contribuição, resultando em um processo mais lento, mesmo com menos trabalho na fila.

Em síntese, o cenário pós-revivência é de um projeto que opera em "modo de economia de energia": o volume de novas contribuições é baixo, mas há uma tentativa de aumentar a disciplina (commits convencionais), ao mesmo tempo que os processos de revisão e integração se tornam o principal ponto de atrito.

4.2 Limitações do Estudo

É fundamental reconhecer as limitações desta pesquisa para contextualizar adequadamente os resultados.

- **Validade Externa:** O estudo foi conduzido com uma amostra de 25 repositórios. Embora o fenômeno de revivência seja específico, este tamanho de amostra pode não ser suficiente para generalizar as conclusões para todo o ecossistema de software de código aberto. Os resultados podem ser particulares aos tipos de projetos analisados.

- **Validade de Construto:** As métricas utilizadas são proxies para conceitos complexos. Por exemplo, o "tempo de merge de PR"foi usado para inferir sobre a "eficiência da revisão", mas um tempo mais longo pode, alternativamente, indicar uma revisão mais cuidadosa e detalhada, e não necessariamente um gargalo negativo. Da mesma forma, a contagem de commits não distingue entre grandes contribuições de funcionalidades e pequenas correções.

- **Correlação não implica Causalidade:** Sendo este um estudo observacional, identificamos correlações entre a revivência de um projeto e mudanças em suas métricas de atividade. No entanto, não podemos afirmar uma relação de causa e efeito. Fatores externos não medidos, como a mudança de mantenedores, a popularidade da tecnologia subjacente ou o surgimento de um projeto concorrente, podem ter influenciado os padrões observados.

Essas limitações não invalidam os achados, mas reforçam a necessidade de cautela na sua interpretação e destacam a importância de pesquisas futuras para aprofundar a compreensão do fenômeno.

4.3 Trabalhos Futuros

Com base nos resultados e limitações deste estudo, propomos as seguintes direções para pesquisas futuras:

- **Análise Qualitativa:** Investigar o porquê por trás dos números. Entrevistas com mantenedores de projetos revividos ou a análise de conteúdo das discussões em Issues e PRs poderiam revelar as causas dos gargalos no processo de revisão.

- **Análise do Perfil dos Contribuidores:** Diferenciar a atividade dos mantenedores originais versus novos contribuidores após a revivência. Isso permitiria entender se a retomada é impulsionada pelos membros antigos ou por uma nova comunidade, complementando estudos sobre engajamento e onboarding.
- **Expandir o Dataset:** Replicar o estudo com um conjunto de dados maior e mais diversificado para aumentar a validade externa e a generalização dos resultados aqui encontrados.

4.4 Conclusão Geral

Este estudo teve como objetivo analisar e comparar as características de repositórios de software de código aberto antes de um período de inatividade e após sua revivência. Por meio de uma análise quantitativa de 25 repositórios, concluímos que a revivência não é um retorno ao status quo, mas sim uma transição para um estado de desenvolvimento de menor escala e com dinâmicas distintas.

A principal contribuição deste trabalho é a evidência empírica de que, embora projetos OSS possam ser "ressuscitados", eles operam com um volume de atividade drasticamente inferior. Além disso, identificamos uma dualidade nas práticas de desenvolvimento pós-revivência: enquanto a disciplina na escrita de commits parece melhorar, a eficiência do processo de revisão de Pull Requests se deteriora consideravelmente, tornando-se um potencial gargalo para a sustentabilidade do projeto a longo prazo. Esses achados desafiam uma visão otimista da revivência e fornecem uma perspectiva mais realista sobre a saúde e a capacidade de projetos que retornam da inatividade.

Referências

[1] G. Avelino, M. T. Valente, and A. Hora, "A large-scale study on the usage of lifecycles in open source projects," *Journal of Systems and Software*, vol. 153, pp. 1-16, 2019.

[2] Y. Ma, A. F. Bissyandé, D. Lo, and J. Klein, "An empirical study of the impact of pull request metrics on review latency," in *Proc. 27th IEEE/ACM Int. Conf. Program Comprehension (ICPC)*, 2019, pp. 264-275.

[3] F. Calefato, F. Lanubile, and N. Novielli, "A large-scale empirical study of newcomer behavior in GitHub," *Information and Software Technology*, vol. 141, art. no. 106720, 2022.

[4] A. Alami, F. Khomh, and G. Antoniol, "On the Sustainability of Open-Source Projects: An Empirical Investigation of GitHub Repositories," *IEEE Transactions on Software Engineering*, vol. 50, no. 1, pp. 248-265, 2024.

[5] J. Coelho, I. Wiese, and M. A. Gerosa, "Project abandonment in the wild: a study of GitHub's most popular projects," in *Proc. 16th Int. Conf. Open Source Systems*, 2020, pp. 53-63.

[6] P. Kaur and B. S. Lee, "Investigating Community Smells and their Impact on Developer Onboarding and Retention in OSS Projects," in *Proc. ACM/IEEE Int. Symp. Empirical Software Engineering and Measurement (ESEM)*, 2022, pp. 1-11.