

Mining Software Repositories with AIDev

1. Analysis of the AIDev Dataset

The AIDev dataset is a large-scale relational dataset designed to capture the interactions between AI coding agents and human developers on GitHub through agent-authored pull requests (Agentic-PRs). Based on the provided Entity-Relationship Diagram (ERD), the dataset is structured around a central `pull_request` table, which serves as the core analytical unit.

This central table is linked to several auxiliary tables, each contributing a specific analytical perspective. The repository table provides contextual metadata such as programming language, popularity indicators (stars, forks), and project characteristics. The user table describes developer profiles, including organizational affiliation and geographical information. Temporal and process-oriented aspects of pull requests are captured in the `pr_timeline` table, which records lifecycle events such as creation, assignment, review, and merge.

The dataset further includes `pr_reviews` and `pr_comments` tables, enabling a detailed analysis of human and automated review interactions. Code-level evolution is represented through `pr_commits` and `pr_commit_details`, allowing fine-grained inspection of modifications introduced by each pull request. Finally, the `related_issue` and `issue` tables link pull requests to issue tracking systems, enabling the study of bug fixes and feature implementations.

From a semantic perspective, AIDev captures the activity of five autonomous AI coding agents—OpenAI Codex, Devin, GitHub Copilot, Cursor, and Claude Code—alongside human-authored pull requests for comparative analysis. Two subsets are provided: the full AIDev dataset, comprising 456,535 agent-authored pull requests, and AIDev-pop, a curated subset limited to repositories with at least 500 GitHub stars, containing 7,122 agent-authored pull requests.

The richness and diversity of the dataset enable the investigation of multiple research themes, including the adoption of AI coding agents, the quality and acceptance of AI-generated pull requests, human–AI collaboration dynamics during code review, productivity impacts, and risk management aspects such as security and bias.

2. Study Design Using the Goal–Question–Metric (GQM) Approach

2.1 Research Goal

The goal of this study is to analyze the influence of automated code review bots on pull requests generated by AI coding agents in order to understand their impact on pull request

acceptance, review efficiency, and collaboration quality, from the perspective of human developers, within the context of popular GitHub repositories.

2.2 Research Questions

RQ1: Are pull requests generated by AI agents that are reviewed by automated bots accepted or rejected more quickly than those reviewed exclusively by human developers?

RQ2: What types of review comments (e.g., corrective, stylistic, security-related, or testing-related) are most frequently produced by automated review bots on AI-generated pull requests, and how are these comments addressed by human developers?

RQ3: Does a “closed-loop” bias exist when the AI coding agent and the automated review bot originate from the same provider, and how does this affect pull request outcomes and review behavior?

2.3 Metrics

To address RQ1, the study will measure the median review duration (time between pull request creation and closure), pull request acceptance rate, and effect sizes using non-parametric statistical tests such as the Mann–Whitney U test and Cliff’s delta.

For RQ2, review comments will be categorized into thematic classes (corrective, style, security, testing). The resolution rate of comments will be measured by tracking follow-up commits, and sentiment analysis techniques will be applied to assess the tone of the feedback.

For RQ3, the proportion of pull requests involving agents and bots from the same provider will be quantified. Acceptance rates and review durations will be compared between closed-loop and open-loop configurations, complemented by a qualitative analysis of a sampled subset of review comments to identify potential bias patterns.

3. Dataset Selection and Characterization

This study will utilize the AIDev-pop subset, which includes pull requests from repositories with at least 500 GitHub stars. This subset is selected due to its higher data quality, richer metadata, and greater relevance for generalizable conclusions.

The dataset will be further filtered to include pull requests generated by the five AI agents and a matched set of human-authored pull requests from the same repositories and time periods. The resulting dataset is expected to contain approximately 7,122 AI-generated pull requests and 6,628 human-generated pull requests.

Key variables include pull request identifiers, agent names, reviewer type (human or bot), review comments, time-to-closure, and merge status. Relevant relational links will be established across `pr_reviews`, `pr_comments`, `pr_timeline`, and `pr_commit_details` tables. This curated subset enables a focused and rigorous analysis of code review dynamics while

maintaining practical feasibility and analytical depth.