Data Curation Pipeline

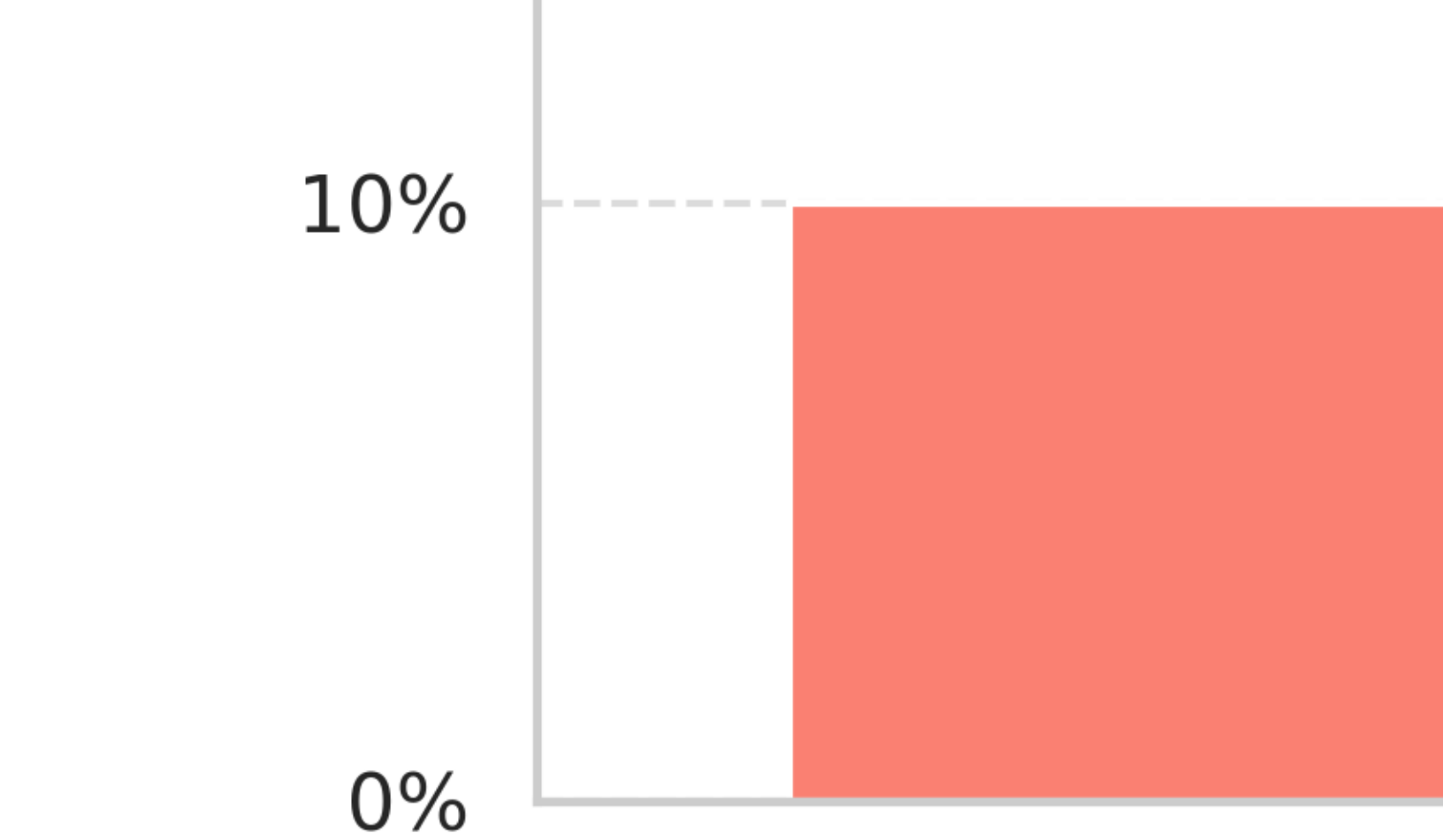
Filtering Results Report

May 16, 2025

Analysis of 1 GitHub Repositories

Generated with SWE-RL Inspired Data Curation Pipeline


Total PRs


10%
0%

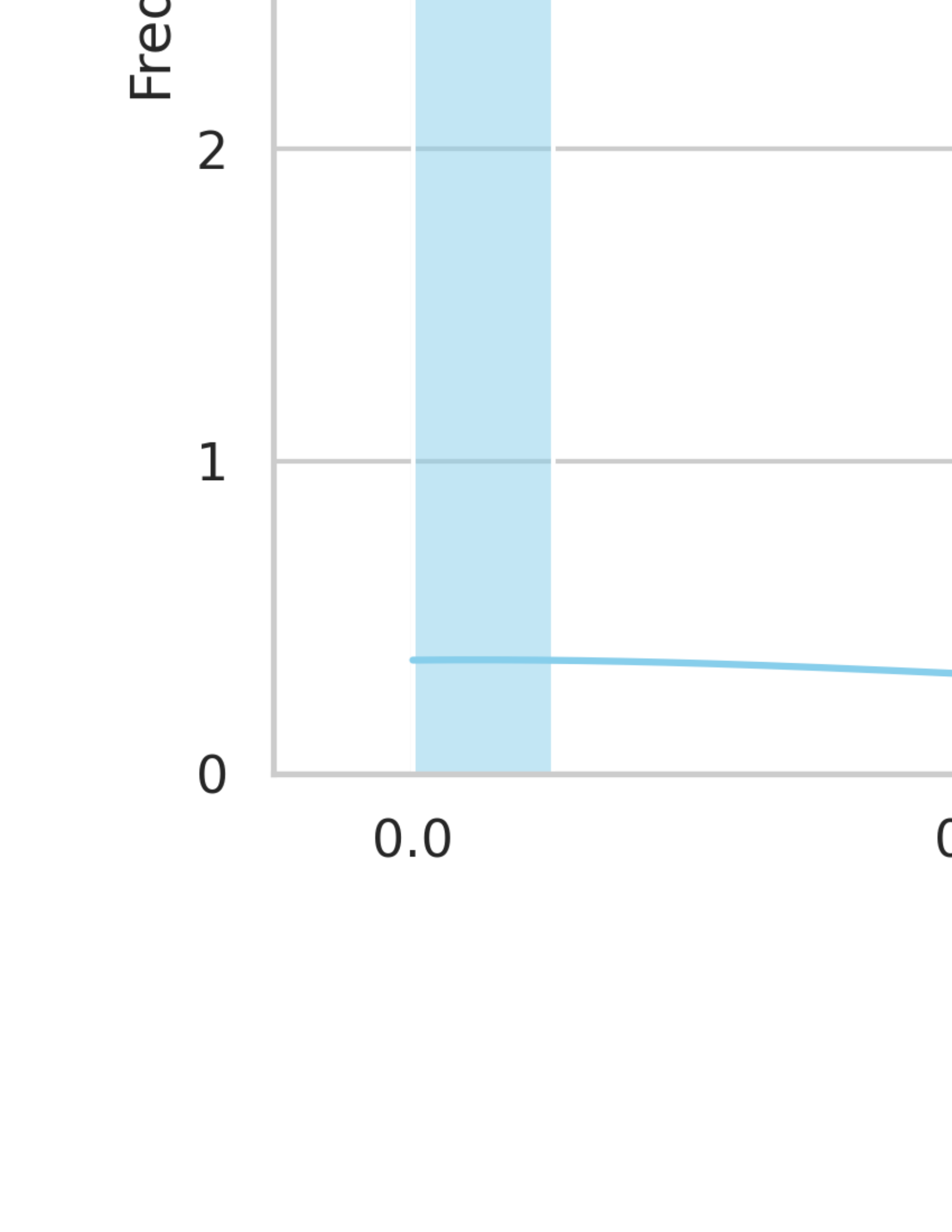
Data
20%
10%
0%

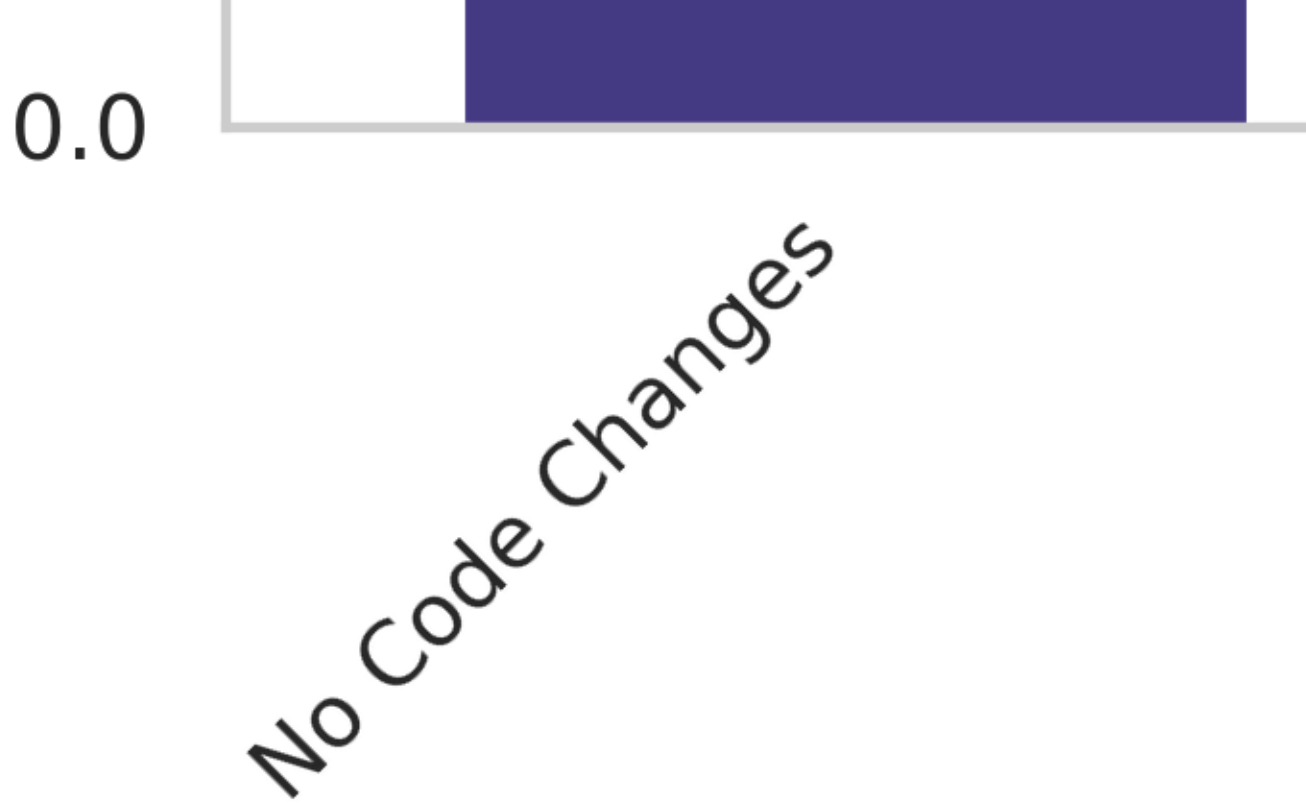

0.2
0.0

Rep

Total PRs: 10
Passed PRs: 5 (
Data Reduction
Average Quality

Filtering Breakc
- Bot Filter: 1 PF
- Size Filter: 4 P
- Content Filter:


Frec
2
1
0
0.0


0.0
No Code Changes

Quality Profiles of Exemplary PRs

The following pages showcase high-quality PRs that passed all filtering stages. These PRs represent exemplary software engineering data with meaningful problem-solving content, appropriate size, and high relevance scores.

Each scorecard provides detailed metrics on the PR's quality dimensions, including file composition, code changes, and identified relevant files that provide context for understanding the changes.

Number of Lines
10
8
6
4
2
0
Additions

Number of Lines
30
20
10
0
Additions

Number of Lines
30
25
20
15
10
5
0
Additions

Methodology

The data curation pipeline implements a multi-stage filtering approach inspired by the SWE-RL paper, focusing on extracting high-quality software engineering data from GitHub repositories. The pipeline consists of the following key components:

1. Data Acquisition
   - GitHub API integration for PR events and metadata
   - Repository cloning for file content access
   - Linked issue resolution and context gathering

2. Multi-Stage Filtering
   - Bot and Automation Detection: Identifies and filters out automated PRs
   - Size and Complexity Filtering: Ensures PRs are neither trivial nor unwieldy
   - Content Relevance Filtering: Focuses on meaningful software engineering content

3. Relevant Files Prediction
   - Identifies semantically related files not modified in the PR
   - Uses import analysis and directory structure heuristics
   - Enhances context for understanding code changes

4. Quality Metrics Generation
   - Comprehensive quality scoring across multiple dimensions
   - Metadata extraction for filtering decisions
   - Relevance scoring based on problem-solving indicators

The filtering pipeline maintains high precision by using progressive refinement, ensuring that only PRs with genuine software engineering value are retained while capturing detailed metadata about filtering decisions.