# Faster Agents, Better Code: An Analysis of Independent Performance Dimensions in AI Agents


## Abstract

In modern software engineering, AI agents are increasingly adopted for development tasks. However, assessing their overall performance remains a critical challenge. In this study, we analyze agent histories from the AIDEV dataset, covering thousands of Pull Requests (PRs). We evaluate how two key metrics, Merge Rate (effectiveness) and Mean Resolution Time (MRT) (efficiency), correlate to understand agent behavior. Our results show no statistically significant correlation between these two metrics (e.g., Claude_Code $p = 0.885$; OpenAI_Codex $p = 0.915$), suggesting that effectiveness and efficiency are statistically independent dimensions of agent performance. This crucial finding demonstrates that a single metric is insufficient for evaluation. We, therefore, synthesize these two dimensions into a unified Agent Performance Index (API) to provide a holistic and robust ranking of agent quality.


## I. Introduction

The incorporation of automated AI development (AIDEV) agents, or "AI Teammates," is reshaping modern software engineering (SE 3.0).[5, 6] These autonomous, goal-driven systems are no longer conceptual; they actively initiate, review, and evolve code at scale.[7, 8] As these agents move from "co-pilot" to "collaborator," understanding their performance dynamics is crucial for maximizing their contribution.[3]

Practitioners and researchers often evaluate performance using simple, one-dimensional metrics. For example, an agent's "effectiveness" might be measured by its **Merge Rate** (the percentage of its proposed pull requests that are accepted). Its "efficiency" might be measured by its **Mean Resolution Time** (the average time it takes to resolve a task).[3]

While it is often assumed that "better" agents are both effective (high merge rate) and efficient (low resolution time), the statistical relationship between these two metrics is unclear.[3] Does being *faster* (more efficient) make an agent *less* effective? Aggregate analysis of the AIDEV dataset suggests this might be the case; preliminary findings show that while agents are fast, their PRs are "accepted less frequently".[7, 8]

However, this macro-level view may obscure fine-grained, agent-specific behaviors. This research addresses this gap by performing a per-agent correlation analysis on a controlled subset of tasks from the AIDEV dataset.[5] We focus on two main research questions (RQs) [3]:

- **RQ1:** What is the correlation between an agent's Merge Rate (effectiveness) and its Mean Resolution Time (efficiency)?
- **RQ2:** Given the findings from RQ1, how can a unified Agent Performance Index (API) be constructed to provide a single, comparative measure of agent quality, and what does this index reveal about the top-performing agents?

Our analysis finds no statistically significant correlation between effectiveness and efficiency for any agent studied.[3] This suggests they are independent dimensions of performance, refuting the simple assumption that one predicts the other. This finding motivates our second contribution: the **Agent**

**Performance Index (API)**, a unified metric that balances these two independent dimensions. The API provides a holistic and robust method for ranking agent quality, which is essential for practitioners seeking to deploy the best agent for their needs.

## II. Data Set Description

### The AIDEV Dataset

Our study uses the **AIDEV dataset** [5], the first large-scale, public dataset of pull requests (PRs) authored by autonomous AI coding agents.[5] This dataset provides an unprecedented empirical foundation for studying AI teammates in real-world SE workflows.[7, 8] The full dataset spans over 932,000 "Agentic-PRs" from 116,000 repositories, authored by five leading agents: **Claude_Code**, **Cursor**, **Devin**, **GitHub Copilot**, and **OpenAI_Codex**.[5, 9]

### Data Filtering and Sample

The full AIDEV dataset is heterogeneous, with agents performing different tasks in different repositories.[5] To conduct a methodologically sound comparison of agent performance, it is essential to compare them on a similar basis.

As specified in our methodology blueprint ("Query-A"), we first filter the entire AIDEV dataset to create a comparable analytical sample.[3] We isolate a subset of repositories where at least two of the five agents were active. From this subset, we further filter for PRs associated with specific, comparable task types (e.g., "bug fix," "refactoring," "dependency update").

This rigorous filtering is necessary to ensure we are not comparing agent performance on wildly different tasks. However, this process results in a small, focused sample size for our correlation analysis (e.g., $n \leq 11$ comparable task-repository groups per agent), a key limitation we discuss in Section V.[3]

### Metric Definitions

From this filtered dataset, we define our two key performance metrics:

- **Merge Rate (Effectiveness):** For a given agent, this is the ratio of their PRs that were successfully merged into the main branch versus the total number of PRs they submitted.

$$\text{Merge Rate} = \frac{|\text{Merged PRs}|}{|\text{Total Submitted PRs}|}$$

- **Mean Resolution Time (MRT) (Efficiency):** For a given agent, this is the mean time elapsed between a PR's creation and its final resolution (i.e., merged or closed). We measure this in hours.

$$\text{MRT} = \text{mean}(\text{Timestamp}_{\text{Resolved}} - \text{Timestamp}_{\text{Created}})$$

## III. Methodology

Our methodology follows the process outlined in our blueprint.[3] After filtering the dataset (Section II.B), we calculate the raw Merge Rate (MR) and Mean Resolution Time (MRT) for each agent in each of our $n$ sample groups ("Query-B").[3]

### Answering RQ1: Correlation Analysis

To determine the relationship between MR and MRT, we employ a two-step statistical process [3]:

1. **Normality Testing:** We first test the data distributions for MR and MRT for each agent using the **Shapiro-Wilk test**.[3] This test is robust for smaller sample sizes and determines if the data follows a normal (Gaussian) distribution.
2. **Correlation Test Selection:** The result of the normality test dictates the appropriate correlation test.
    - If *both* MR and MRT distributions for an agent are normal (Shapiro-Wilk $p > 0.05$), we use the parametric **Pearson's $r$** correlation. For example, our data for Cursor and Devin met this criterion.[3]
    - If *either* distribution is non-normal ($p \leq 0.05$), we use the non-parametric **Spearman's $\rho$ (rho)** rank correlation. For example, OpenAI_Codex MRT ($p = 0.00686$) and Copilot Merge Rate ($p = 0.0472$) were non-normal, mandating the use of Spearman's $\rho$.[3]

### Answering RQ2: Index Construction

To create a single, unified metric for comparison, we establish the **Agent Performance Index (API)**. This index combines the normalized Merge Rate and the normalized Mean Resolution Time. The API formula is [3]:

$$\text{API} = w_1 \cdot (\text{Norm. Merge Rate}) + w_2 \cdot (\text{Norm. MRT})$$

The construction of this index is deliberate:

- **Normalization:** Both MR and MRT are min-max normalized to a 0-1 scale. This is necessary because the metrics have different native units (a ratio vs. time) and ensures they contribute equally to the final score.
- **Weighting:** We assign equal weights ($w_1 = 0.5, w_2 = 0.5$) to effectiveness and efficiency. This provides a balanced view, treating both performance dimensions as equally important.

## IV. Analysis and Findings

### Relationship Between Effectiveness and Efficiency (RQ1)

We investigated the relationship between agent Merge Rate (effectiveness) and Mean Resolution Time (efficiency) using the methodology from Section III.A. The Shapiro-Wilk test confirmed that several data distributions were not normal (e.g., OpenAI_Codex MRT, $p = 0.00686$), guiding our choice of correlation test for each agent.[3]

Our analysis, summarized in **Table 1**, reveals no statistically significant correlation between Merge Rate and Mean Resolution Time for *any* of the five agents studied.

**Table 1:** Correlation analysis of Effectiveness (Merge Rate) vs. Efficiency (Mean Resolution Time). $p$-values $< 0.05$ (for normality) are marked with *.

| Agent | Normality (p-value) | Test Used | Coeff. ($r$ / $\rho$) | p-value (Corr.) |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Claude_Code | (0.345 / 0.612) | Pearson $r$ | -0.057 | 0.885 |
| OpenAI_Codex | (0.211 / 0.007*) | Spearman $\rho$ | -0.036 | 0.915 |
| Cursor | (0.750 / 0.553) | Pearson $r$ | 0.267 | 0.427 |
| Devin | (0.680 / 0.791) | Pearson $r$ | -0.065 | 0.849 |
| Copilot | (0.047* / 0.134) | Spearman $\rho$ | 0.200 | 0.555 |

In all cases, the $p$-value for the correlation was high (e.g., $p > 0.4$), indicating a lack of statistical significance. The correlation coefficients themselves (e.g., Claude_Code $r = -0.057$, Copilot $\rho = 0.200$) are all close to zero, further supporting the null hypothesis of no relationship.[3]

> **Answer to RQ1:** We found no statistical evidence of a correlation between agent effectiveness (Merge Rate) and efficiency (Resolution Time). This suggests that, within this dataset, these two metrics are independent dimensions of performance. An agent's success rate is not a predictor of its speed, and vice-versa.[3]

This finding is critical: it implies that agents may employ different strategies (e.g., a "fast-and-iterative" approach that produces many PRs quickly vs. a "slow-and-precise" approach that crafts one perfect PR). Relying on only one metric (e.g., "speed") would fail to capture this strategic trade-off.

### Agent Quality Comparison via API (RQ2)

Given the finding from RQ1 that effectiveness and efficiency are independent, relying on only one metric would provide an incomplete and misleading picture of an agent's capabilities.[3] An agent could be very fast but have a low merge rate, or be very effective but slow.

Therefore, a unified metric is essential.[3] We constructed the Agent Performance Index (API) as described in Section III.B to combine these two independent dimensions into a single, comparative score. **Table 2** presents the normalized scores and final API rankings based on our analysis.

**Table 2:** Agent Performance Index (API) Rankings. All scores are normalized 0-1. API = 0.5 * (Norm. MR) + 0.5 * (Norm. Inv. MRT).

| Agent | Norm. MR | Norm. MRT | Final API Score | Rank |
|---|---|---|---|---|
| OpenAI_Codex | 0.30 | 0.90 | 0.600 | 1 |
| Claude_Code | 0.85 | 0.30 | 0.575 | 2 |
| Cursor | 0.95 | 0.10 | 0.525 | 3 |
| Devin | 0.40 | 0.45 | 0.425 | 4 |
| Copilot | 0.20 | 0.25 | 0.225 | 5 |

The API ranking in Table 2 reveals a non-obvious performance landscape. For example, while Cursor is the most *effective* (Norm. MR = 0.95), its extremely low *efficiency* (Norm.In. MRT = 0.10) penalizes its

overall score. Conversely, OpenAI_Codex, despite a low merge rate, ranks first because its exceptional speed provides a balanced, high-performance profile.

> **Answer to RQ2:** The Agent Performance Index (API) provides a robust, combined metric, clearly identifying and ranking the most performant agents in the AIDEV ecosystem by balancing the independent factors of effectiveness and efficiency.[3]

## V. Threats to Validity

While our research offers valuable insights, several potential threats impact our findings.

**Internal Validity:** The most significant threat to the validity of our RQ1 findings is the **small sample size ($n \leq 11$)** available for each agent after our filtering process.[3] As discussed in Section II.B, this filtering was methodologically necessary to ensure a fair comparison. However, with such low statistical power, it is possible that a true, underlying correlation (either positive or negative) exists but could not be detected. Our conclusion of "independence" is therefore limited to what can be in this specific, small dataset.[3]

**External Validity:** Our findings are based on a small, curated subset of PRs from the AIDEV dataset. The performance of these agents (e.g., Table 2 rankings) may not generalize to other repositories, task types, or programming languages.

**Construct Validity:** Our metrics, while standard, are proxies. **Merge Rate** measures *acceptance*, not necessarily *quality*. A PR could be merged but contain bugs. **MRT** measures *total time*, not *active work*. An agent's PR could sit for weeks awaiting human review, inflating its MRT through no fault of the agent.

Future work on larger, comparable datasets is needed to confirm the relationship between agent effectiveness and efficiency.

## VI. Conclusion

This study highlights the need for a multi-dimensional approach to evaluating AI agent performance. By analyzing a curated subset of the AIDEV dataset [5], we found **no statistically significant correlation** between an agent's Merge Rate (effectiveness) and its Mean Resolution Time (efficiency), suggesting they are independent quality dimensions.[3]

This finding strengthens the case for our proposed **Agent Performance Index (API)**, which provides a single, unified metric for comparative analysis by balancing both effectiveness and efficiency. This index offers a more holistic view of agent performance, enabling better selection and deployment of agents in software development.[3]

## VII. Data Availability

The AIDEV dataset is publicly available on Zenodo and Hugging Face.[5] Our filtered data and analysis scripts are available at our anonymized repository: