Gemini 3.8 Flash

Performance on the CRED single-error benchmark

Overall performance

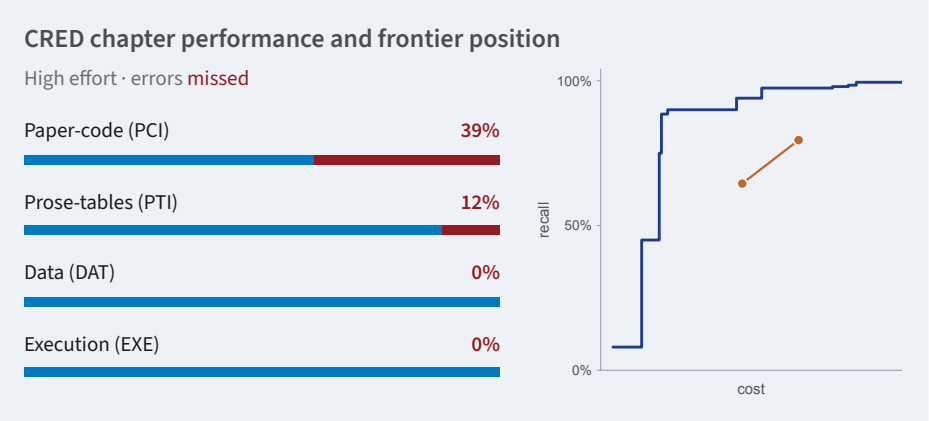
Gemini 3.8 Flash improves with higher reasoning effort, but remains below the cost-recall frontier at both tested settings. Its principal weakness is paper-code consistency: it often fails to detect when the manuscript describes a different procedure from the one actually implemented.

This assessment uses the CRED taxonomy and controlled error-injection framework introduced in *Verifying the Verifiers*. The model receives paper and code in a single request through OpenRouter. It is tested on 100 host papers, each prepared in two separate single-error versions, giving 200 trials with known defects.

Recall rises from 64.5% at low effort to 79.5% at high effort. The improvement comes at 5.2 times the cost. High effort costs $0.144 per paper, yet still misses about one in five injected errors under the published scoring rule.

Cheaper single-shot configurations do better. GLM 5.3 Flash high detects 88.5% for less than a cent per paper. GPT-5.6 Luna xhigh reaches 97.5% at about a third of Gemini high's cost. Gemini is therefore not just below a more expensive accuracy leader: it is outperformed by alternatives that also cost less.

Most of the gain from higher effort comes from paper-code inconsistencies, even though these remain its weakest category. More reasoning does help Gemini compare the paper with its implementation; it reduces, but does not remove, the main weakness.



Gemini misses paper-code inconsistencies most often, and neither reasoning setting reaches the cost-recall frontier.

Where it is weak

At high effort, 35 of 41 scored misses are paper-code inconsistencies (PCI). The remaining misses are prose-table inconsistencies (PTI). PCI is also the weakest chapter after allowing for its larger size: the model misses about 39% of its injected defects, compared with 12% in PTI.

Variable construction accounts for 21 of the 35 PCI misses, the largest missed PCI subtype. This points to a recurring gap in tracing a declared data transformation into the implementation. Detecting a contradiction within the manuscript appears easier here than checking whether the code fulfills what the manuscript promises.

All injected data-integrity and execution defects receive detection credit. But the execution cases include a supplied code-running verdict. Those results do not establish that Gemini can independently reproduce a study; they measure its ability to use the evidence supplied in this benchmark.

A symptomatic miss

In a benchmark copy of a paper on criminal-record clearing, the manuscript says earnings are adjusted for inflation. The code instead takes the logarithm of nominal earnings, with no inflation adjustment. Taking logs changes the scale; it does not convert nominal earnings into real earnings.

Gemini high returns three other prose-table findings but never flags this mismatch. It detects contradictions within the manuscript while leaving the stated outcome construction unchecked. The omission fits the broader PCI pattern: the model does not follow a declared data transformation into the code.

The observed weakness is incomplete cross-checking. It is not evidence that Gemini cannot understand inflation adjustment, and the mismatch alone does not establish that the estimated policy effect changes. What the response shows is narrower: the model does not reconcile this explicit manuscript claim with the code.

Based on the framework in *Verifying the Verifiers*. Scores follow CRED's published matcher, which can under-credit valid detections. Costs use dated price lists; comparisons are descriptive.
Live benchmark