

Stability of a Large Language Model-Based Automated Grading System: A Diagnostic Analysis of a Multi-Phase Architecture with Constrained Revision

Nianyun Liu, Yang Gao

School of Business, Jiangnan University, Hubei, Wuhan 430056, China

Abstract: Large language models (LLMs) can interpret open-ended responses and generate scores with written rationales. Their outputs may vary across calls, even when the assignment, rubric, model, and sampling settings remain unchanged. This variation can arise from prompt design, context organization, sampling, or the service environment. It may become more difficult to trace in a grading workflow that includes initial scoring, score revision, and holistic review, because a later phase can either moderate or enlarge an earlier difference. This study investigates that problem through a three-phase grading workflow: ensemble-based initial scoring, self-critique-based revision, and holistic review. System logs were examined to identify where score differences emerged. The logs suggest that incompatible structured-output formats may interfere with the aggregation of initial scores. They also show that the revision and review phases can make relatively large changes when no explicit adjustment limit is imposed. In response, the system combined prompt-level guidance with programmatic clipping. In Phases 2 and 3, the adjustment for each dimension was restricted to $\pm 30\%$ of the corresponding score produced in the preceding phase. The diagnostic sample consisted of one student assignment graded by DeepSeek-Chat. Six baseline tests and four tests after the constraint mechanism was added were conducted. Among the two baseline tests with complete phase-level records, the final scores differed by 5.80 points. Most of this difference appeared during self-critique-based revision and holistic review. The four post-deployment final scores were 5.40, 4.50, 3.70, and 4.20, producing a range of 1.70 points, or 11.3% of the assignment's maximum score. The narrower range suggests that adjustment limits can reduce large swings in the total score in this workflow. The control was less successful at the item level: Phase 3 still showed more cross-test variation than Phase 1, and only four of the twenty scoring items received the same score in all four tests. These results are consistent with prior discussions of LLM output variability, but they also indicate that total-score stability can conceal changes in individual scoring dimensions. The evidence is limited to one assignment, one model, and ten repeated tests, so the findings should be read as a system diagnosis rather than a general estimate of grading reliability. Further tests across assignments, models, rubrics, and sampling settings are needed before the mechanism can be applied more broadly.

Keywords: Large language models; automated grading; grading stability; ensemble scoring; constrained revision.

1. Introduction

Large language models can relate ideas in open-ended responses to rubric criteria and produce both scores and explanations. This makes them suitable for tasks such as essay assessment, case analysis, and the grading of open-ended questions. Unlike systems based on fixed rules or manually selected features, LLM-based systems can handle a wider range of wording and argument structures [1]. That flexibility comes with a known difficulty: repeated calls do not always return the same judgment. Differences may occur even when the assignment, rubric, model, and sampling parameters are held constant.

For educational assessment, agreement with human raters is only one part of system evaluation. A second question is whether the system gives similar results when the same response is submitted repeatedly. A difference caused by the generation path or service timing, rather than by the content of the response, can weaken confidence in the resulting grade [2]. The issue is harder to isolate in a multi-phase workflow. An initial score may be revised after self-critique and then changed again during holistic review. These later steps may correct an earlier discrepancy, but they may also introduce a new one if the model changes its interpretation of the rubric or the scoring scale [3].

The present study examines this problem in a three-phase

LLM grading system. The first phase generates ensemble scores, the second revises them through self-critique, and the third conducts a holistic review. System logs were used to trace two types of behavior: failures in structured-score aggregation and changes in scores between phases[4]. The study then compares repeated tests conducted before and after the introduction of a constraint mechanism. The mechanism uses prompt instructions to discourage excessive changes and code-level clipping to prevent changes beyond a fixed range.

The analysis addresses three questions:

How much variation remains after ensemble-based initial scoring?

Do self-critique and holistic review introduce larger score differences than the initial phase?

Does restricting inter-phase adjustments improve consistency in total scores and individual scoring items?

The available data do not support a broad claim about the reliability of LLM grading. Six baseline tests and four constrained tests were conducted on one assignment, and only two baseline tests contained complete records for all phases. The study is therefore treated as a diagnostic case rather than a comparative evaluation of models or grading methods. Its purpose is to locate possible sources of instability in the workflow and to determine whether a simple adjustment limit changes the observed pattern. The results can inform the

design of larger studies, but they cannot establish that the same pattern will occur across other courses, assignments, rubrics, or models.

2. System Architecture and Constrained-Revision Mechanism

2.1. Three-Phase Grading Workflow

The grading workflow contains three phases. Phase 1 uses ensemble scoring. For each rubric dimension, the system calls the model three times and averages the returned scores. If $s_i^{(j)}$ is the score for dimension j in call i , the Phase 1 score is

$$s_{P1}^{(j)} = \frac{1}{N} \sum_{i=1}^N s_i^{(j)}, N=3.$$

Averaging limits the effect of an unusually high or low score from one call. With three calls, for example, a one-point deviation in a single output changes the mean by approximately 0.33 points when the other two scores remain fixed. This approach follows the usual rationale for ensemble estimation: repeated judgments can reduce the influence of an isolated output. It does not guarantee stable scores, particularly when the calls share a systematic bias or produce similar errors.

Phase 2 revises the initial result through self-critique. The model receives the assignment, rubric, and Phase 1 scores, and checks whether relevant evidence was missed, assigned to the wrong rubric item, or matched to an unsuitable score band. Phase 3 conducts a holistic review using information from the first two phases. It returns the dimension scores, item-level scores, final total, and grading rationale.

The later phases differ from simple verification because they require the model to interpret the assignment and rubric again. If the prompt does not define what may be revised and by how much, the model may treat the review as a new grading task. In the baseline tests reported in this study, two runs with the same Phase 1 total of 3.85 ended with scores of 8.60 and 2.80. Although these two records are insufficient for a general comparison, they show why inter-phase changes required explicit control in this workflow.

2.2. Two-Layer Constraint Mechanism

Phases 2 and 3 use two controls: a prompt-level adjustment rule and code-level clipping. The prompt instructs the model to keep each dimension score within $\pm 30\%$ of the score from the preceding phase. It also asks the model to identify the evidence supporting a proposed change. The prompt is intended to preserve continuity between phases, but compliance depends on the model’s response in each call.

Code-level clipping is applied after the structured output has been parsed. For dimension j , the accepted revised score is

$$s_{rev}^{(j)} = clip(s_{LLM}^{(j)}, 0.7s_{prev}^{(j)}, 1.3s_{prev}^{(j)}),$$

where $s_{prev}^{(j)}$ is the preceding phase’s score, $s_{LLM}^{(j)}$ is the model’s proposed score, and $s_{rev}^{(j)}$ is the score retained by the system. The clipping operation replaces a value below the lower bound with $0.7s_{prev}^{(j)}$ and a value above the upper bound with $1.3s_{prev}^{(j)}$. Scores must also remain within the valid range of the rubric dimension.

The two controls have different purposes. The prompt shapes the revision process and requests an explanation, whereas clipping enforces the numerical boundary. This

distinction is consistent with prior work on structured LLM systems, which generally treats instruction following and programmatic validation as separate safeguards. The logs in the present study support that separation: three outputs exceeded the prompt-defined range and required clipping.

The 30% threshold was selected as an engineering setting for this diagnostic test, not as a validated standard. Its absolute effect depends on the preceding score. A score of 1.20, for instance, can change by only 0.36 points, whereas a score of 6.00 can change by 1.80 points. The rule may therefore prevent large score shifts but also restrict a justified correction when the initial score is low or inaccurate. A later implementation could combine a proportional boundary with an absolute-point allowance. The appropriate thresholds would need to be tested against human ratings and a larger set of assignments; the present experiment does not establish which rule produces the most accurate grades.

2.3. Grading-Rationale Configuration

For every score, the system required the model to return a short explanation. Once the constrained-revision mechanism was added, the maximum length of this explanation was raised from 80 to 200 Chinese characters. The purpose was to allow the model to link three elements more clearly: the rubric criterion, the evidence found in the assignment, and the score assigned to that evidence.

The study uses the term “grading rationale” for this output field. It refers to the explanation visible in the API response, not to the model’s internal reasoning. Increasing the character limit only allows a longer explanation; it does not show that the model identified better evidence or assigned a more accurate score. In the later tests, this change was therefore treated as a configuration adjustment rather than as evidence of improved grading quality.

3. Experimental Design

3.1. Experimental Environment

The tests used DeepSeek-Chat through the API endpoint <https://api.deepseek.com/v1/chat/completions>. The temperature was set to 0.05, the maximum output length to 8,192 tokens, and the context window to 128,000 tokens. These values were unchanged in all three grading phases. The client was run with `skip_cache=True` to reduce the possibility that a cached response would be returned in a repeated test.

The settings controlled by the client were held constant, but the online service could still change between calls. For example, server-side model updates, request scheduling, or changes to the inference backend might affect the result. The repeated tests should therefore be understood as identical in terms of input and client configuration, not as runs performed in a completely identical server environment.

3.2. Study Object and Scoring Rubric

The experiment used one student assignment as a fixed diagnostic case (`submission_id=1`). It contained approximately 1,078 characters, or about 431 tokens after tokenization. Using the same assignment in every run removed differences caused by the content of the student responses. This made it possible to examine score changes under a fixed input. It also narrowed the scope of the study: the results cannot be extended to assignments with different subjects, structures, or quality levels without further testing.

The rubric had a maximum of 15 points. It comprised two dimensions, each containing ten scoring items. Dim1, “job-role awareness, operational decision-making, and overall coordination,” accounted for 7.5 points. Dim2, “financial risk, team collaboration, and operational reflection,” accounted for the other 7.5 points.

The assignment, rubric, model, sampling parameters, and three-phase workflow remained unchanged during the experiment. This control supports comparisons between repeated runs, but it does not remove variation that may arise from the online API or from the model’s interpretation of the rubric.

3.3. Experimental Procedure

The baseline phase included six runs. The first three were used mainly to investigate problems with the structured-output format. According to the system logs, errors in parsing the rubric could affect the scores produced in the initial phase. After the format problem was corrected, the system was run three more times. Only two of these runs contained complete records for all grading phases and could be used for direct phase-level comparison.

The baseline runs were therefore not a uniform experimental group. They combined fault diagnosis, format correction, and repeated scoring under subsequently adjusted system conditions. The difference between baseline and later results should be interpreted with this limitation in mind.

Afterward, the system was updated in three ways: a prompt-level adjustment rule was added, programmatic clipping was applied to later-phase scores, and the maximum length of the grading rationale was increased. Four further runs were completed and labeled Run 1 to Run 4. For each run, the system stored the Phase 1 aggregated scores, the dimension scores from Phases 2 and 3, the final total, the twenty item-level scores, any clipping events, and the number of output tokens.

Because this section reports the experimental setup rather than an empirical comparison with previous studies, it does not make claims about whether the procedure performs better than methods reported elsewhere. Such a comparison would require a common assignment set, scoring rubric, model or baseline system, and evaluation criteria.

Methods

Means, sample standard deviations, ranges, and coefficients of variation were used to describe score variability. The range of final scores was defined as

$$\Delta = \max(G_i) - \min(G_r),$$

and the standardized range was defined as

$$\Delta_r = \frac{\Delta}{G_{\max}} \times 100\%,$$

where G_r denotes the final score in the r -th test and $G_{\max} = 15$. A value of 15% was used as a reference threshold for system diagnosis. This threshold is an engineering benchmark for the present study rather than a general standard for educational assessment.

The 95% confidence interval was estimated using the small-sample t distribution:

$$CI_{95\%} = \bar{x} \pm t_{0.975, n-1} \frac{s}{\sqrt{n}}$$

The constrained system was tested four times, giving three degrees of freedom. Because of the small sample size, the confidence interval is used only to describe the uncertainty associated with the present data and is not intended to support general statistical inference.

4. Experimental Results

4.1. Phase-Level Differences in the Baseline System

In the two baseline tests with complete phase-level records, the Phase 1 score for Dim1 was 3.00 in both tests, the Phase 1 score for Dim2 was 0.85 in both tests, and the initial aggregate score was 3.85 in both cases. After the second phase, the Dim1 scores were revised to 7.50 and 2.50, respectively. During holistic review, the Dim1 scores were 6.35 and 2.30, while the Dim2 scores were 2.30 and 0.50. The resulting final total scores were 8.60 and 2.80, a difference of 5.80 points.

4.2. Scores after Deployment of the Constrained-Revision Mechanism

The results obtained after deployment of the constrained-revision mechanism are presented in Table 1.

Table 1. Scores from the Four Tests after Deployment of the Constrained-Revision Mechanism

Indicator	Run 1	Run 2	Run 3	Run 4	Range	SD
P1: Dim1	3.70	3.10	3.10	3.00	0.70	0.32
P1: Dim2	1.20	1.20	1.20	1.20	0.00	0.00
P1 total	4.90	4.30	4.30	4.20	0.70	0.32
P2: Dim2	1.56*	1.55	1.56*	1.20	0.36	0.18
P3: Dim1	3.35	3.35	2.20	3.00	1.15	0.54
P3: Dim2	2.03*	1.20	1.50	1.20	0.83	0.39
Final total score	5.40	4.50	3.70	4.20	1.70	0.71

Note: P1 denotes initial ensemble scoring, P3 denotes holistic review, and SD denotes the sample standard deviation. An asterisk indicates that the score triggered programmatic hard clipping.

The mean final score was 4.45, the range was 1.70 points, and the standardized range was 11.3%, below the 15% reference threshold used in this study. Using the t distribution with three degrees of freedom, the 95% confidence interval for the final score was [3.31, 5.59]. The range of the Phase 1 aggregate score was 0.70 points, compared with 1.70 points

for the final score. The ranges for Dim1 and Dim2 in Phase 3 were 1.15 and 0.83 points, respectively, both larger than the corresponding variation observed in Phase 1. This suggests that later phases may still increase score differences, although the magnitude of such changes was restricted by the programmatic boundary.

The range after deployment of the constrained-revision mechanism was smaller than the 5.80-point difference observed between the two complete baseline records. However, the two sets of tests did not constitute a symmetric randomized comparison. The baseline phase included format-fault diagnosis, and the number and operating conditions of the tests were not fully equivalent. The reduction in score variation therefore cannot be attributed entirely to the constrained-revision mechanism.

4.3. Programmatic Clipping and Item-Level Stability

The logs recorded three clipping events in the four constrained runs. The Dim2 score in Phase 2 was clipped to 1.56 in Runs 1 and 3, and the Dim2 score in Phase 3 was clipped to 2.03 in Run 1. These cases show that the model did not always follow the $\pm 30\%$ limit stated in the prompt. The prompt rule could guide the output, but it could not enforce the boundary in every call. The clipping procedure therefore served as a separate check after the model response had been parsed.

Clipping limits the size of a score change; it does not show whether the score before clipping was justified. For example, if the Phase 1 score is too high or too low, a fixed boundary may also restrict a correction that would otherwise be reasonable. The clipping results should therefore be examined together with human scores, evidence in the student response, and the relevant rubric criteria. Without these comparisons, the present data can show that the mechanism controlled the numerical range, but not that it improved grading accuracy.

The holistic-review phase produced scores for twenty items in each run. Only four items received exactly the same score in all four runs, giving an item-level consistency rate of 20%. The remaining sixteen items changed to different degrees. This result contrasts with the relatively narrow range of the final total scores, which was 1.70 points. The two findings are not contradictory. An increase in one item can offset a decrease in another, leaving the total score relatively stable while the underlying scoring decisions change. This pattern is consistent with a point raised in earlier work on automated assessment: agreement at the total-score level does not necessarily mean agreement at the criterion or item level. In the present study, however, the evidence is limited to one assignment and four constrained runs, so it cannot establish how often this pattern occurs in other grading tasks.

4.4. Output-Token Counts

The length of the grading rationale was increased from a maximum of 80 Chinese characters to 200. After this change, the number of output tokens for one Dim1 call in Phase 1 rose from 274–285 to 402–530. The corresponding Dim2 range changed from 282–320 to 302–435. These figures show that the longer response limit was reflected in some Phase 1 outputs. The token counts in Phases 2 and 3, however, did not move in one consistent direction, so the available data do not support a general claim that the change increased output length throughout the workflow.

The four constrained runs also showed variation in the length of the three calls used for Phase 1 ensemble scoring. The average within-run token range was 65.0 for Dim1 and 76.5 for Dim2. At the same time, the Phase 1 Dim2 score was 1.20 in all four runs. In this small sample, longer or shorter rationales were therefore not accompanied by a

corresponding change in the Dim2 score.

Output-token count measures how much text the model generated. It does not measure the quality of the evidence cited, the correctness of the score, or the adequacy of the grading decision. This distinction is also important when interpreting studies that use response length as an indirect indicator of model reasoning. For the present experiment, the token data support only a limited conclusion: changing the rationale-length limit altered the amount of generated text in some calls, but no direct relationship between output length and scoring variation was observed.

5. Discussion

5.1. Variation across grading phases

The observed variation was smaller in initial ensemble scoring than in the later phases. Across the four constrained runs, the Phase 1 total ranged from 4.20 to 4.90, a difference of 0.70 points. The final scores ranged from 3.70 to 5.40, a difference of 1.70 points. At the dimension level, the Phase 1 ranges were 0.70 for Dim1 and 0 for Dim2, whereas the Phase 3 ranges were 1.15 and 0.83, respectively. In this case, therefore, score variation increased after self-critique and holistic review.

One possible explanation is that the phases performed different tasks. Phase 1 averaged the results of three calls, which reduced the influence of an unusually high or low result from a single call. Phases 2 and 3 required the model to reassess the links among the assignment, the rubric, and the assigned scores. This reassessment created additional opportunities for the model to reinterpret evidence or apply a different score band. The two complete baseline records illustrate this problem: both began with the same Phase 1 total of 3.85, but ended with final scores of 8.60 and 2.80. The 5.80-point difference arose mainly during revision and holistic review rather than initial scoring.

This pattern is broadly consistent with previous research showing that repeated LLM judgments can vary despite fixed inputs and low-temperature settings. It also qualifies the common assumption that self-critique necessarily makes an output more reliable. In the present workflow, an additional review phase did not consistently preserve the initial scoring scale. A direct numerical comparison with earlier studies is not possible because no common dataset, rubric, model, or stability measure was used.

5.2. Role and limits of the constraint mechanism

The prompt rule and programmatic clipping had separate roles. The prompt asked the model to keep each adjustment within $\pm 30\%$ of the preceding dimension score. Clipping enforced the same boundary after the response had been parsed. Three outputs exceeded the stated limit: the Phase 2 Dim2 scores in Runs 1 and 3 were clipped to 1.56, and the Phase 3 Dim2 score in Run 1 was clipped to 2.03. These cases show that the instruction alone did not enforce the boundary in every call.

After the mechanism was deployed, the four final scores had a range of 1.70 points, compared with a 5.80-point difference between the two complete baseline records. This comparison suggests that explicit boundaries restricted large inter-phase changes. It does not establish a causal effect. The baseline records were few and came from a fault-diagnosis

period that included a structured-output correction. The later configuration also increased the maximum rationale length from 80 to 200 Chinese characters. Because these changes were introduced together and the two periods were not randomized or otherwise matched, the reduction from 5.80 to 1.70 points cannot be attributed solely to clipping.

The mechanism controls how far a later score can move; it does not determine whether the resulting score is correct. If the Phase 1 score is inaccurate, a $\pm 30\%$ boundary may prevent an appropriate correction. This concern is particularly relevant for low scores because a proportional limit permits only a small absolute change. For example, a preceding score of 1.20 allows an adjustment of only 0.36 points in either direction. Whether that interval is suitable depends on the rubric and the quality of the initial score. Human ratings were not collected in this study, so the results do not show whether constrained scores were closer to a defensible benchmark.

5.3. Total-score and item-level consistency

The total-score range does not fully represent stability within the rubric. Only four of the twenty items received the same score in all four constrained runs, corresponding to an exact item-level consistency rate of 20%. Thus, sixteen items changed in at least one run, even though the final-score range was limited to 1.70 points.

Offsetting item-level changes may account for this difference. A higher score on one item can be balanced by a lower score on another, leaving the total nearly unchanged. As a result, two runs may produce similar totals while relying on different interpretations of the assignment. This distinction matters when item scores or rationales are reported to students. Similar total grades would not ensure that students receive consistent feedback about particular strengths and weaknesses.

Prior work on automated assessment commonly distinguishes aggregate agreement from criterion-level agreement. The present results follow that distinction: the aggregate score was more stable than the individual scoring decisions. However, the evidence comes from one assignment and four constrained runs. It does not show whether the 20% consistency rate is typical of other assignments or whether the observed differences would also appear when scores are compared with human raters.

For diagnostic purposes, the system should record at least four forms of variation: the total-score range, dimension-level changes, item-level agreement, and changes in the evidence cited for each score. These records serve different purposes. Total-score variation describes the result seen by the student, whereas item- and evidence-level variation helps locate changes in the model's application of the rubric.

5.4. Implications for workflow design

The results support treating Phases 2 and 3 as reviews of an existing judgment rather than unrestricted rescore. A revision prompt can require the model to identify the disputed item, cite the relevant evidence, and explain why the Phase 1 score should change. The program can then check whether the proposed adjustment falls within a predefined range.

The $\pm 30\%$ limit used here was an engineering choice, not an empirically validated grading standard. A proportional rule may be too restrictive for low initial scores and too permissive for high ones. One possible alternative is to combine a proportional limit with an absolute-point limit, but the

appropriate values would need to be tested against human ratings and a larger assignment sample.

Human review may also be warranted when a score reaches the clipping boundary, when repeated runs disagree at the item level, or when the rationale does not cite evidence relevant to the rubric. The present experiment did not test such a review process. These conditions should therefore be treated as proposed diagnostic triggers rather than validated decision rules.

6. Conclusion

This study examined repeated scoring in a three-phase LLM grading workflow using one assignment and the DeepSeek-Chat model. The system logs identified two sources of variation. First, incompatible structured-output formats interfered with initial score aggregation during the baseline period. Second, self-critique and holistic review introduced larger differences than initial ensemble scoring when later adjustments were not explicitly bounded. In the two complete baseline records, the same Phase 1 total of 3.85 led to final scores of 8.60 and 2.80.

After the system added a $\pm 30\%$ prompt rule and programmatic clipping, the four final scores were 5.40, 4.50, 3.70, and 4.20. Their range was 1.70 points, equal to 11.3% of the 15-point maximum. This range was smaller than the 5.80-point baseline difference. The result is consistent with the intended function of the mechanism: restricting large changes between phases. It does not demonstrate that the mechanism alone caused the reduction, because the baseline and constrained runs were not equivalent experimental groups and other system settings also changed.

Stability remained limited below the total-score level. The Phase 3 ranges for Dim1 and Dim2 were 1.15 and 0.83 points, compared with 0.70 and 0 in Phase 1. Only four of the twenty items received identical scores in all four constrained runs. A narrower total-score range therefore did not correspond to uniform item-level judgments.

The study supports a limited conclusion: in this single diagnostic case, explicit adjustment boundaries restricted the size of some later-phase score changes, but they did not eliminate variation across dimensions and items. The experiment did not include human ratings, so it cannot show whether clipping improved grading accuracy. It also cannot determine whether the $\pm 30\%$ threshold is appropriate for other rubrics.

The findings are limited to one assignment, one model, ten runs, and only two complete baseline records. In addition, the baseline period included fault diagnosis and system changes, which prevents a controlled before-and-after comparison. Further evaluation would require more assignments, matched experimental conditions, human-rating benchmarks, and separate tests of format correction, prompt constraints, rationale length, and programmatic clipping. Crossed or mixed-effects models could then be used to distinguish variation attributable to assignments, models, raters, and repeated calls.

Reference Documentation

- [1] Goldstein, S., et al. (2024). Can large language models grade student code? In Proceedings of ACM Technical Symposium on Computer Science Education (pp. 312–318).

- [2] Zheng, Y., et al. (2024). Large language models as judges: Stability and consistency. arXiv preprint arXiv:2402.
- [3] Liang, W., et al. (2023). Large language models can provide accurate and fair scores. In Proceedings of NeurIPS 2023 Workshop on Generative AI for Education.
- [4] Korthik, A., & Tchantchuleff, M. (2024). Position bias in LLM-based evaluation. In Proceedings of ACL 2024 (pp. 2890–2905).