

How Reproducible Are LLM-Generated Q-Matrices? Evidence from Downstream Classification

Minkwon Kim

Teachers College, Columbia University

New York, NY, USA

mkk2166@tc.columbia.edu

Abstract

Large language models (LLMs) are increasingly proposed as a way to construct Q-matrices, the expert-specified item-by-attribute mappings that drive cognitive diagnosis. Existing studies report a single generated Q-matrix or aggregate several runs by majority vote, discarding the run-to-run variation a fixed prompt produces. We ask whether that variation is benign. Regenerating the Q-matrix ten times each for three open-weight models on the released TIMSS 2011 grade-8 mathematics items and fitting the deterministic inputs, noisy “and” gate (DINA) model to real responses ($N = 741$), we find that two regenerations of the same model, at the same temperature, assign different mastery profiles to 61–88% of examinees. This instability is a property of stochastic decoding: it vanishes entirely under greedy decoding, yet the models’ disagreement with the expert Q-matrix persists even then (63–76%). Majority voting over runs does not help: the voted matrix agrees with the greedy matrix on 89–99% of cells yet still reclassifies 48–89% of examinees relative to it. Disagreement is modestly enriched on items that independent statistical validation also flags as contested. All findings replicate under an independent estimator and a second sample ($N = 559$). Single-run and majority-vote reporting render the instability invisible.

1 Introduction

Cognitive diagnosis models (CDMs) estimate which fine-grained skills, or attributes, each examinee has mastered. Every CDM depends on a *Q-matrix*: a binary matrix specifying which attributes each item requires (Tatsuoka, 1983). The Q-matrix is conventionally built by expert content review, a costly and partly subjective process, and its accuracy bounds the validity of every downstream classification (de la Torre, 2008; Rupp and Templin, 2008).

LLMs have been proposed to draft Q-matrices automatically from item text, with several recent studies reporting that open- and closed-weight models reproduce expert Q-matrices at usable rates (Zhang et al., 2026; Du et al., 2026). A recurring methodological choice unites this literature: a model is queried once, or queried several times and reduced to a consensus by majority vote, so that only one Q-matrix reaches the psychometric stage. This treats run-to-run variation as noise to be removed before analysis.

We argue that the variation is itself the quantity of interest. If the same model, given the same prompt at a fixed temperature, yields materially different Q-matrices across regenerations, and if those differences propagate into different student mastery classifications, then a single reported run misrepresents the reliability of the whole procedure. A practitioner who regenerates the matrix a day later may reclassify many students without any change to the data, the model, or the prompt.

We apply this idea to the released TIMSS 2011 grade-8 mathematics assessment, for which a published expert Q-matrix and multiple independent statistical validations exist. Our contributions are: (i) we measure run-to-run stability as a first-class, distributional outcome, separating how much an LLM disagrees with the expert from how much it disagrees with itself across regenerations; (ii) we show that within-model regeneration alone reassigns the mastery profiles of a majority of examinees, and trace this instability to stochastic decoding by showing that it disappears under greedy decoding while the expert disagreement does not, and that majority voting repairs neither; and (iii) we find that disagreement is modestly enriched on items that independent statistical validation also flags as contested. All findings replicate under an independent DINA estimator and a second analytic sample.

2 Background and Related Work

2.1 The DINA model and the Q-matrix

Under the deterministic inputs, noisy “and” gate (DINA) model (Junker and Sijtsma, 2001; de la Torre, 2009), examinee i answers item j correctly with probability governed by whether they have mastered all attributes item j requires. Two item parameters, the slip and the guess, absorb departures from this deterministic rule. The Q-matrix entry q_{jk} is 1 if item j requires attribute k . A misspecified Q-matrix biases attribute classifications, motivating both expert review and statistical validation methods (de la Torre and Chiu, 2016; Ma and de la Torre, 2020).

2.2 LLM-generated Q-matrices

Recent work prompts LLMs with item text and an attribute inventory and asks for the Q-row of each item. Zhang et al. (2026) propose a human-AI Q-matrix refinement framework in which LLMs generate candidate item-attribute mappings and a Neural CDM provides an empirical evaluation layer; they benchmark locally deployed open-weight models against cloud-hosted models and frame LLMs as decision-support tools rather than autonomous sources of ground truth. Du et al. (2026) study an EFL reading test and report that a human-AI collaborative Q-matrix, obtained by expert revision of an LLM-generated matrix, outperforms both purely expert- and purely model-constructed matrices on model fit and classification accuracy. Fan et al. (2026) compare several AI tools against a validated Q-matrix and document version drift, finding that newer model releases can agree *less* with the validated matrix than older ones.

These studies share a methodological posture that our work questions. Each evaluates a single generated Q-matrix, or a human-revised one, against an expert or validated reference; none characterizes the variation a single model produces across repeated generations under fixed conditions, nor traces that variation into the student classifications the Q-matrix ultimately drives. The version drift that Fan et al. (2026) document points to instability across model releases, but the complementary question—how much a fixed model varies from one run to the next, and whether that variation changes who is classified as having mastered what—remains open. That gap is what we address.

2.3 Statistical Q-matrix validation

Independently of LLMs, a long line of work revises expert Q-matrices from response data. For TIMSS 2011 grade-8 mathematics, Ma and de la Torre (2020) apply a stepwise G-DINA validation and Sun et al. (2025) apply four regularization- and regression-based methods to the same 23-item set, each flagging specific item-attribute cells as contested. We use these published cases as an external, non-LLM reference against which to compare where LLM instability concentrates.

3 Method

3.1 Data

We use the 23 released Block 6–7 items of the TIMSS 2011 grade-8 mathematics assessment and the seven-attribute expert Q-matrix of Park et al. (2017), the same items and attributes validated by Ma and de la Torre (2020) and Sun et al. (2025). Real item responses for United States grade-8 examinees were obtained from the IEA TIMSS 2011 International Database and scored dichotomously from the published answer key. Our primary analytic sample comprises the 741 examinees administered the relevant block; a strict complete-case sample of 559 examinees serves as a robustness check.

Park et al. (2017) developed a nine-attribute Q-matrix covering all released TIMSS 2011 grade-8 mathematics items; the 23 items in Blocks 6–7 measure seven of those nine attributes, and we adopt the validated seven-attribute specification as given in Table 9 of Ma and de la Torre (2020). The seven attributes are: whole numbers and integers; fractions, decimals and proportions; patterns; expressions, equations and functions; lines, angles and shapes; location and movement; and data organization, representation and interpretation.

3.2 Q-matrix generation

We prompt three open-weight instruction-tuned models — Qwen2.5-7B-Instruct, Mistral-7B-Instruct-v0.3, and Phi-3-medium-4k-instruct — with the seven attribute definitions and the released text of each item (answer keys removed). Each run is a single zero-shot call requesting the full 23-item matrix as a JSON object; the complete prompt is reproduced in Appendix A.

Because this paper argues that decoding configuration should be reported,

we state ours in full. Generation used HuggingFace transformers 4.46.3 (AutoModelForCausalLM.generate) on a single-GPU Google Colab runtime, with 4-bit NF4 quantization, double quantization, and bfloat16 compute dtype via bitsandbytes. The sampling condition sets temperature 1.0, top- p 0.95, and max_new_tokens 1200 explicitly, and applies no repetition penalty. Top- k was not passed and therefore fell back to the models’ own generation_config default of 20, not the transformers default of 50. We report this because the effective value differs from the one a reader would infer from the library documentation — exactly the kind of silent divergence that motivates reporting the decoding configuration in full. The greedy condition sets do_sample=False and passes no temperature or top- p . Each model generates the full matrix ten times under each condition, varying only the torch seed (0, 1000, ..., 9000), so that all variation in the sampling condition is run-to-run sampling variation; this yields 30 sampled and 30 greedy matrices. Model weights were pulled from the HuggingFace Hub at run time without pinning a revision.

3.3 Estimation and outcomes

For each generated Q-matrix we fit the DINA model to the real responses and obtain each examinee’s maximum a posteriori attribute profile. Our primary outcome is the *within-model pairwise flip rate*: for each pair of runs of the same model, the proportion of examinees whose mastery profile differs between the two runs. This isolates run-to-run instability and does not involve the expert at all. For comparison we also report the *flip rate against an expert-Q baseline*. We report distributions across runs, not point estimates. As a second outcome we record, per item, how often a generated Q-row differs from the expert row, and test whether this disagreement is concentrated on the items flagged as contested by Ma and de la Torre (2020) and Sun et al. (2025).

All fits are carried out in our own Python DINA implementation and independently reproduced with the CDM R package (George et al., 2016), a peer-reviewed implementation, to guard against estimator-specific artifacts. We report the CDM numbers throughout and use the Python results as the corroborating check. A response is accepted only if all 23 rows parse as length-7 binary

Model	Mean	SD	Min	Max	vs Exp.
Qwen2.5-7B	0.677	0.159	0.000	0.853	0.692
Mistral-7B	0.609	0.079	0.476	0.776	0.712
Phi-3-med.	0.881	0.101	0.532	1.000	0.760

Table 1: Within-model pairwise flip rate across ten regenerations at temperature 1.0 (CDM, $N = 741$). “vs Exp.” is the mean flip rate against the expert-Q baseline, shown for comparison. Two runs of the same model disagree on 61–88% of classifications.

vectors; malformed output triggers regeneration at an incremented seed, up to six attempts. No run in the final set exhausted its retries, so all 60 generations parsed successfully. An all-zero Q-row is a valid parse rather than a failure: it records that the model assigned the item no attribute. Such items are non-diagnostic and are excluded from that matrix’s calibration.

4 Results

4.1 Baseline fit and estimator agreement

The expert Q-matrix fits the real responses well: under CDM ($N = 741$) the model attains a log-likelihood of $-9,190$ (AIC 18,726; BIC 19,523) with a mean slip of 0.215 and mean guess of 0.374. Our independent Python estimator agrees closely (log-likelihood $-9,326$; mean slip 0.217), and the two implementations recover the same parameter count (173). This agreement establishes that the pipeline is correctly specified before any LLM matrices are introduced.

4.2 Result A: run-to-run instability

Table 1 reports the within-model pairwise flip rate: how often two regenerations of the *same* model, at the same temperature, assign an examinee a different mastery profile. The expert plays no role in this quantity, so it measures instability alone. Two runs of the same model disagree on the classification of 61% (Mistral) to 88% (Phi-3) of examinees on average. Phi-3 is the most volatile; strikingly, its runs disagree with one another (0.88) more than they disagree with the expert baseline (0.76). Qwen exhibits the widest spread, with run pairs ranging from 0.00 (one pair produced identical classifications) to 0.85 (another reclassified most examinees), a vivid illustration that two researchers following an identical protocol can obtain materially different diagnoses. Figure 1 shows the full distribution.

Because a profile flips if any of seven attributes changes, the full-profile rate is an upper envelope.

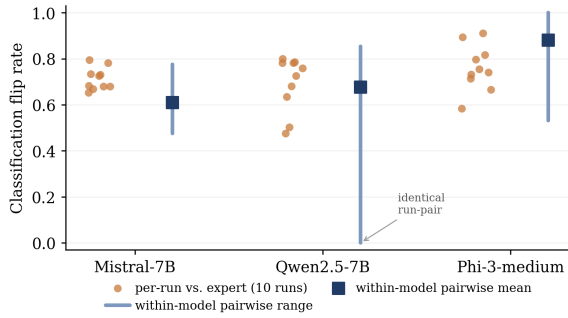


Figure 1: Classification flip rate by model. Orange points: the ten individual per-run rates against the expert baseline. Navy square: mean within-model pairwise rate (run vs. run). Blue bar: full min–max range of the pairwise rate.

The mean per-attribute flip rate against the expert baseline is lower, at 0.25, but still substantial.

4.3 Per-attribute structure of the instability

The per-attribute view also distinguishes the models qualitatively. Mistral and Qwen concentrate their instability on a few attributes and remain comparatively stable on the rest: Qwen’s per-attribute flip rate ranges from 0.09 (fractions and decimals) to 0.38 (whole numbers and integers), and Mistral shows a similar spread. Phi-3 behaves differently, with elevated and nearly uniform flip rates across all seven attributes (0.26–0.37). In other words, Phi-3’s instability is diffuse, touching every skill, whereas the other two models are unstable in localized, identifiable places. This distinction matters for practice: a model whose instability is concentrated can in principle be audited attribute by attribute, whereas diffuse instability offers no such foothold.

4.4 The instability is a decoding effect

Regenerating each model under greedy (deterministic) decoding eliminates the instability entirely: all ten runs of every model produce an identical Q-matrix, so the within-model pairwise flip rate is exactly zero. The ten greedy outputs are byte-for-byte identical within each model. We verify this at the level of the generated matrices, on a single device, and make no claim about bitwise-identical logits. The run-to-run instability of Section 4.2 is therefore a consequence of sampling at non-zero temperature, the prevailing default in the LLM-Q-matrix literature, rather than a deeper non-determinism. Crucially, the models’ disagreement with the expert Q-matrix does *not* vanish under

greedy decoding: it remains at 0.63 (Mistral), 0.73 (Phi-3), and 0.76 (Qwen). Determinism removes the variance between runs but not the structural disagreement with expert judgment.

This second observation deserves emphasis. Eliminating run-to-run variation does not make the generated Q-matrix correct; it makes the disagreement with expert judgment a particular and reproducible one. Even at temperature zero, the three models reassign 63–76% of mastery profiles relative to the expert baseline. Determinism thus replaces an unstable disagreement with a stable one, which is an improvement for reproducibility but leaves the question of validity untouched.

4.5 Majority voting does not remove the instability

Majority voting across runs is the prevailing remedy for run-to-run variation in this literature. We construct element-wise majority Q-matrices (threshold 5 of 10) from the ten sampled runs per model and fit them exactly as before. Voting does not yield a better point estimate: its flip rate against the expert equals or exceeds the mean single run for every model (Table 2).

Its relationship to greedy decoding is more revealing. At the cell level the voted matrix nearly reproduces the greedy one, agreeing on 89–99% of the 161 item-attribute cells. Yet those few differing cells reassign the mastery profiles of 48% (Qwen), 56% (Mistral), and 89% (Phi-3) of examinees relative to greedy. Two matrices that are all but identical on paper therefore diverge sharply downstream — a recursive instance of this paper’s central finding, now applied to the remedy rather than to the problem. Cell-level unanimity tracks the same ordering as the flip rates: all ten sampled runs agree on 89% of cells for Qwen, 61% for Mistral, and 16% for Phi-3. Majority voting neither measures the instability nor removes it.

4.6 Result B: where disagreement concentrates

Pooling the 30 sampled matrices, mean per-item disagreement with the expert Q-row is 0.847 on the five items flagged as contested by multiple statistical validation methods (items 5, 11, 15, 21, 23) versus 0.659 on the remaining items. A permutation test that randomly reassigns the contested label gives a one-sided p of 0.060, suggestive but not significant at conventional levels, as expected with only five contested items. A second, inde-

Model	Flip vs expert			Vote vs greedy	
	Run	Vote	Greedy	Cells	Flip
Qwen2.5-7B	0.692	0.807	0.758	0.99	0.48
Mistral-7B	0.712	0.718	0.633	0.93	0.56
Phi-3-med.	0.760	0.885	0.731	0.89	0.89

Table 2: Majority voting compared with single runs and with greedy decoding. “Run” is the mean per-run flip rate against the expert baseline; “Vote” is the element-wise majority matrix. “Cells” is the proportion of the 161 item-attribute cells on which the voted and greedy matrices agree; “Flip” is the proportion of examinees they classify differently.

pendent angle agrees: the share of each model’s differing items that fall on the five contested items exceeds the chance baseline of 0.217 for all three models (Mistral 0.30, Qwen 0.29, Phi-3 0.23). The enrichment is weakest for Phi-3, the most unstable model. We therefore read Result B as directional, mutually corroborating evidence that part of the disagreement reflects genuine ambiguity in the measurement problem, rather than as a confirmed effect.

4.7 Robustness

All results reproduce under the independent Python estimator and under the $N = 559$ sample. CDM flip rates change by at most two percentage points between samples, the ordering of models by volatility is preserved, and the contested-versus-other item gap (0.847 vs. 0.659) is identical across both samples and both estimators because it is a structural property of the generated matrices. A secondary observation concerns completeness: under sampling, models left at least one item unassigned in 17 of 30 runs, with Phi-3 once omitting nine of 23 items. These are genuine all-zero Q-rows, not parse failures; every generation in the final set parsed successfully.

5 Discussion

The practical implication is direct. Most LLM-Q-matrix studies sample at non-zero temperature and report a single run or a majority vote. Under those standard settings, our results show that a second run of the same model reassigns the mastery profiles of most examinees, so a reliability claim resting on one run does not characterize what a re-run would produce, and two researchers following an identical protocol can report different student diagnoses. Majority voting, the prevailing remedy, hides this

variance rather than measuring it, and Section 4.5 shows it does not remove it either: the voted matrix is nearly the greedy matrix on paper, yet reclassifies half to nine-tenths of students relative to it. Reporting the distribution is the minimal honest alternative.

Our temperature analysis localizes the cause and points to a mitigation: the instability is a sampling effect and disappears under greedy decoding. This is not a reason to dismiss it, because greedy decoding is neither the default in the existing literature nor always available in deployed systems, but it is an actionable recommendation, namely that Q-matrix generation be run deterministically and that the decoding configuration be reported. Determinism does not, however, resolve the more basic issue: even under greedy decoding the models disagree with the expert Q-matrix on most classifications, so deterministic generation yields a stable answer that is not necessarily a correct one.

The convergent-validity analysis tempers a purely negative reading. Disagreement is enriched, above chance, on items that independent statistical methods also contest, which suggests that some of the disagreement localizes genuinely underdetermined item-attribute specifications rather than reflecting arbitrary error. The evidence is directional rather than conclusive, given the small number of contested items and the weak enrichment for the most unstable model. Read cautiously, it motivates a constructive use of regeneration: not as a one-shot oracle, but as a screening signal that flags items warranting expert attention, used alongside statistical validation rather than in place of it.

Three practical recommendations follow. First, fix the decoding configuration and report it; deterministic decoding adds no computational overhead. Second, never treat a single sampled run as the final Q-matrix. If sampling is used, report the distribution of downstream classification results over multiple runs, since reporting only a mean conceals how wide the spread around it is. Third, use regeneration not as an oracle but as a diagnostic instrument: the items on which models are least stable, and on which they deviate most from expert judgment, overlap with the items that independent statistical validation also flags, and are therefore the items that most warrant human review.

Limitations. We study one assessment, one CDM (DINA), and three open-weight models. Closed-weight models, other CDMs, and other

prompt phrasings may behave differently. Three items reference figures that the text-only prompt cannot convey, which may inflate their instability. Importantly, the flip rate is computed against an expert baseline that is itself imperfect, and that independent statistical validations find contestable on specific items; it therefore measures disagreement with expert judgment, not error against ground truth. We examine sampling temperature directly but vary it only between full sampling and greedy decoding; intermediate temperatures and alternative decoding strategies remain to be mapped.

Two estimation choices warrant note. First, when a generated matrix leaves an item non-diagnostic (an all-zero Q-row), we exclude that item from the calibration, so generated matrices are occasionally fit on slightly different item sets; we therefore base our central claims on classification flip rates rather than on cross-matrix comparisons of absolute fit. Second, the flip rate counts any change in an examinee's maximum a posteriori profile and is not weighted by posterior certainty, so a flip driven by a near-50/50 attribute is counted equally with a decisive one. The per-attribute analysis above partially addresses this by showing that flips are distributed across attributes rather than concentrated in a single unstable one.

6 Conclusion

Treating the reproducibility of LLM-generated Q-matrices as a distributional, downstream question reveals instability that single-run and majority-vote reporting conceal: under standard sampling, two regenerations of the same model reassign the mastery profiles of most examinees. The instability is a decoding effect, removable by deterministic generation, but the models' disagreement with expert judgment is structural and survives that fix, concentrating modestly on independently contested items. Majority voting, the standard remedy, does neither job: it is no better than a single run against the expert, and it reclassifies most students relative to the greedy matrix it almost exactly reproduces. We recommend that studies using LLMs for Q-matrix construction generate deterministically, report their decoding configuration, and report the distribution of downstream classifications rather than a single run, treating regeneration as a validation signal rather than a one-shot result.

Data and Code Availability

The TIMSS 2011 released items and the IEA International Database are publicly available. Our generation scripts, the full prompt, the DINA estimator, the CDM validation script, and all 60 generated Q-matrices are available from the author on request. The scored response matrix is derived from the public IEA database; our scoring script reproduces it from the released files.

References

- Jimmy de la Torre. 2008. [An empirically based method of Q-matrix validation for the DINA model: Development and applications](#). *Journal of Educational Measurement*, 45(4):343–362.
- Jimmy de la Torre. 2009. [DINA model and parameter estimation: A didactic](#). *Journal of Educational and Behavioral Statistics*, 34(1):115–130.
- Jimmy de la Torre and Chia-Yi Chiu. 2016. [A general method of empirical Q-matrix validation](#). *Psychometrika*, 81(2):253–273.
- W. Du, Jiayi Shen, and Xiaomei Ma. 2026. [Applying GenAI to optimize Q-matrix construction for cognitive diagnostic assessment in EFL reading](#). *Journal of Intelligence*, 14(5):79.
- Kevin Fan, Jacquelyn A. Bialo, and Hongli Li. 2026. [The use of AI tools to develop and validate Q-matrices](#). ArXiv:2602.08796. An earlier version appeared in the Proceedings of the 90th Annual Meeting of the Psychometric Society, Minneapolis, 2025.
- Ann Cathrice George, Alexander Robitzsch, Thomas Kiefer, Jürgen Groß, and Ali Ünlü. 2016. [The R package CDM for cognitive diagnosis models](#). *Journal of Statistical Software*, 74(2).
- Brian W. Junker and Klaas Sijtsma. 2001. [Cognitive assessment models with few assumptions, and connections with nonparametric item response theory](#). *Applied Psychological Measurement*, 25(3):258–272.
- Wenchao Ma and Jimmy de la Torre. 2020. [An empirical Q-matrix validation method for the sequential generalized DINA model](#). *British Journal of Mathematical and Statistical Psychology*, 73:142–163.
- Jung Yeon Park, Young-Sun Lee, and Matthew S. Johnson. 2017. [An efficient standard error estimator of the DINA model parameters when analysing clustered data](#). *International Journal of Quantitative Research in Education*, 4(1/2):159–190.
- André A. Rupp and Jonathan Templin. 2008. [The effects of Q-matrix misspecification on parameter estimates and classification accuracy in the DINA model](#). *Educational and Psychological Measurement*, 68(1):78–96.

Xiaojian Sun, Tongxin Zhang, Chang Nie, Naiqing Song, and Tao Xin. 2025. [Combining regularization and logistic regression model to validate the Q-matrix for cognitive diagnosis model](#). *British Journal of Mathematical and Statistical Psychology*, 78(1):1–21.

Kikumi K. Tatsuoka. 1983. [Rule space: An approach for dealing with misconceptions based on item response theory](#). *Journal of Educational Measurement*, 20(4):345–354.

Ying Zhang, Ningxi Cheng, Yizhu Gao, Hongmei Li, Lehong Shi, Nicholas Young, Geng Yuan, and Xiaoming Zhai. 2026. [A framework for human-AI Q-matrix refinement: A NeuralCDM evaluation](#). ArXiv:2604.16398. Accepted at AIED 2026.

A Prompt Template

The prompt below is reproduced as issued. One call per run requested the full 23-item matrix; no in-context examples were given.

System prompt.

You are an expert in mathematics education and cognitive diagnostic assessment. You construct Q-matrices: binary matrices specifying which fine-grained math skills each test item requires. You answer ONLY with the requested JSON.

Attribute definitions. These were inserted into the user prompt at the point marked {attribute definitions block}:

- skill11 — Whole numbers and integers: operations, place value, properties of integers.
- skill12 — Fractions, decimals and proportions: equivalence, ordering, operations, ratio and percent.
- skill13 — Patterns: recognizing, extending, generalizing numeric or geometric patterns and sequences.
- skill14 — Expressions, equations and functions: algebraic expressions, solving equations, functional relationships.
- skill15 — Lines, angles and shapes: geometric properties, angle measure, properties of 2D/3D figures.
- skill16 — Location and movement: coordinates, transformations, rotations, reflections, translations.
- skill17 — Data organization, representation and interpretation: tables, graphs, charts; basic probability.

User prompt.

These TIMSS grade-8 mathematics items each measure one or more of seven skills:

{attribute definitions block}

For EACH of the following 23 items, decide which skills are REQUIRED to answer it correctly. Assign 1 if necessary, 0 if not.

Items:

{numbered list of the 23 item stems, answer keys removed}

Return ONLY a JSON object: {"1": [s1, s2, s3, s4, s5, s6, s7], ..., "23": [...]} where each s is 0 or 1 in order skill1..skill7. Output only the JSON, no other text.

The 23 item stems were the released TIMSS 2011 Block 6–7 stems with correct-answer keys removed.