

1                   **It’s Hard to Know How Hard It Is:**  
2                   **Mapping the Design Space of LLM Item**  
3                   **Difficulty Estimation**

4                   Anonymous

5                   Anonymous Institution

6                   **Abstract.** Can LLMs estimate how hard a test question is? Published  
7                   results disagree wildly—correlations range from near-zero to  $r = 0.87$ —  
8                   but each study typically tests one prompt on one model on one dataset.  
9                   We map the design space systematically: 15 prompts, 6 models, 3 datasets  
10                  ( $\sim 120$  configurations). Prompts that direct the model to enumerate struc-  
11                  tural item features—prerequisites, cognitive load, error paths—achieve  
12                   $\rho = 0.65\text{--}0.69$  on open-ended items. This advantage attenuates on MCQs  
13                  ( $\rho \approx 0.45\text{--}0.58$ ), where a simple teacher-role prompt nearly matches.  
14                  Prompt design is the primary lever; model size, temperature, and delib-  
15                  eration each contributes minimally ( $\eta^2 < 0.05$ ). Methodologically, cor-  
16                  relations from small item sets ( $n < 50$ ) are unreliable, and unbalanced  
17                  hyperparameter sweeps produce Simpson’s paradox artifacts. At  $\sim \$0.01$   
18                  per item, LLM estimates can triage items into difficulty bands, but the  
19                  signal is strongest for open-ended items spanning knowledge domains.

20                  **Keywords:** item difficulty estimation · LLM evaluation · design of ex-  
21                  periments · psychometrics · prompt engineering

22                  **1 Introduction**

23                  A well-designed test gives each student questions at the right level of challenge—  
24                  hard enough to reveal what they know, easy enough to avoid frustration. Getting  
25                  this right depends on knowing how difficult each question is before it reaches a  
26                  student. In practice, this means field-testing items with hundreds or thousands  
27                  of real students and computing how many answer correctly. In classical test  
28                  theory, this proportion is called  $p$ -correct; item response theory refines it with a  
29                  latent difficulty parameter  $b$  that accounts for examinee ability [8]. Either way,  
30                  calibrating each item requires collecting responses from 200 to 1,000+ students  
31                  depending on the IRT model [5], making difficulty estimation one of the most  
32                  resource-intensive steps in assessment development.

33                  That cost is becoming a bottleneck. LLMs can now generate assessment items  
34                  at scale [15], but generated items arrive without calibrated difficulty estimates.  
35                  Without such estimates, items cannot be assembled into well-targeted tests,  
36                  used in adaptive systems, or compared to existing item banks. The question  
37                  is whether LLMs can also estimate difficulty—replacing or supplementing the  
38                  expensive field-testing step.

39 The task is harder than it appears. LLMs are trained on expert-written text  
 40 and tend to find most items trivial; Li, Chen et al. [13] find that high model per-  
 41 formance paradoxically impedes difficulty estimation—models that solve items  
 42 easily cannot perceive what makes them hard for humans. This “curse of knowl-  
 43 edge” creates a systematic bias: models overestimate student ability and com-  
 44 press difficulty estimates toward the easy end of the scale. Expert teacher judg-  
 45 ment, by comparison, achieves correlations of  $r = 0.63\text{--}0.66$  with actual stu-  
 46 dent performance [11,19]—a useful reference point, though these studies measure  
 47 judgment of individual students rather than aggregate item difficulty.

48 Recent work shows wide variation: some studies report strong results (up  
 49 to  $r = 0.87$  with feature extraction and gradient boosting [17]), while others  
 50 find near-zero correlations for direct estimation [1]. A systematic review of 37  
 51 papers found no consensus on methods [16], and the first shared task on difficulty  
 52 prediction (BEA 2024) found best results only marginally beat baselines [22]. Li,  
 53 Chen et al. [13] evaluated 20+ models and found they converge on a “machine  
 54 consensus” that diverges from human perception.

55 A key limitation is that each study typically tests one model with one prompt  
 56 on one dataset. When results disagree, it is unclear whether the discrepancy  
 57 reflects model choice, prompt design, item properties, or student population.  
 58 We address this by mapping the design space rather than testing any single  
 59 configuration.

60 Our optimization target is **rank-order agreement** between LLM-predicted  
 61 difficulty and empirical  $p$ -correct, measured by Spearman’s  $\rho$ . We choose rank  
 62 correlation because practical applications—item sequencing, adaptive test as-  
 63 sembly, item bank stratification—depend on difficulty *ordering* rather than ex-  
 64 act calibration. Each experimental condition prompts an LLM to estimate the  
 65 proportion of students who would answer each item correctly; we then correlate  
 66 these estimates against observed  $p$ -correct from real student responses.

67 *Contributions.* (1) The first systematic comparison of 15 theory-grounded prompts  
 68 for LLM difficulty estimation, tested across 6 models and 3 datasets ( $\sim 120$  config-  
 69 urations). (2) An *item-centric vs. population-centric* distinction explaining why  
 70 hybrid prompts like `buggy_rules` ( $\rho = 0.66$ ) succeed while pure simulation (`syn-  
 71 thetic_students`,  $\rho = 0.19$ ) fails. (3) Methodological guidance: Simpson’s paradox  
 72 artifacts in unbalanced hyperparameter sweeps,  $\geq 100$  items required for reliable  
 73 validation, and two-stage sequential DOE as the recommended design.

## 74 2 Related Work

75 We organize prior work by approach and extract testable claims from each.

76 *Direct LLM Estimation.* Razavi & Powers [17] report  $r = 0.83$  (math) and  
 77  $r = 0.81$  (reading) on K–5 items using GPT-4o. However, Acquaye et al. [1] find  
 78  $r \approx 0$  for direct estimation on NAEP items, and Li, Chen et al. [13] find average  
 79  $\rho = 0.28$  across 20+ models on four assessment domains. *Testable claim: direct*

80 *estimation produces meaningful correlations ( $\rho > 0.30$ ) on curriculum-aligned  
81 items.*

82 *Feature Extraction + ML.* Razavi & Powers [17] achieve their strongest results  
83 ( $r = 0.87$ ) by extracting features from LLMs and training gradient-boosted  
84 models. *Testable claim: LLM-extracted features combined with ML outperform  
85 direct estimation.*

86 *Simulated Classrooms.* Acquaye et al. [1] simulate classrooms of LLM students  
87 at multiple ability levels, achieving  $r = 0.75\text{--}0.82$  on NAEP items. Counter-  
88 intuitively, weaker math models predicted difficulty better than stronger ones.  
89 However, Li, Chen et al. [13] find significant misalignment between AI and human  
90 difficulty perception. *Testable claim: simulation outperforms direct estimation.*

91 *Model Uncertainty.* Zotos et al. [23] use LLM uncertainty features (first-token  
92 probability, choice-order sensitivity) to predict difficulty. *Testable claim: items  
93 the model finds uncertain are items students find difficult.*

94 *Reasoning Augmentation.* Feng et al. [9] report up to 28% MSE reduction when  
95 generating reasoning before predicting difficulty. However, Li, Jiao et al. [14] find  
96 minimal improvements from chain-of-thought prompting for fine-tuned models.  
97 *Testable claim: structured deliberation improves difficulty estimation.*

98 *Cognitive Item Models.* A separate tradition predicts difficulty from structural  
99 item features. Embretson's [7] cognitive design system counts processing steps  
100 and working memory demands. Gorin [10] manipulates item features to generate  
101 items at target difficulties. The KLI framework [12] posits that the number and  
102 nature of knowledge components drive learning difficulty. Recent work has used  
103 LLMs for KC tagging [15], but not to operationalize cognitive item models as  
104 difficulty estimation prompts. *Testable claim: prompts grounded in cognitive item  
105 modeling outperform atheoretical prompts.*

### 106 3 Hypotheses

107 We derive hypotheses from three sources: replication of prior claims, predictions  
108 from learning science theory, and exploratory questions about the design space.  
109 Table 1 lists all 12 hypotheses with their sources and predictions.

## 110 4 Method

### 111 4.1 Datasets

112 *SmartPaper (India, Open-ended)—Primary.* 140 open-ended questions from In-  
113 dian state assessments across four subjects (English, Mathematics, Science, So-  
114 cial Science), Grades 6–8, with 728,000+ responses. Ground truth: classical  $p$ -  
115 correct (proportion scoring full marks). Difficulty range: 0.04–0.83, mean 0.29.

Table 1: Hypotheses tested in this study. Verdicts: ✓ = supported, ✗ = rejected, ~ = partial/mixed.

#	Hypothesis	Source/Theory	Prediction	Verdict
<i>From Prior Work (Replication)</i>				
H1	Direct estimation yields $\rho > 0.30$	Razavi & Powers	$\rho > 0.30$	✓ .56
H2	Simulation beats direct	Acquaye et al.	$\rho_{\text{sim}} > \rho_{\text{direct}}$	✗ .19<.56
H3	Larger models do better	Scaling expectation	Larger → higher $\rho$	~ mixed
<i>From Learning Science Theory</i>				
H4	Item analysis beats direct	Embretson (1998)	$\rho_{\text{analysis}} > \rho_{\text{direct}}$	✓ .69>.56
H5	Prerequisite counting works	KLI (Koedinger)	More prereqs → harder	✓ .69
H6	Cognitive load counting works	CLT (Sweller)	More elements → harder	✓ .67
H7	Buggy rules analysis works	Brown & Burton	More error paths → harder	✓ .66
<i>Exploratory (Design Space)</i>				
H8	Temperature enhances predictions	Stochastic diversity	Higher temp → higher $\rho$	✗ $\eta^2=.001$
H9	Multi-sample averaging helps	Wisdom of crowds	$\rho_{3\text{-rep}} > \rho_{1\text{-rep}}$	~ +.03
H10	Prompt is primary lever	Design space	Prompt > model + temp	✓ on open; ~ on MCQ
H11	Analysis needs capable models	Capacity limits	Analysis × capability	✓ interaction
H12	Signal transfers across datasets	Generalizability	Signal on D1 → D2	✓ attenuates

Table 2: Experimental stages.

Stage	Purpose	Dataset	Design
Screening	Map prompt space	SmartPaper (140)	15 prompts × 2–5 temps × 3 reps
Model survey	Model generalization	SmartPaper (140)	6 models × 3 prompts × 3 reps
Confirmation	Cross-dataset transfer	DBE-KT22 (168)	3 prompts × 2 temps × 3 reps
Validation	Published benchmark	BEA 2024 (595)	3 prompts × 1 temp × 3 reps

<sup>116</sup> *DBE-KT22 (South Africa, MCQ)—Confirmation.* 168 undergraduate computer  
<sup>117</sup> science MCQs spanning 27 knowledge components, administered to 1,300+ stu-  
<sup>118</sup> dents. Ground truth: classical  $p$ -correct. Differs from SmartPaper in format,  
<sup>119</sup> domain, and population.

<sup>120</sup> *BEA 2024 Shared Task (USA, USMLE MCQ)—Validation.* 595 text-only items  
<sup>121</sup> from the BEA 2024 automated difficulty prediction shared task [22], compris-  
<sup>122</sup> ing USMLE Steps 1–3 questions (667 total items minus 72 requiring images).  
<sup>123</sup> Ground truth: IRT difficulty parameters from operational administrations. This  
<sup>124</sup> benchmark allows direct comparison with other difficulty estimation approaches.

## <sup>125</sup> 4.2 Design Space

## <sup>126</sup> 4.3 Prompts

<sup>127</sup> All 16 prompts share a common output: the LLM predicts what proportion of  
<sup>128</sup> students would answer each item correctly. They differ along two binary di-

Table 3: Prompt taxonomy with factor coding. Item = analyzes item structure; Pop. = models student population. All 15 prompts except synthetic\_students were tested in screening.

Prompt	Item	Pop.	Theory/Prior Work
teacher			PCK (Shulman, 1986)
verbalized_sampling			Wisdom of crowds
familiarity_gradient	✓		Transfer distance
contrastive	✓		Comparative judgment
prerequisite_chain	✓		KC theory (Koedinger et al.)
cognitive_load	✓		CLT (Sweller, 1988)
cognitive_profile	✓		CLT + profiling
devil_advocate		✓	Debiasing heuristics
teacher_decomposed		✓	IRT stratification
classroom_sim		✓	Acquaye et al. (2025)
imagine_classroom		✓	Imagery-based reasoning
synthetic_students*		✓	Student simulation
error_analysis	✓	✓	Error analysis tradition
error_affordance	✓	✓	Brown & Burton (1978)
buggy_rules	✓	✓	Brown & Burton (1978)
misconception_holistic	✓	✓	Chi et al. (1994)

\*Tested in model survey only ( $\rho = 0.19$ , worst performer).

129 dimensions: whether the prompt directs the model to *analyze structural item fea-*  
 130 *tures* (prerequisites, steps, cognitive demands) before estimating, and whether  
 131 it directs the model to *model the student population* (reason about proficiency  
 132 levels, struggles). Table 3 lists all 16 prompts; 15 were tested in screening (syn-  
 133 *thetic\_students* was added in the model survey). All prompts are available in  
 134 the project repository.

#### 135 4.4 Models

136 Six models spanning 8B to frontier-scale: Gemini 3 Flash (screening model),  
 137 GPT-4o, Llama-3.3-70B, Gemma-3-27B, Llama-4-Scout (17B active/109B MoE),  
 138 and Llama-3.1-8B. Qwen3-32B was tested but excluded due to low parse rates  
 139 (<50% valid responses).

#### 140 4.5 Statistical Approach

141 Our primary metric is Spearman’s  $\rho$  between LLM-predicted and observed  $p$ -  
 142 correct. We report 95% bootstrap confidence intervals (10,000 resamples) for  
 143 primary results and MAE as a calibration measure. Each condition uses 3 repli-  
 144 cations; reported  $\rho$  is the averaged-prediction correlation (mean prediction across  
 145 reps correlated with ground truth).

Table 4: Prompt screening results (SmartPaper,  $n = 140$ ): top 10 of 15 screened prompts, ranked by best  $\rho$ . Five additional prompts (verbalized\_sampling, familiarity\_gradient, imagine\_classroom, irt\_sim, kc\_mastery) achieved  $\rho < 0.55$ . CIs shown for prompts advanced to model survey.

Prompt	Item Pop.	Best $\rho$	Temp	95% CI	MAE	
<b>prerequisite_chain</b>	✓	<b>0.686</b>	0.5	[.576,.775]	.155	
<b>cognitive_load</b>	✓	<b>0.673</b>	2.0	[.550,.766]	.190	
<b>buggy_rules</b>	✓	✓	<b>0.655</b>	1.0	[.532,.752]	.117
misconception_holistic	✓	✓	0.636	2.0	—	.204
error_analysis	✓	✓	0.596	2.0	—	.121
devil_advocate		✓	0.596	1.0	—	.098
cognitive_profile	✓		0.586	1.0	[.456,.693]	.214
contrastive		✓	0.584	1.0	—	.123
classroom_sim		✓	0.562	2.0	—	.240
teacher			0.555	1.0	[.422,.664]	.439

## 146 5 Results

147 We present results in five parts: the headline finding on item analysis prompts,  
 148 the mechanism behind it, factors that matter less than expected, generalization  
 149 across three datasets, and practical implications.

150 

### 5.1 Item Analysis Prompts Achieve $\rho = 0.65\text{--}0.69$

151 Fifteen prompts were screened on SmartPaper (140 open-ended items) using  
 152 Gemini 3 Flash at 2–5 temperature settings with 3 replications per condition.  
 153 The top three prompts—all directing the model to analyze structural item fea-  
 154 tures before estimating difficulty—achieved  $\rho = 0.65\text{--}0.69$  (Table 4, Figure 1):  
 155 **prerequisite\_chain** ( $\rho = 0.686$ ) enumerates knowledge prerequisites before  
 156 estimating; **cognitive\_load** ( $\rho = 0.673$ ) counts interacting memory elements;  
 157 and **buggy\_rules** ( $\rho = 0.655$ ) identifies procedural error paths. For context,  
 158 meta-analyses report that experienced teachers can predict individual student  
 159 performance with  $r = 0.63\text{--}0.66$  [11,19]. The constructs differ: teachers judge in-  
 160 dividual students while we predict aggregate item difficulty. Direct comparison  
 161 is inappropriate, but the magnitude suggests LLM estimates capture meaningful  
 162 difficulty information.

163 

### 5.2 The Mechanism: Item Analysis, Not Population Modeling

164 Grouping prompts by their  $2\times 2$  factor membership (Table 3) reveals a clear  
 165 pattern: item analysis is the primary driver. Prompts that model student pop-  
 166ulations without analyzing item structure perform *worse* than baseline. Mean  
 167  $\rho$  by group: item analysis only = 0.61 (4 prompts); item + population = 0.59  
 168 (5 prompts); neither factor = 0.52 (3 prompts); population only = 0.47 (4 prompts).

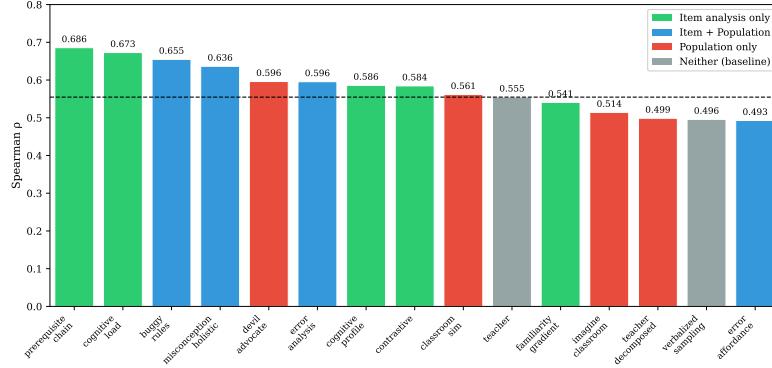


Fig. 1: Prompt screening results on SmartPaper ( $n = 140$ ). Color indicates factor membership: green = item analysis only, blue = item + population, red = population only, grey = neither. Dashed line = teacher-role prompt baseline. Error bars show 95% bootstrap CIs.

169 This parallels Embretson’s [7] cognitive design system, where item difficulty is  
 170 predicted from counts of processing steps and working memory demands. The  
 171 LLM operationalizes similar constructs when directed to enumerate prerequisites  
 172 or interacting elements.

173 The critical distinction is *item-centric* vs. *population-centric* framing. Hybrid  
 174 prompts like *buggy\_rules* ( $\rho = 0.66$ ) succeed because “identify procedural errors  
 175 students might make” is fundamentally item analysis—it asks what error paths  
 176 the item affords, not how a simulated population would respond. Population  
 177 modeling *alone* is unreliable: explicit student simulation (*synthetic\_students*:  
 178  $\rho = 0.19$ ) performs worst, consistent with Li, Chen et al.’s [13] finding that  
 179 proficiency simulation does not reliably improve difficulty estimation. Framing  
 180 analysis through student behavior works; simulating student populations does  
 181 not.

### 182 5.3 What Doesn’t Matter

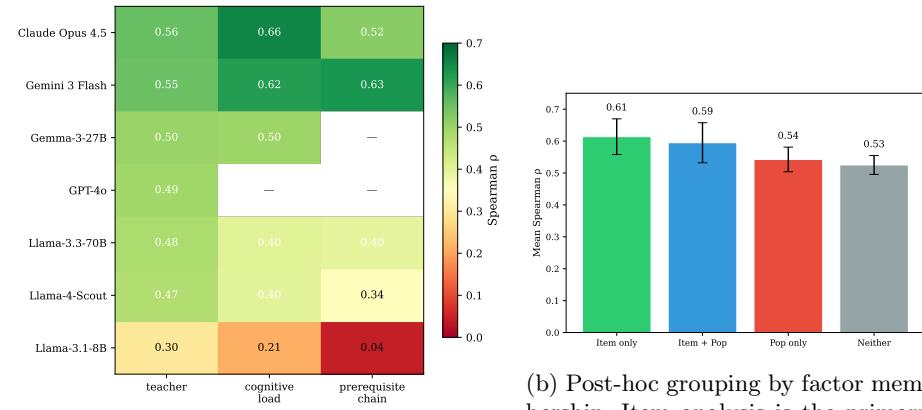
183 Three factors had minimal effects (each unique  $\eta^2 < 0.05$ ):

184 *Temperature*. Most prompts show  $\Delta\rho < 0.05$  across temperatures 0.5–2.0. The  
 185 exception: *prerequisite\_chain* peaks at temperature 0.5 while *cognitive\_load*  
 186 peaks at 2.0.

187 *Model size*. Among frontier models, size does not predict success. GPT-4o ( $\rho =$   
 188 0.49) underperforms Gemini 3 Flash ( $\rho = 0.55$ ) and matches the smaller Gemma-  
 189 3-27B. A capacity floor exists—Llama-3.1-8B struggles ( $\rho = 0.30$ )—but above  
 190 that threshold, other factors dominate (Table 5).

Table 5: Model survey ( $\rho$ , SmartPaper,  $t = 1.0$ ). “—” indicates condition not run or <50 valid responses.

Model	Params	teacher	cog_load	prereq
Gemini 3 Flash	—	0.55	0.62	<b>0.66</b>
Gemma-3-27B	27B	0.50	0.50	—
GPT-4o	—	0.49	—	—
Llama-3.3-70B	70B	0.48	0.40	0.40
Llama-4-Scout	17B/109B	0.47	0.40	0.35
Llama-3.1-8B	8B	0.30	0.21	0.04



(a) Model survey. Frontier models achieve  $\rho > 0.50$ ; Llama-8B fails on structured prompts.

(b) Post-hoc grouping by factor membership. Item analysis is the primary driver; population modeling alone underperforms baseline.

Fig. 2: (a) Model  $\times$  prompt heatmap on SmartPaper. (b) Mean  $\rho$  by  $2 \times 2$  factor membership with 95% CIs.

191 *Deliberation.* Undirected chain-of-thought reasoning degrades performance. The  
 192 structured prompts that work use *directed* analysis—they specify exactly what to  
 193 enumerate (prerequisites, interacting elements, error paths). Undirected “think  
 194 step by step” produces unfocused reasoning that drifts toward solving the item.

195 *Surface features.* Text length correlates  $\rho = -0.44$  with difficulty on SmartPaper  
 196 (shorter questions are harder). Controlling for text length, partial  $\rho = 0.59\text{--}0.66$  for item analysis prompts—the correlation changes minimally ( $\Delta\rho \leq 0.06$ ),  
 197 confirming that LLM predictions are not merely proxies for question length.

198 *Model  $\times$  prompt interaction.* Item analysis prompts help capable models but  
 199 hurt weak ones. On Gemini, prerequisite\_chain gains 0.08 over teacher; on  
 200 Llama-8B, it loses 0.26.

Table 6: Cross-dataset generalization (Gemini 3 Flash, best temperature per dataset).

Dataset	<i>n</i>	teacher	prereq	buggy	Best	95% CI
SmartPaper (open-ended)	140	0.56	<b>0.69</b>	0.66	[.58,.78]	
DBE-KT22 (CS MCQ)	168	0.52	0.53	<b>0.58</b>	[.46,.68]	
BEA 2024 (USMLE MCQ)	595	0.45	0.44	<b>0.45</b>	[.39,.52]	

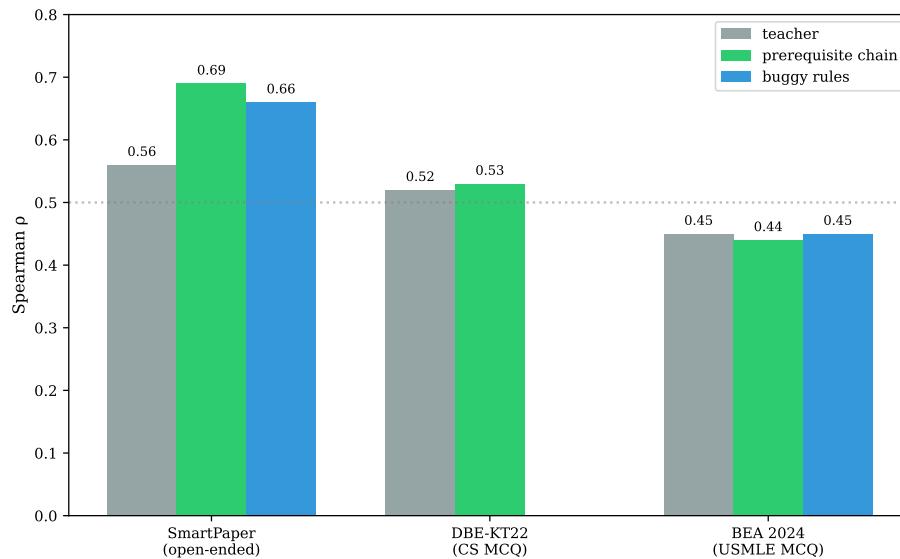


Fig. 3: Cross-dataset generalization. The item-analysis advantage (prerequisite\_chain, green) is strongest on open-ended items and attenuates on MCQs, where the teacher-role prompt (grey) nearly matches structured prompts.

#### 202 5.4 Generalization Across Datasets

203 We validated on two additional datasets: DBE-KT22 (168 undergraduate CS  
 204 MCQs) and the BEA 2024 shared task (595 USMLE medical MCQs with IRT  
 205 difficulty). Table 6 shows results.

206 The signal transfers: all prompts achieve significant correlations on all three  
 207 datasets. However, the item analysis advantage attenuates on MCQs. On Smart-  
 208 Paper, prerequisite\_chain beats teacher by 0.13; on DBE-KT22, buggy\_rules  
 209 achieves the highest  $\rho = 0.58$  but the gap over teacher is only 0.06; on BEA, all  
 210 three prompts are equivalent ( $\rho \approx 0.45$ ).

211 This suggests a **structural predictability hypothesis**: LLMs estimate dif-  
 212 ficulty well when it varies across knowledge domains (open-ended items span-  
 213 ning topics), but the advantage of counting prompts diminishes when difficulty is

214 driven by distractor design within a domain. The teacher-role prompt is equally  
215 effective for MCQs.

216 *BEA 2024 Benchmark Comparison.* The BEA 2024 shared task [22] evaluated  
217 difficulty prediction on 667 USMLE items using RMSE on IRT difficulty as the  
218 primary metric. The winning system (EduTec) achieved RMSE=0.299, barely  
219 improving over a mean-prediction baseline of 0.311; the best post-competition  
220 result (UnibucLLM, SVR+BERT features [18]) achieved RMSE=0.281 with  
221 Kendall  $\tau = 0.28$ . Our zero-shot predictions show systematic bias: models over-  
222 estimate student ability. However, when calibrated via linear scaling on the BEA  
223 training set, our three-prompt ensemble (averaging teacher, prerequisite\_chain,  
224 buggy\_rules) achieves RMSE=0.280 and  $\tau = 0.31$ . This matches or slightly  
225 exceeds the best published result—despite using prompts designed for K–8 ed-  
226 ucation rather than medical licensing exams. This suggests that zero-shot rank-  
227 order signal, combined with simple calibration, can match supervised feature-  
228 extraction approaches.

### 229 5.5 Practical Implications

230 At approximately \$0.01 per item (3 replications  $\times$  Gemini 3 Flash at \$0.50/\$3.00  
231 per 1M tokens input/output), LLM difficulty estimation is orders of magnitude  
232 cheaper than field testing. Rankings transfer across datasets; absolute calibration  
233 does not—models overestimate student ability on SmartPaper. Notably, item  
234 analysis prompts achieve lower MAE (0.12–0.19) than the teacher-role prompt  
235 (MAE=0.44), but the rank-order is preserved regardless of calibration bias.

236 For practitioners: use a capable model (Gemini, GPT-4o), prompt it to ana-  
237 lyze structural item features, and average 3+ replications. For standard MCQs,  
238 the teacher-role prompt suffices.

## 239 6 Discussion

240 *Why Directed Analysis Works.* The success of item analysis prompts suggests  
241 LLMs possess implicit pedagogical content knowledge—understanding of what  
242 makes content difficult—but this knowledge must be *elicited* through directed  
243 enumeration. Undirected chain-of-thought drifts toward solving the item; di-  
244 rected analysis (“list prerequisites, then estimate”) focuses that knowledge on  
245 the estimation task. This parallels findings in expertise research: experts often  
246 cannot articulate tacit knowledge until prompted with specific frameworks.

247 *Boundary Conditions.* The attenuation on MCQs reveals when LLM estimation  
248 fails: difficulty driven by *structural* features visible in the item text (prerequisites,  
249 cognitive load, error paths) is predictable; difficulty driven by *population* features  
250 absent from the text (prior exposure, distractor plausibility for specific miscon-  
251 ceptions) is not. Practical heuristic: use structured prompts for open-ended items  
252 spanning knowledge domains; use the teacher-role prompt for MCQs within a  
253 domain.

*Methodological Recommendations.* Our design-space exploration surfaced three lessons. (1) *Beware Simpson’s paradox in unbalanced designs.* We initially swept temperature only for promising prompts; a marginal ANOVA showed temperature as dominant ( $\eta^2 = 0.74$ ), but this was an artifact—low temperatures were confounded with high-performing prompts. When restricted to balanced conditions, temperature explained < 1% of variance. (2) *Use two-stage sequential DOE.* Screen all prompts cheaply (1 rep, single temp) to eliminate the bottom half, then run a balanced factorial on survivors. (3) *Validate on  $\geq 100$  items.* Our 20-item probe yielded  $\rho = 0.46$ ; the full 140-item set achieved  $\rho = 0.55$ . Bootstrap analysis showed only 2.5% of 20-item subsets would produce  $\rho \geq 0.50$ . Many published results use <50 items—treat such samples as exploratory only.

*Limitations.* Four limitations warrant mention. First, all 15 prompts were screened on a single model (Gemini 3 Flash); the model survey shows effects generalize but with diminished magnitude. Second, calibration does not transfer—models overestimate ability on SmartPaper but rankings are preserved, so population-specific adjustment would be needed for absolute predictions. Third, we do not compare to fine-tuned models; fine-tuned BERT/RoBERTa can outperform zero-shot LLMs when training data is available [14], but our approach targets the zero-shot case where items arrive without calibration data. Fourth, and most fundamentally: LLM estimation tells you how hard an item *looks*, not how hard it *is* for real students. The model has never been a confused 12-year-old parsing an English word problem. Field testing yields individual-level diagnostics, validity evidence, and distractor analysis that no proxy can replace.

*Future Work.* Three directions warrant investigation. First, batching multiple items per prompt showed preliminary improvements (+0.03–0.21  $\rho$ ) but requires systematic validation; if robust, this would further reduce per-item costs. Second, extended reasoning models (o1, DeepSeek-R1) may behave differently than the standard models tested here—our preliminary finding that thinking tokens *hurt* performance deserves replication on reasoning-optimized architectures. Third, the mechanism behind enumeration prompts remains unclear; ablation studies removing the “count the X” instruction could isolate the active ingredient and inform prompt design for other estimation tasks.

## 7 Conclusion

We mapped the design space for LLM item difficulty estimation across 15 prompts, 6 models, and 3 datasets ( $\sim 120$  configurations,  $\sim \$100$  in API calls). Item analysis prompts that enumerate structural features (prerequisites, cognitive load, error paths) achieve  $\rho = 0.65$ –0.69 on open-ended items, but this advantage attenuates on MCQs ( $\rho \approx 0.45$ –0.58), where the teacher-role prompt nearly matches. Prompt design is the primary lever; model size, temperature, and deliberation each contribute  $\leq 0.05 \rho$ . Rankings transfer across datasets but calibration does not—at  $\sim \$0.01$  per item, LLM estimates can triage items into difficulty bands,

295 but absolute predictions require population-specific adjustment. Methodologically,  
 296 correlations from <50 items are unreliable, and unbalanced hyperparameter  
 297 sweeps produce Simpson’s paradox artifacts; we recommend two-stage se-  
 298 quential DOE.

299 *Ethics Statement.* This study uses anonymized, aggregated student response  
 300 data. SmartPaper data consists of item-level statistics without individual student  
 301 identifiers. DBE-KT22 and BEA 2024 are publicly available research datasets.  
 302 No personally identifiable information was accessed or processed.

303 *Data and Code Availability.* Prompts and analysis code are available in the  
 304 project repository. SmartPaper data are available upon request from the origi-  
 305 nating organization. DBE-KT22 is publicly available [6]. BEA 2024 data are  
 306 available through the shared task organizers.

## 307 References

- 308 1. Acquaye, C., Huang, Y.T., Carpuat, M., Rudinger, R.: Take out your calculators:  
 309 Estimating the real difficulty of question items with LLM student simulations.  
 310 arXiv:2601.09953 (2025)
- 311 2. Benedetto, L., et al.: A survey on recent approaches to question difficulty estima-  
 312 tion from text. ACM Computing Surveys **56**(5), 1–37 (2023)
- 313 3. Brown, J.S., Burton, R.R.: Diagnostic models for procedural bugs in basic mathe-  
 314 matical skills. Cognitive Science **2**(2), 155–192 (1978)
- 315 4. Chi, M.T.H., Slotta, J.D., de Leeuw, N.: From things to processes: A theory of  
 316 conceptual change. Learning and Instruction **4**(1), 27–43 (1994)
- 317 5. de Ayala, R.J.: The Theory and Practice of Item Response Theory. Guilford Press  
 318 (2009)
- 319 6. Du Plessis, C., van Vuuren, J.: DBE-KT22: A Knowledge Tracing Dataset from  
 320 South African Secondary Schools. Zenodo (2022). <https://doi.org/10.5281/zenodo.7096666>
- 321 7. Embretson, S.E.: A cognitive design system approach to generating valid tests.  
 322 Psychological Methods **3**(3), 380–396 (1998)
- 323 8. Embretson, S.E., Reise, S.P.: Item Response Theory for Psychologists. Lawrence  
 324 Erlbaum (2000)
- 325 9. Feng, W., Tran, P., Sireci, S.G., Lan, A.: Reasoning and sampling-augmented MCQ  
 326 difficulty prediction via LLMs. arXiv:2503.08551 (2025)
- 327 10. Gorin, J.S.: Manipulating processing difficulty of reading comprehension questions.  
 328 J. Educational Measurement **42**(4), 351–373 (2005)
- 329 11. Hoge, R.D., Coladarci, T.: Teacher-based judgments of academic achievement: A  
 330 review. Review of Educational Research **59**(3), 297–313 (1989)
- 331 12. Koedinger, K.R., Corbett, A.T., Perfetti, C.: The Knowledge-Learning-Instruction  
 332 framework. Cognitive Science **36**(5), 757–798 (2012)
- 333 13. Li, M., Chen, H., Xiao, Y., et al.: Can LLMs estimate student struggles?  
 334 arXiv:2512.18880 (2025)
- 335 14. Li, M., Jiao, H., Zhou, T., et al.: Item difficulty modeling using fine-tuned small and  
 336 large language models. Educ. and Psych. Measurement **85**(6), 1065–1090 (2025)

- 338 15. Moore, S., Schmucker, R., Mitchell, T., Stamper, J.: Automated generation and  
 339 tagging of knowledge components from multiple-choice questions. In: L@S 2024,  
 340 pp. 122–133 (2024)
- 341 16. Peters, S., et al.: Text-based approaches to item difficulty modeling: A systematic  
 342 review. arXiv:2509.23486 (2025)
- 343 17. Razavi, P., Powers, S.J.: Estimating item difficulty using large language models  
 344 and tree-based ML. arXiv:2504.08804 (2025)
- 345 18. Rogoz, A., Ionescu, R.T.: UnibucLLM: Harnessing LLMs for automated prediction  
 346 of item difficulty and response time. In: BEA 2024, pp. 502–508 (2024)
- 347 19. Südkamp, A., Kaiser, J., Möller, J.: Accuracy of teachers' judgments: A meta-  
 348 analysis. *J. Educational Psychology* **104**(3), 743–763 (2012)
- 349 20. Sweller, J.: Cognitive load during problem solving. *Cognitive Science* **12**(2), 257–  
 350 285 (1988)
- 351 21. Thurstone, L.L.: A law of comparative judgment. *Psychological Review* **34**(4),  
 352 273–286 (1927)
- 353 22. Yaneva, V., et al.: Findings from the first shared task on automated prediction of  
 354 difficulty. In: BEA 2024, pp. 470–482 (2024)
- 355 23. Zotos, L., van Rijn, H., Nissim, M.: Are you doubtful? Exploring model uncertainty  
 356 for difficulty estimation. In: EDM 2025, pp. 77–89 (2025)

357 **A Example Prompts**

358 We show four prompts: baseline, best performer, best hybrid (item + popula-  
 359 tion), and worst performer. All share common elements: grade level, subject,  
 360 question text, rubric, and maximum score. They differ in what analysis they  
 361 request before the final estimate. Full prompts for all 16 conditions are available  
 362 in the project repository.

363 *Teacher (baseline).*

364 You are an experienced teacher in [subject] for Grade [N] students in  
 365 India.  
 366 For this open-ended question, estimate what proportion of students would  
 367 score full marks.  
 368 Question: [question\_text]  
 369 Rubric: [rubric]  
 370 Maximum score: [max\_score]  
 371 Think about:  
 372 - What specific errors or misunderstandings would cause students to  
 373 lose marks?  
 374 - How clearly does the question communicate what's expected?  
 375 - What prerequisite knowledge is needed?  
 376 - How likely are students at this grade level to have that knowledge?  
 377 Respond with ONLY a number between 0 and 1 representing the proportion  
 378 of students who would get full marks. For example: 0.45

379    *Prerequisite Chain (best performer,  $\rho = 0.69$ ).*

380    [Population context: economically weaker sections, Hindi-medium backgrounds]  
 381    For this question, identify the prerequisite knowledge and skills a  
 382    student needs. Count how many independent things must ALL go right for  
 383    a correct answer. Each prerequisite is a potential failure point.  
 384    Examples of prerequisites: reading comprehension, specific vocabulary,  
 385    a math operation, a concept definition, multi-step reasoning, writing  
 386    ability.  
 387    [Question, rubric, max score]  
 388    List the prerequisites, then estimate what proportion would get full  
 389    marks.  
 390    PREREQUISITES: [list them]  
 391    COUNT: [N]  
 392    Respond with ONLY a number between 0 and 1 on the last line.

393    *Buggy Rules (best hybrid,  $\rho = 0.66$ ).*

394    You are an expert in mathematical cognition and systematic student errors  
 395    (Brown & Burton, 1978).  
 396    [Population context]  
 397    For the following test item, analyze the cognitive demands:  
 398    [Grade, subject, question, rubric, max score]  
 399    Step 1: List the specific procedural steps a student must execute correctly.  
 400    Step 2: For each step, identify any known ‘buggy rules’ -- systematic  
 401    procedural errors students commonly make (e.g., subtracting smaller  
 402    from larger regardless of position, forgetting to carry).  
 403    Step 3: Consider the target student population.  
 404    Step 4: Taking into account ALL of the above analysis holistically,  
 405    estimate what proportion of students would produce the fully correct  
 406    answer.  
 407    Respond with ONLY a number between 0 and 1 on the last line.

408    *Synthetic Students (worst performer,  $\rho = 0.19$ ).* This two-stage approach first  
 409    generates 10 student personas, then simulates each attempting each item.

410    *Stage 1 — Persona generation:*

411    Generate 10 diverse student profiles for a Class [N] government school  
 412    in India. [Population context]  
 413    The class distribution should reflect a typical government school:  
 414    - 4 students: Below Basic (barely literate, struggle with basic concepts)  
 415    - 3 students: Basic (can read Hindi well, weak English)  
 416    - 2 students: Competent (understand most concepts, some errors)  
 417    - 1 student: Advanced (strong understanding, rarely makes mistakes)  
 418    STUDENT 1: [Name] | Level: [level] | [2-3 specific traits]  
 419    . . .

420    *Stage 2 — Student simulation (per item, per persona):*

421 You are role-playing as this student: [persona]  
422 [Grade, question, rubric, max score]  
423 Role-play this specific student attempting this question. Consider their  
424 reading ability, knowledge level, attention, and typical behaviors.  
425 Write their actual response as they would write it, then score it.  
426 RESPONSE: [what this student would actually write]  
427 SCORE: [0 to max\_score]