Classification of Student Struggle in Mathematics

Hannah Levin*, **Madhura Padwal***, **and Nchimunya Mwiinga***
Graduate School of Education & Computer Science Department
Stanford University
Correspondence: levinh@cs.stanford.edu

Abstract

Productive struggle is a critical component of mathematics education, requiring students to actively work through ideas rather than just making errors. However, identifying this struggle from text transcripts is challenging because students often mask confusion with epistemic hedging rather than direct statements. Zero-shot large language models exhibit a conservative bias, systematically under-detecting struggle in classroom discourse. We introduce a two-stage NLP pipeline comprising a lexical heuristic gate and an LLM subtype classifier. Our model achieves 90.0% binary accuracy and 84.0% 4-category accuracy. We demonstrate the pedagogical value of this tool by showing that struggle is uniquely concentrated during explicit mathematical reasoning, offering educators a scalable method for root-cause analysis.

1 Introduction

Productive struggle in mathematics involves students actively working through ideas they do not yet fully understand (Hiebert and Grouws, 2007). In classroom discourse, this struggle rarely manifests as explicit statements of inability; instead, students often rely on epistemic hedging, such as "I think" or "maybe", acting as a "shield" against being wrong (Rowland, 1995; Hyland, 1998). Identifying this struggle in real time is particularly challenging because its visibility depends heavily on instructional conditions: struggle surfaces most readily when teachers actively elicit student reasoning, build on student ideas, and create space for back-and-forth discussion. In more teacher-directed lessons, students may not be given the opportunity to verbalize their thinking, leaving the same cognitive roadblocks undetected. Furthermore, even when struggle does surface, its interpretation requires contextual judgment: a delayed explanation in an introductory lesson signals something very different from the same behavior after repeated modeling and scaffolded practice.

For educators managing large classrooms, conducting root-cause analysis to determine if a student is facing a conceptual misunderstanding or a linguistic barrier is incredibly time-consuming. Scalable Natural Language Processing (NLP) methods offer a promising path forward, enabling systematic analysis of classroom discourse patterns that would be impractical to review manually.

While the NCTE Classroom Transcript Analysis dataset (Demszky and Hill, 2022) provides a rich corpus of elementary math discourse in English, identifying genuine cognitive roadblocks from text alone is challenging. Off-the-shelf NLP models, including zero-shot large language models (LLMs), struggle to accurately parse these highly contextual linguistic markers. As our initial evaluations demonstrated, zero-shot LLMs exhibit a strong conservative bias, systematically under-detecting struggle by defaulting to a "no struggle" classification when faced with ambiguous student language. This bias is further exacerbated by transcription artifacts like "[inaudible]", which models frequently misinterpret as articulatory failure. To address this gap, we introduce a custom, two-stage NLP classification pipeline that identifies and categorizes student struggle into actionable subtypes: Implied Uncertainty, Impasse, and Cannot Explain. By coupling a high-recall, rule-based lexical heuristic gate (Stage 1) with an LLM subtype classifier powered by Gemini (Stage 2), our architecture directly counteracts the conservative bias of baseline models. Evaluated against a rigorously annotated validation dataset, our two-stage pipeline achieves 90.0% binary accuracy and completely eliminates false positives. Furthermore, exploratory application of the model to a holdout set of explicit mathematical reasoning reveals that 37.7% of these turns contain productive struggle. Crucially, our analysis demonstrates

*Equal contribution.

Proceedings of the 21st Workshop on Innovative Use of NLP for Building Educational Applications, pages 513–528

that this struggle is characterized by a stark linguistic shift: struggling utterances exhibit a significant drop in mathematical vocabulary density (averaging 0.55 math words) compared to fluent reasoning (0.87 math words). Ultimately, this model provides a scalable method for educators to conduct root-cause analysis, distinguishing conceptual gaps from linguistic barriers to better support student learning at scale.

2 Prior Literature

Our approach to detecting and classifying student difficulty sits at the intersection of mathematics education research and natural language processing for classroom discourse. We draw upon two primary bodies of work to motivate our construct definition and computational architecture.

In mathematics education, productive struggle is defined as the intellectual effort students expend to make sense of mathematical concepts that are not immediately apparent (Hiebert and Grouws, 2007; Warshauer, 2015). Crucially, this cognitive effort rarely manifests as explicit declarations of inability. Instead, students frequently operate within a "Zone of Conjectural Neutrality" (Rowland, 1995, 2000; Wagner et al., 2015), a cognitive space where they explore ideas they are not yet fully willing to assert. Within this zone, struggle is heavily characterized by epistemic hedging—the use of tentative language such as "maybe," "I think," or "kind of" (Hyland, 1998). Children use these linguistic hedges as a "shield" against the risk of being incorrect (Rowland, 1995).

Another signal of cognitive mathematical struggle is self-correction, where students stop mid-sentence to rephrase or restart. This kind of mid-turn repair often reflects active thinking rather than confusion (Schegloff et al., 1977), and in mathematics, it tends to appear during explanation and justification (Warshauer, 2015).

Pronoun use also matters. In mathematics classrooms, personal pronouns like "I," "we," and "you" serve complex functions related to ownership, authority, and generalization (Pimm, 1987; Rowland, 1999). When students shift from direct language like "I found" to vague phrasing like "you just do this" or "it works," this can reflect uncertainty about whether their approach is correct or generalizable (Rowland, 1999). Conversely, more agentive first-person language may emerge as understanding stabilizes.

Our operationalization of struggle builds directly on this framework, separating explicit breakdowns in articulation (Cannot Explain) or help-seeking (Impasse) from the far more common, but subtle, hedging behaviors (Implied Uncertainty). These categories are described in detail in the Validation Dataset section below, where we present our codebook and annotation framework.

While several educational dialogue datasets exist, none directly address the detection of productive struggle as a cognitive state in whole-class elementary mathematics discourse. The TalkMoves dataset (Suresh et al., 2022) annotates both teacher and student discursive moves across 567 K-12 mathematics transcripts, but its focus on accountable talk practices centers teacher behavior rather than student cognitive states. The Multi-turn Classroom Dialogue (MCD) dataset (Chen et al., 2024) targets mastery prediction from student-tutor dialogue, but is drawn from one-on-one tutoring sessions rather than whole-class instruction, and measures knowledge mastery rather than in-the-moment struggle. Unlike tutoring, where help-seeking is the explicit purpose of the interaction, whole-class discourse requires students to publicly commit to mathematical claims under social risk, which are precisely the conditions that produce the epistemic hedging and articulatory breakdown that our pipeline targets. The NCTE corpus (Demszky and Hill, 2022) is uniquely suited to our task: it captures authentic, whole-class elementary mathematics discourse during cognitively demanding instruction, providing the naturalistic linguistic environment necessary for studying how struggle surfaces in student talk.

Recent advancements in NLP have enabled the large-scale analysis of instructional dynamics. The release of corpora such as the NCTE dataset (Demszky and Hill, 2022) and the TalkMoves dataset (Suresh et al., 2022) has facilitated the automated extraction of teacher dialogue moves and student reasoning patterns. However, detecting nuanced affective or cognitive states, such as uncertainty or struggle, remains an open challenge.

Prior work has demonstrated that identifying student uncertainty in spoken educational dialogue is inherently difficult, often yielding moderate inter-rater reliability even among human experts (Forbes-Riley and Litman, 2011). While recent studies have shown that zero-shot LLMs can approximate human annotation quality on certain text-classification tasks (Gilardi et al., 2023), highly

context-dependent discourse phenomena remain a vulnerability. As our baseline evaluations indicate, when faced with the pragmatic ambiguity of epistemic hedging in transcripts stripped of prosodic cues, zero-shot LLMs exhibit a severe conservative bias. Our two-stage classification pipeline builds upon these findings, leveraging a high-recall lexical heuristic to bypass the zero-shot conservative bias while preserving the LLM's capacity for nuanced subtype classification.

Despite the rich theoretical grounding of productive struggle in mathematics education, the existing literature remains predominantly qualitative and teacher-centered (Young et al., 2023). On the computational side, NLP work on classroom discourse has focused primarily on teacher discourse moves and instructional quality (Suresh et al., 2022; Alic et al., 2022) rather than student cognitive states. Adjacent work on automated confusion detection relies on multimodal signals, prosodic, facial, and acoustic features, that are unavailable in transcript-only corpora (Ma et al., 2024). Text-based approaches to student challenge detection exist in collaborative learning contexts (Suraworachet et al., 2024) but have not been applied to the specific construct of productive struggle in whole-class elementary mathematics. This paper addresses that gap directly.

3 Data

3.1 Dataset

For this study, we used the NCTE Classroom Transcript Analysis dataset (Demszky and Hill, 2022), which is comprised of transcripts for fourth and fifth-grade mathematics instruction in four U.S. school districts serving largely historically marginalized student populations. The transcribed lessons cover core elementary mathematical concepts, including multi-digit multiplication, long division, decimals, and fractions, providing a rich linguistic environment for analyzing student reasoning. Lessons were audio-recorded in classrooms using a capture system that included a lapel microphone worn by the teacher and microphones positioned to pick up student talk. Recordings were fully transcribed by professional human transcribers and anonymized (Demszky and Hill, 2022). Student turns may include multiple speakers and do not track individual students over time. The transcripts capture spoken discourse during whole-class instruction only. They record if the speaker is a student or a teacher, in what order, and for how long, but do not represent written work, gestures, visual representations, or small-group interactions. As a result, the dataset reflects a partial view of mathematical activity, centered on publicly audible talk in teacher-led discussions.

The original purpose of the dataset was to examine instructional quality and teacher effectiveness, particularly how classroom practices relate to observation scores and student learning outcomes, rather than to study student struggle directly (Kane et al., 2015).

Because struggle is not explicitly labeled in our transcripts, we rely on these structural and lexical features as indirect markers: the timing of reasoning, the length and frequency of student turns, and patterns of tentative or evolving explanation. These features do not measure struggle directly, but prior research suggests they are reasonable signals of students working through uncertainty in whole-class discussion.

3.2 Task Definition: Operationalizing Struggle

Our primary NLP task was the automated detection and classification of student struggle within classroom discourse. We defined struggle as the intellectual effort expended to make sense of mathematical concepts or tasks that are not immediately apparent. To make this construct analytically tractable and pedagogically actionable, we operationalized it into four categories, extracted from the literature:

- **Implied Uncertainty:** Instances where students distance themselves from a mathematical proposition using hedging language (e.g., "maybe," "I think") to protect against potential errors, operating within a "Zone of Conjectural Neutrality" (Rowland, 1995, 2000).
- **Impasse (Call for Help):** Utterances containing clear evidence that a student is blocked from proceeding with a mathematical task, such as explicit statements of confusion or direct requests for help.
- **Cannot Explain:** Moments where a student has an answer but explicitly demonstrates an inability to articulate their reasoning or justify their thinking (e.g., "I just guessed").
- **No Struggle:** Fluent reasoning, procedural questions, or purely logistical discourse.

3.3 Sampling Strategy

To ensure annotators evaluated moments of genuine cognitive demand rather than procedural or off-task talk, we first filtered the corpus using edu_convokit to isolate turns containing explicit student mathematical reasoning. However, because struggle is rare even within reasoning turns, purely random sampling would still yield too few instances of minority categories for reliable annotation or model evaluation. We therefore used a hybrid stratified sampling approach across four buckets— one per codebook category—following Suresh et al. (2022), who address the same class imbalance problem when annotating rare discursive moves in K-12 math transcripts. Importantly, bucket assignment determined only which utterances were selected, not how they were labeled; annotators coded all items against the codebook without knowing their source bucket, preserving the independence of ground truth labels from the sampling mechanism. Because struggle markers are highly context-dependent, we extracted 11-turn contextual windows around each target utterance to preserve the surrounding teacher prompts and student responses needed for accurate coding. See Figure A in the appendix for examples of annotated target utterances among their contextual windows.

3.4 Annotation and Reliability

The development of our gold-standard validation set required an iterative codebook revision process to capture the nuanced realities of classroom dialogue. Our final gold-standard validation set consisted of 100 annotated student utterances. This dataset was developed in two distinct phases: an initial set of 50 utterances from our first sampling iteration, and a fresh set of 50 utterances generated using our finalized stratified sampling pipeline.

To ensure robust ground truth labels, all utterances were independently annotated by multiple raters. For the initial 50 utterances, three research team members (the authors) annotated all items independently. To provide critical practitioner validation, we also recruited two practicing elementary school mathematics teachers who split the initial set, annotating 25 items each. For the fresh set of 50 utterances using our finalized stratified sampling pipeline, the three research team members independently annotated all items under the revised codebook. A held-out test set of 50 items was drawn from the same stratified sampling pipeline and underwent identical independent annotation and consensus review by all three authors. The unannotated holdout (n=159) is an entirely separate corpus drawn from the remaining NCTE transcripts after all annotated sets were finalized and was used solely for exploratory analyses reported in Figures 5 and 6.

The development of this dataset required an iterative codebook revision process. Following the initial round of independent annotation, our team conducted consensus review sessions. This collaborative re-annotation yielded several critical clarifications: context-sensitive words such as "could" must be evaluated based on surrounding dialogue rather than automatically treated as hedges; Implied Uncertainty must be grounded in observable hedging language within the utterance itself; and Cannot Explain requires visible articulatory failure.

To address frequent misclassifications of confirmation-seeking questions, we introduced a "cover-up test" in our codebook: if removing the question mark leaves a substantive mathematical claim, the utterance is coded as Implied Uncertainty; otherwise, it is an Impasse. Two items could not be resolved and were excluded, leaving 98 usable validation items. After consensus, human inter-rater reliability on the fresh 50 items reached Krippendorff's $\alpha = 0.961$. See Figure 3 for an overview of the iterative pipeline development process.

4 Methods

4.1 Baseline Evaluations

Before developing our custom architecture, we evaluated four off-the-shelf approaches. A supervised classifier using sentence-level embeddings (all-MiniLM-L6-v2) performed at chance (binary $\kappa = 0.098$), confirming our dataset is too small for standard supervised methods. BART-large-MNLI collapsed to predicting a single category for 92% of utterances ($\kappa = 0.03$). Claude Sonnet 4 and Gemini 3 Pro both fell within the human agreement range ($\kappa = 0.52$ and 0.53) but exhibited a consistent conservative bias, missing genuine struggle 25-30% of the time by defaulting to No Struggle when student language was ambiguous. Critically, the two LLMs agreed with each other at $\kappa = 0.87$, making nearly identical errors— adding a second LLM provided no independent signal. This motivated a two-stage architecture: a lightweight rule-based heuristic handles binary detection cheaply and at

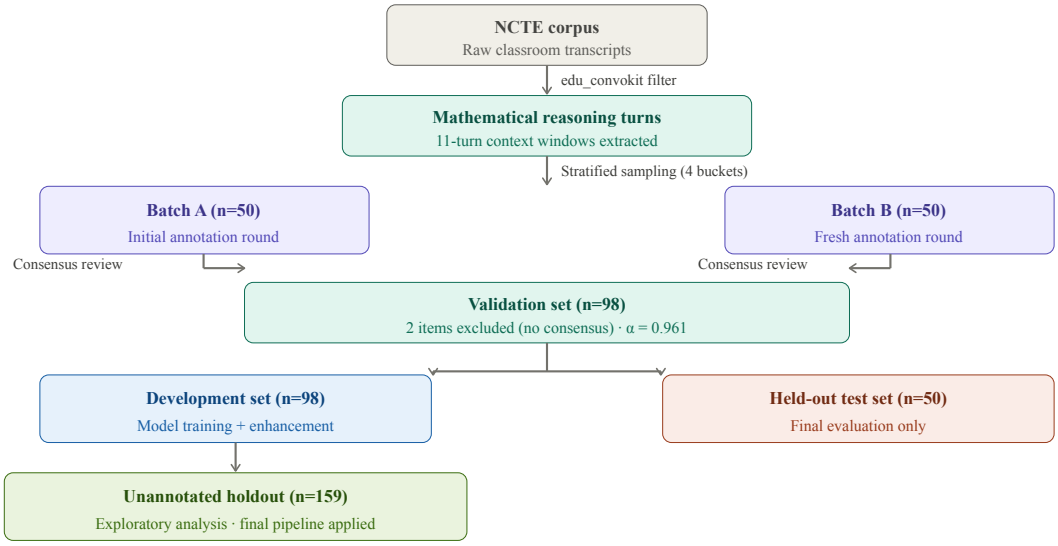

Figure 1: Pipeline for constructing and validating the annotated dataset

high recall, passing only flagged utterances to the LLM for subtype classification. This keeps the system computationally efficient while ensuring the LLM operates on an input where struggle is already the majority class, directly counteracting the conservative bias.

4.2 A Two-Stage Classification Pipeline

To address this conservative bias, we designed a custom, two-stage classification architecture that separates the binary detection task (Struggle vs. No Struggle) from the nuanced sub-categorization task.

Stage 1: High-Recall Lexical Heuristic Gate The first stage serves as a binary gate optimized strictly for high recall, ensuring that instances of struggle are not permanently lost early in the pipeline. This stage employs a seven-step, rule-based lexical heuristic that assigns weighted scores across three pattern dictionaries derived directly from our codebook. To overcome the limitations of simple keyword matching, we engineered several context-aware components into the heuristic:

- **Confirmation-Seeking Question Detector:** The system applies a "cover-up test" to utterances ending in a question mark. If removing the question mark leaves a substantive mathematical claim, the utterance is flagged as Implied Uncertainty rather than a procedural question.
- **Articulatory Failure Detection:** The heuristic distinguishes genuine self-correction from transcription artifacts by counting textual dashes, specifically subtracting those that co-occur with [inaudible] markers.
- **False Positive Filters:** The system suppresses context-dependent trigger words when used in non-struggle contexts, such as the word "help" used to describe a mathematical strategy (e.g., "I used the number line to help me") or "like" used in geometric comparisons.

Stage 2: LLM Subtype Classifier

Utterances flagged as struggle by Stage 1 are passed to Stage 2 for subtype classification (Implied Uncertainty, Impasse, Cannot Explain, or No Struggle). By pre-filtering the input, Stage 2 evaluates a class distribution where struggle is the majority, directly counteracting the LLM's baseline conservative bias.

This stage utilizes Gemini 3 Pro powered by a highly structured prompt. The prompt embeds all fourteen rules from our revised codebook and includes 50 few-shot examples targeting known boundary cases. It also includes an explicit anti-conservative-bias instruction, directing the model to prioritize finding the struggle over explaining it away.

Finally, our error analysis of early pipeline iterations revealed that LLMs systematically hallucinated articulatory failure when processing brack-

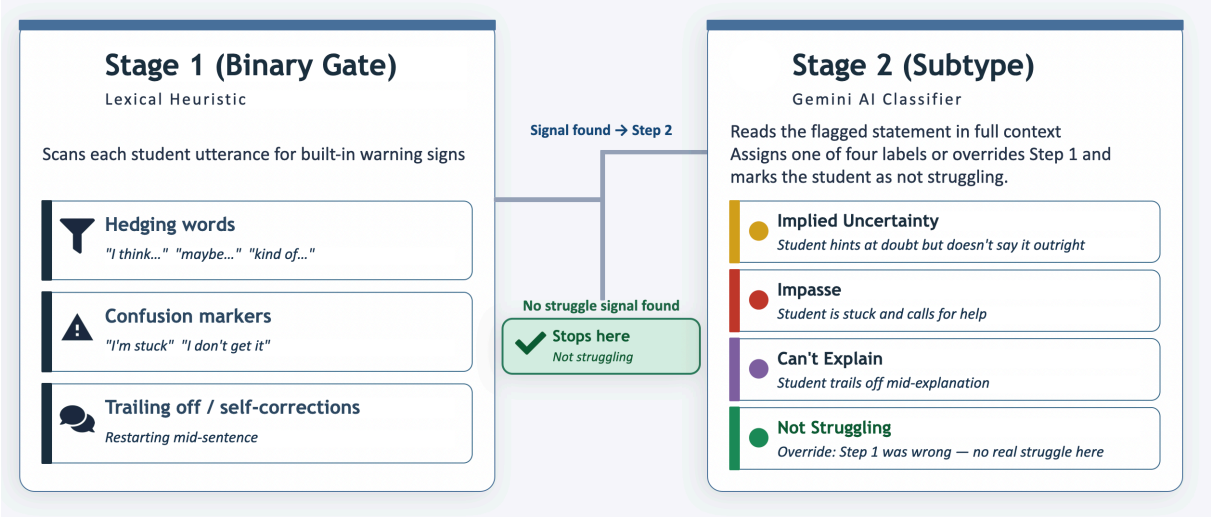

Figure 2: The two-stage struggle classification pipeline. A high-recall lexical heuristic (Stage 1) filters utterances for surface-level warning signs before passing flagged items to a Gemini LLM (Stage 2) for context-aware subtype classification and false-positive correction.

eted transcription markers (e.g., classifying an utterance with an [inaudible] tag as Cannot Explain). To eliminate this noise, all bracketed annotations were programmatically stripped from the context windows before being processed by Stage 2.

5 Results

We report final model performance on the held-out test set ($n = 50$), which was set aside at the start and never used during pipeline development. Table 1 shows how the pipeline improved through development on the validation set, while Table 2 details the per-category performance on the actual test set. The final two-stage pipeline reached an 84.0% four-category accuracy (Cohen's $\kappa = 0.716$) and a 90.0% binary accuracy in distinguishing struggle from non-struggle. In comparison, the baseline—a zero-shot Gemini model run as a standalone classifier—reached only 68.0% four-category accuracy and 76.0% binary accuracy on the same test set.

Crucially, the two-stage architecture successfully mitigated the LLM's baseline errors. By offloading the binary decision to the high-recall heuristic, the pipeline completely eliminated false positives; it never incorrectly flagged a non-struggling student as struggling. Total errors on the test set dropped by half, from 16 to 8.

As illustrated by the comparative F1 scores in Table 4, the most significant performance gains occurred in the minority struggle categories. The pipeline improved the F1 score for *Impasse* by 27 points (from 0.40 to 0.67), *Implied Uncertainty* by 24 points (from 0.36 to 0.60), and *Cannot Explain* by 20 points (from 0.58 to 0.78). The *No Struggle* category also saw a 12-point increase, reaching a final F1 of 0.92. Despite these substantial improvements, *Implied Uncertainty* remained the hardest category for the pipeline (F1 = 0.60). Through our error analysis, we found that these categories easily blur together in natural speech; the model plausibly often could not distinguish whether a student was simply expressing uncertainty or entirely lacking the words to articulate the concept.

Finally, we noted that categories such as *Impasse* and *Implied Uncertainty* appeared very rarely in the test set. Because a single misclassification could shift the F1 score by 10 to 30 points, we interpreted these specific metrics as general signs of pipeline improvement rather than absolute measures of the model's maximum performance.

6 Analysis

The evaluation demonstrated that the final model (84% accuracy, $\kappa = 0.716$) successfully mitigated the conservative bias observed across all off-the-shelf baselines. We found that by offloading binary detection to a high-recall lexical heuristic, Stage 2 received a class distribution where struggle was the majority, which directly counteracted the LLM's tendency to default to *No Struggle*. Furthermore, the error analysis revealed that the LLMs had conflated [inaudible] transcription tags with genuine articulatory failure, leading them to incorrectly la-

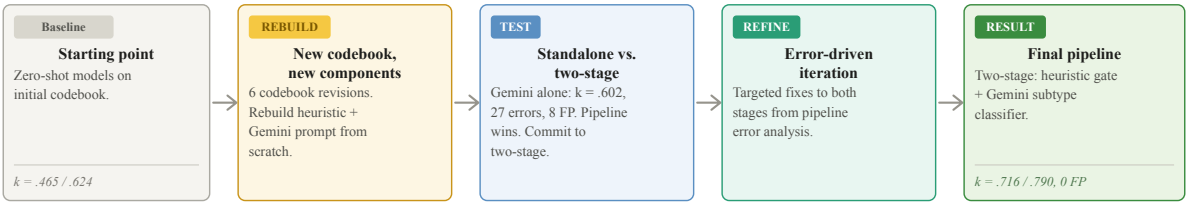

Figure 3: Pipeline development: from codebook revisions to architecture selection and final error-driven refinement

Model / benchmark	4-cat κ	Binary accuracy	Binary κ
Gemini standalone (baseline)	0.602	83.5%	0.657
Two-stage pipeline (Enh. 1)	0.766	95.9%	0.916
+ stripped brackets (Enh. 2)	0.827	95.0%	0.897
Human pre-discussion	0.534*	—	0.607*
Human post-discussion	0.961*	—	—

Table 1: Pipeline development on the validation set (n=98). Human benchmarks are Fleiss' κ (3 raters). Model kappa is Cohen's κ against consensus ground truth. These use different evaluation conditions and are not directly comparable.

* Fleiss' κ, 3 raters, fresh 50 items.

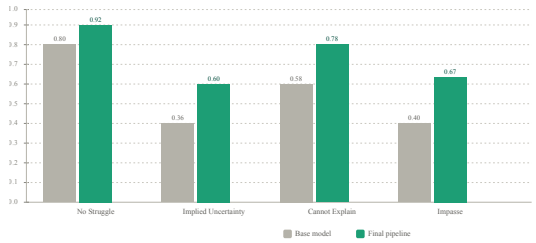

Figure 4: F1 score by category, base model vs. final pipeline (held-out test set, n=50).

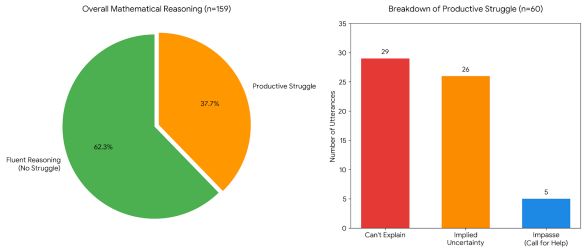

Figure 5: Distribution and categorical breakdown of student mathematical struggle

bel hedged utterances as *Cannot Explain*. It was observed that stripping bracketed annotations prior to Stage 2 resolved this issue and yielded the highest overall accuracy.

Regarding the eight remaining errors, the analysis indicated that seven persisted across both models, pointing to three irreducible limitations of text-only classification. It was noted that three of these errors involved the *Implied Uncertainty* and *Cannot Explain* boundary; both stages confused trailing dashes and restarts because distinguishing them required reading communicative intent, which surface patterns could not resolve. Finally, we observed that one *Cannot Explain* item failed Stage 1 entirely because the breakdown was semantic rather than lexical—the student produced fluent, unhedged speech, but their reasoning was mathematically incoherent. We concluded that further gains would require audio features, cleaner transcription, or the collapsing of *Implied Uncertainty* and *Cannot Explain* into a single category.

To evaluate pedagogical utility at scale, we applied the final model to 159 previously unseen mathematical reasoning turns filtered from the NCTE corpus using edu_convokit. As shown in Figure 5, the model flagged 37.7% (60 of 159) as containing productive struggle – consistent with the pipeline's 90% binary accuracy and reflecting a corpus already filtered for cognitively demanding turns. Cannot Explain was the most prevalent subtype (29 instances, 48%), followed by Implied Uncertainty (26 instances, 43%), while Impasse was the least frequent (5 instances, 8%). Students rarely voiced explicit calls for help; instead they either attempted an answer they could not fully justify, or hedged tentatively to protect against being wrong.

As seen in Figure 6, fluent reasoning turns av-

Category (n)	Base model Precision	Recall	F1	Final pipeline Precision	Recall	F1
No Struggle (31)	83%	77%	80%	93%	90%	92%
Implied Uncertainty (6)	40%	33%	36%	75%	50%	60%
Cannot Explain (10)	50%	70%	58%	69%	90%	78%
Impasse (3)	50%	33%	40%	67%	67%	67%
Macro avg	56%	53%	54%	76%	74%	74%

Table 2: Per-category performance on the held-out test set ($n = 50$)
Note: Small support for Implied Uncertainty (n=6) and Impasse (n=3) means a single item shifting changes F1 by 10-30 points. Read these figures as directional.

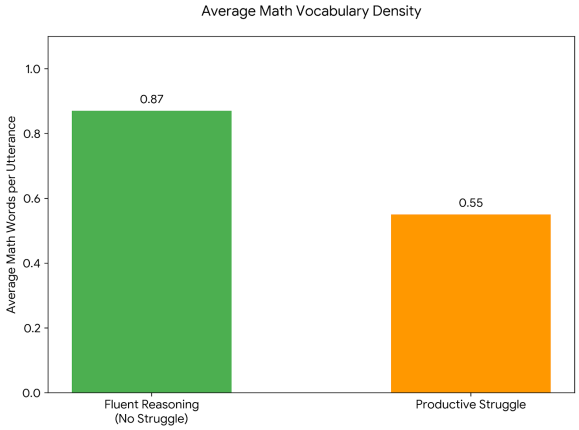

Figure 6: Average mathematical vocabulary density of student utterances using edu_convokit's get_math_density() function

eraged 0.87 domain-specific math words per utterance, while struggle-flagged turns averaged only 0.55 – a 37% drop. This aligns with the prevalence of Cannot Explain and Implied Uncertainty: when students hit a conceptual wall, they abandon formal math vocabulary and fall back on everyday language or extreme brevity. For educators, this proves the value of the tool for root-cause analysis: by isolating these moments, teachers can discern whether a student requires conceptual reteaching or simply needs linguistic scaffolding and sentence starters.

7 Conclusion

Identifying productive struggle in real-time classroom discourse is a complex challenge, primarily because students frequently mask their cognitive roadblocks with epistemic hedging. As our evaluations demonstrated, zero-shot large language models are ill-equipped to handle this pragmatic ambiguity, exhibiting a conservative bias that systematically under-detects genuine struggle. By implementing a custom, two-stage classification pipeline that pairs a high-recall lexical heuristic with a precision-tuned LLM, we successfully counteracted this bias. Our final architecture achieved an 84.0% 4-category classification accuracy and entirely eliminated false positives, ensuring the system never incorrectly flags a non-struggling student.

Beyond the technical architecture, our exploratory analysis confirmed the pedagogical value of this approach. The data revealed that explicit mathematical reasoning carries a substantial cognitive load, with productive struggle occurring in 37.7% of observed student participation. Furthermore, this struggle is characterized not just by hesitance, but by a measurable drop in domain-specific mathematical vocabulary, falling from an average of 0.87 to 0.55 math words per utterance. By successfully isolating these moments at scale, our pipeline provides a reliable mechanism for root-cause analysis, empowering educators to distinguish between conceptual misunderstandings and linguistic barriers.

8 Future Work

Because our model moves beyond binary detection and classifies the nature of the struggle (e.g., Implied Uncertainty, Impasse, Can't Explain), it offers highly targeted insights that can empower educators to make precise pedagogical adjustments.

Rather than functioning as an evaluative measure, this model can be used to power a post-lesson reflective dashboard for teachers. By processing classroom audio or transcripts, the dashboard could provide educators with a high-level summary of discourse trends. For example, it could highlight that "Your students showed high levels of Implied Uncertainty during today's fraction introduction." This preserves teacher agency; it does not tell the

educator how to teach, but rather cues them to areas requiring curriculum adaptation. If the data shows high rates of Impasse, the teacher may need to revisit foundational concepts. Conversely, if students are displaying high rates of Implied Uncertainty, the teacher might focus on building mathematical confidence and reinforcing a classroom culture where students feel safe asserting their knowledge.

At the same time, the struggle could be a language barrier and not conceptual. If the model identifies a cluster of Can't Explain moments, this signals to the teacher that students might actually grasp the underlying math but lack the academic vocabulary to articulate it. Teachers can use this specific feedback to shift their scaffolding strategy by providing linguistic support, such as sentence starters or targeted vocabulary review, rather than reteaching the mathematical concept itself.

A critical direction for future work is extending this pipeline beyond English-language classrooms. The current model is trained exclusively on English transcripts and relies on lexical markers of epistemic hedging that are English-specific. Different languages encode uncertainty and tentativeness through distinct grammatical and discourse mechanisms. Future work should investigate what text-based markers of productive struggle look like across typologically diverse languages, and whether a universal framework for struggle detection is achievable or whether language-specific codebooks are necessary.

Translating this backend classification model into a real-world, formative feedback dashboard will require dedicated co-design projects with teachers. Engaging directly with educators in this next phase of development will be critical to ensure that the resulting tool respects teacher agency, integrates smoothly into instructional workflows, and provides equitable, actionable insights.

9 Limitations

While our two-stage pipeline demonstrates strong performance, this approach has several methodological and technical limitations. First, our model relies entirely on text-based transcripts, which inherently strips away critical acoustic-prosodic features and non-verbal cues. Prior research has demonstrated that prosodic cues—such as rising intonation, trailing off, and physical hesitation—are significant predictors of student uncertainty in spoken educational dialogue. Because our pipeline evaluates text alone, it systematically misses instances of implicit struggle conveyed through tone of voice or body language. Second, our evaluation is constrained by a relatively small and homogeneous ground-truth validation set drawn exclusively from U.S. fourth and fifth-grade mathematics classrooms. Consequently, performance metrics for rare categories, such as Cannot Explain (which comprised only 9% of our validation dataset), suffer from base rate instability. Expanding the dataset across different grade levels and instructional contexts is necessary to stabilize the evaluation of these sub-types.

Third, our reliance on a majority-vote human consensus for ground-truth labels introduces an artificial ceiling on model performance. In cases of genuine boundary ambiguity, the model is penalized for agreeing with a minority human rater, even when that rater's interpretation may be pedagogically valid.

Fourth, our model exhibits an asymmetric error profile by design. In optimizing the pipeline to eliminate false positives, the model retains a slight conservative bias, producing a small number of false negatives. In a real-world deployment, this means the system will occasionally miss a struggling student before it ever incorrectly flags a fluent one.

Finally, a core limitation of this evaluation is the small held-out test set (n=50) and the very low support for minority categories — Impasse appears only 3 times and Implied Uncertainty only 6 times in the test set. At this scale, a single misclassification shifts F1 by 10–30 points, making per-class metrics statistically unstable and difficult to interpret as absolute measures of model performance. The current results are therefore better taken as preliminary evidence of a promising pipeline architecture rather than proof of a mature system's performance. More broadly, the small size of our annotated dataset was drawn exclusively from U.S. fourth and fifth-grade mathematics classrooms, thus limiting the generalizability of our findings. We cannot confidently claim that the pipeline would perform equivalently across different grade levels, subject areas, or instructional contexts. Expanding the annotated dataset through targeted oversampling of rare categories, and validating performance across more diverse classroom settings, are necessary preconditions for establishing the robustness and generalizability of this approach.

10 Ethical Considerations

The deployment of this classification pipeline carries significant ethical considerations, particularly regarding linguistic fairness and responsible educational use.

A primary ethical concern is algorithmic bias regarding English as a Second Language (ESL) learners. A key limitation of the underlying NCTE dataset is that only English speech is transcribed. However, on average there are approximately 260 instances of foreign language redactions (e.g., "[spanish]" or "[speaking in foreign language]") of unknown length per lesson. This is critical to note when evaluating student struggle, as a student's difficulty may not be conceptual but rather a language barrier. We initially assumed that keeping bracketed annotations, although inconsistent, was important for preserving the context of student learning. However, as demonstrated in our error analysis, retaining tags such as [inaudible] introduced significant noise, causing the LLM to systematically misclassify transcription gaps as articulatory failure, which ultimately required us to strip them from the input.

Furthermore, because our pipeline relies heavily on structural markers of hedging and questioning, the model risks misinterpreting the natural discourse patterns of ESL students. These students may frequently utilize hedges or questioning intonations as they master English syntax, rather than as an indication of mathematical confusion. Uncritical deployment could result in educators inappropriately over-scaffolding students who are mathematically proficient but linguistically developing. Future work must focus on evaluating this pipeline across more linguistically diverse classroom environments to address potential performance disparities and biases among ESL students.

Additionally, the system's reliance on observable verbal signals introduces the problem of the "quiet struggler." Students who disengage, give minimal responses, or remain entirely silent when struggling will not be captured by the system, potentially masking the needs of the most vulnerable learners.

Given these constraints, this model must never be utilized for high-stakes decision-making, including student grading, tracking, or teacher evaluation. It is designed strictly as a low-stakes, formative reflection tool to support, rather than replace, the holistic professional judgment of human educators. Finally, because the model classifies isolated utterances rather than individuals, any deployment must actively prevent the aggregation of these classifications into student-level profiles to avoid the ethical risk of label reification.

Acknowledgments

We thank Dorottya Demszky and Mei Tan for their guidance, methodological feedback, and support throughout the development of this research. We extend our deepest gratitude to Jennifer Gray and Erin Hanau, whose expertise as practicing elementary mathematics educators was invaluable. Their dedication to independently annotating our validation data and participating in our codebook review sessions provided critical practitioner validation for our construct definitions.

References

Sterling Alic, Dorottya Demszky, Zid Mancenido, Jing Liu, Heather Hill, and Dan Jurafsky. 2022. Computationally identifying funneling and focusing questions in classroom discourse. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 224–233, Seattle, Washington. Association for Computational Linguistics.

Jiahao Chen, Zitao Liu, Mingliang Hou, Xiangyu Zhao, and Weiqi Luo. 2024. Multi-turn classroom dialogue dataset: Assessing student performance from one-on-one conversations. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 5333–5337, New York, NY, USA. Association for Computing Machinery.

Dorottya Demszky and Heather C Hill. 2022. The ncte transcripts: A dataset of elementary math classroom transcripts. *arXiv preprint arXiv:2211.11772*.

Kate Forbes-Riley and Diane Litman. 2011. Benefits and challenges of real-time uncertainty detection and adaptation in a spoken dialogue computer tutor. *Speech Communication*, 53(9-10):1115–1136.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd-workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120.

James Hiebert and Douglas A Grouws. 2007. The effects of classroom mathematics teaching on students' learning. In Frank K Lester, editor, *Second handbook of research on mathematics teaching and learning*, pages 371–404. Information Age Publishing.

Ken Hyland. 1998. *Hedging in scientific research articles*. John Benjamins Publishing.

Thomas J. Kane, Daniel F. McCaffrey, Trey Miller, and Douglas O. Staiger. 2015. Have we identified effective teachers? Validating measures of effective teaching using random assignment. Technical report, RAND Corporation. Prepared for the Bill & Melinda Gates Foundation.

Yingbo Ma, Yukyeong Song, Mehmet Celepkolu, Kristy Elizabeth Boyer, Eric Wiebe, Collin F Lynch, and Maya Israel. 2024. Automatically detecting confusion and conflict during collaborative learning using linguistic, prosodic, and facial cues. *arXiv preprint arXiv:2401.15201*.

David Pimm. 1987. *Speaking Mathematically: Communication in Mathematics Classrooms*. Routledge, London.

Tim Rowland. 1995. Hedges in mathematics talk: Linguistic pointers to uncertainty. *Educational Studies in Mathematics*, 29(4):327–353.

Tim Rowland. 1999. Pronouns in mathematics talk: Power, vagueness and generalisation. *For the Learning of Mathematics*, 19(2):19–26.

Tim Rowland. 2000. *The Pragmatics of Mathematics Education: Vagueness in Mathematical Discourse*. Falmer Press.

Emanuel A. Schegloff, Gail Jefferson, and Harvey Sacks. 1977. The preference for self-correction in the organization of repair in conversation. *Language*, 53(2):361–382.

Wannapon Suraworachet, Jennifer Seon, and Mutlu Cukurova. 2024. Predicting challenge moments from students' discourse: A comparison of gpt-4 to two traditional natural language processing approaches. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, LAK '24, page 473–485, New York, NY, USA. Association for Computing Machinery.

Abhijit Suresh, Jennifer Jacobs, Charis Harty, Margaret Perkoff, James H. Martin, and Tamara Sumner. 2022. The TalkMoves dataset: K-12 mathematics lesson transcripts annotated for teacher and student discursive moves. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4654–4662, Marseille, France. European Language Resources Association.

David Wagner, Joseph Dicks, and Paula Kristmanson. 2015. Students' language repertoires for prediction. In *CERME 9: Ninth Congress of the European Society for Research in Mathematics Education*, pages 1517–1523.

Hiroko K Warshauer. 2015. Productive struggle in middle school mathematics classrooms. *Journal of Mathematics Teacher Education*, 18(4):375–400.

Jamaal Young, Danielle Bevan, and Miriam Sanders. 2023. How productive is the productive struggle? Lessons learned from a scoping review. *International Journal of Education in Mathematics, Science and Technology*, 12(2):470–495.

A Qualitative Coding Examples

Representative examples from each category in the codebook. Boldface marks the target utterance being classified. Contexts are lightly truncated for space.

Cannot Explain

Target utterance
Student: "I guessed on these two parts."

Context
teacher: [inaudible] so this is what she's done. She drew the model of fifteens, marked off six-fifteenths to prove that—
student: Make it.
student: Ms. M?
student: I guessed on these two parts.
student: Multiplication.
teacher: How much did you eat of this pizza, if he ate this?
student: It's small.
teacher: You make it. So, construct means to make or to build . . .

Implied Uncertainty

Target utterance
Student: "Still too short. I think maybe 35."

Context
teacher: I think you're right. What about 4 times 30?
student: This says, "I don't think this is . . . "
teacher: Awesome. Good. What you got? [Students working on problem].
student: Still too short. I think maybe 35.
teacher: Uh-oh, I'll help you. Let me see . . .
teacher: 35. Let's try 4 times 30 first. I'm gonna use repeated addition to solve that. 30, 60, 90.

Impasse

Target utterance
Student: "Wait – I'm stuck on three hundredths."

Context
teacher: Okay. Speak loud so I can understand you.
teacher: Like, your ice is about to break . . .
student: Yes.
student: Wait – I'm stuck on three hundredths.
student: I put them in order by doing the first number just like the thirty-something. I just did the tens first.
student: [inaudible]. Regroup the ones. Oh, and then I had to [inaudible].

No Struggle

Target utterance
Student: "Well, I knew that when you added five sixths with seven sixths that's an improper fraction."

Context
teacher: Perfect. I knew that's what you did. Can we reduce seven twelfths? . . .
teacher: What kind of math did you do?
student: Well, I knew that when you added five sixths with seven sixths that's an improper fraction.
teacher: Good, improper fraction. So then what?
student: So then I had a whole number and I got six divided by seven was one and then I added the one.
teacher: Do I do six divided by seven?
student: No, wait. Seven divided by six.

B Pipeline Implementation Details

B.1 Stage 1: Lexical Heuristic Classifier

Stage 1 applies a weighted regex scoring system over normalized utterance text. Normalization converts smart quotes, em/en dashes, and ellipsis characters to ASCII equivalents before pattern matching. Each utterance is scored independently across three categories; scores do not accumulate across categories.

Pre-Score Filtering. Before any scoring, fourteen patterns unconditionally return No Struggle. Seven cover non-struggle uses of trigger words: "help" used as a mathematical strategy (e.g., "I used the number line to help me"); offers of peer assistance; "be there to help" phrasing; "can I say" openers; "like" or "kind of" in a geometric comparison with no other struggle signals present; "I think" introducing reported speech; and "I don't know why I said that" as metacognitive self-correction. Seven cover utterances that are not mathematically uncertain regardless of wording: offers to help others; logistical questions ("Can we write this down?", "Can we go?"); self-deferral openers ("Let me think", "I don't want to"); forward-readiness statements ("We'll be ready"); and confident "I know" statements not followed by a reversal.

Scoring. Stage 1 computes four scores: an Im (Impasse) score, a CE (Cannot Explain) score, a IU (Implied Uncertainty) score, and a disfluency score. The three category scores are built from: (1) weighted regex pattern matches, where high-confidence patterns contribute $+3$, medium $+2$, and low $+1$; (2) CSQ detection, which adds $+2$ to Im or IU depending on the cover-up test result; and (3) trailing-off boosts, which add directly to CE or Im when dash and "because" patterns co-occur. The fourth score, the disfluency score, is computed independently by twelve surface breakdown signals and is only consulted in routing when all three category scores are zero.

A single utterance can score points in multiple categories simultaneously; the priority order Im $>$ CE $>$ IU reflects signal specificity: explicit help-seeking is the most diagnostic, visible breakdown more specific than hedging, and hedging the most ambiguous but cheapest to route. An utterance is routed to Stage 2 as Struggle, with the subtype below, if the first matching condition holds:

1. **Impasse** if Im score ≥ 2.
2. **Can't Explain** if CE score ≥ 2.
3. **Impasse** if Im score > 0 and Im score $\geq$ IU score (Im wins a tie at score 1).
4. **Implied Uncertainty** if IU score ≥ 1 (deliberately low: a single hedge such as "I think" is sufficient).
5. **Can't Explain** if CE score > 0 (any disfluency-driven CE signal is sufficient).
6. **Implied Uncertainty** (provisional) if any disfluency signal fires and all category scores are zero; subtype is assigned per the disfluency mapping described below.

Utterances matching none of these conditions are labeled No Struggle without LLM consultation.

Confirmation-seeking question (CSQ) detection. A question is identified as a CSQ via the *cover-up test*: the question mark is mentally removed; if a mathematical claim remains (e.g., *"So it's minus five?"* $\rightarrow$ *"So it's minus five"*), the utterance is classified IU. If nothing substantive remains (e.g., *"What do I do?"*), it is classified Im. Procedural questions (*"Should we write this down?"*) and hypotheticals (*"What if it was a different number?"*) are excluded.

Backup Disfluency Score. This fourth score is only computed if the previous three category scores were all zero.

1. `[inaudible]` presence, weighted $+1$ per occurrence capped at 2;
2. "like" as hedge filler, with a filter suppressing comparison and observation uses;
3. dash-based restarts, weighted $+2$ for two or more real dashes and $+1$ for one;
4. ellipsis or trailing off;
5. cut-off utterances ending in a dash;
6. filler markers (*um*, *uh*, *hmm*);
7. repeated "because" indicating circular or restarted reasoning;
8. "though" at utterance end signaling unresolved doubt;

9. brevity of 15 words or fewer as a catch-all signal for minimal responses to *why*/*how* prompts;

10. self-correction markers (*wait*, *actually*, *I mean*, *never mind*);

11. tentative proposals (*what about*, *what if*, *how about*);

12. "I just know" or "I just knew."

If all three category scores are zero but the disfluency total is ≥ 1, the utterance is still routed to Stage 2, with a provisional subtype assigned based on which signal fired: cut-off, restart, or repeated-*because* signals map to *Can't Explain*; *like*-hedge or *though* signals map to *Implied Uncertainty*; tentative proposals map to *Impasse*; all others map to *Implied Uncertainty* as a default.

Tense coverage. All pattern lists include both present- and past-tense variants to capture retrospective struggle narration (e.g., "I got stuck," "I couldn't figure it out," "I didn't know why I did that"). This design decision was motivated by the NCTE transcripts frequently capturing students narrating prior cognitive states during whole-class discussion, where a student may describe a moment of difficulty that occurred during independent work rather than expressing real-time confusion. Without past-tense variants, these retrospective struggle markers would be systematically missed, introducing a structural false negative bias against students who reflect on their process rather than voicing struggle in the moment.

B.2 Stage 2: LLM Classifier

Stage 2 uses gemini-3.1-pro-preview with temperature = 0.0 for deterministic output. Utterances are batched in groups of 3 target utterances, along with their surrounding context. The model is prompted to return a raw JSON array of `{id, label}` objects.

If Gemini refuses an entire batch (safety or recitation block), each utterance in that batch is resent individually. If Gemini returns fewer classifications than utterances sent, the missing items are also resent individually. In both cases a 1-second delay separates individual retries.

B.3 Full System Prompt

An abridged version of the system prompt is reproduced below. Category definitions, classification rules, and few-shot examples are provided verbatim as supplied to the model.

Parameter	Value
Model	gemini-3.1-pro-preview
Temperature	0.0
Max output tokens	4096 (batch), 1024 (retry)
Batch size	3 target utterances
Retry strategy	Item-by-item on block or missing label
Retry delay	1s per item, 3s between batches
Fallback	Stage 1 subtype if retry fails

Table 3: Stage 2 reproducibility parameters.

> You are an expert annotator classifying student struggle in elementary math classroom transcripts. You will receive student utterances that have already been flagged as likely containing struggle. Classify each one into exactly one of four categories.
>
> **Implied Uncertainty (IU):** The student has something to offer but withholds full commitment. They express doubt, hedge, or seek confirmation. Key property: the student CAN continue but is uncertain. Past-tense hedges count: "I thought," "I figured," "I assumed," "I was thinking" all signal retrospective uncertainty.
>
> **Impasse / Call for Help (Im):** The student cannot move forward. They have no answer, are stuck, confused, or need teacher intervention to proceed. Key property: the student CANNOT continue without help. Past-tense impasse counts: "I got stuck," "I was confused," "I didn't know what to do," "I couldn't figure it out."
>
> **Can't Explain (CE):** The student has an answer or has taken action but fails to articulate WHY. The explanation visibly breaks down. Key property: the student is TRYING to explain but FAILING to do so coherently. Past-tense CE counts: "I couldn't explain it," "I didn't know why I did that," "I had guessed."
>
> **No Struggle (NS):** The student does not demonstrate struggle. They answer clearly, or the utterance is about non-mathematical content, or surface-level hedge words are explained by context (filler, mirrored language, reporting).
>
> **Critical Rules:** Rules R1–R14; see Table 4 and Table 5 for full definitions and examples.
>
> **Output format:** Respond with ONLY a JSON array. Each element must have `id` and `label` fields. Use these exact label strings: `"Implied Uncertainty"`, `"Impasse"`, `"Can't Explain"`, `"No Struggle"`. Do not include any other text, explanation, or markdown formatting.

Rule	Name	Description
R1	Bias toward Struggle	When genuinely ambiguous, lean toward Struggle. Do not force Struggle on clearly coherent reasoning that merely contains dashes or `[inaudible]` markers.
R2	Cover-Up Test	Remove the question mark: if a mathematical claim remains, classify IU; if nothing substantive remains, classify Im. Do not classify confirmation-seeking questions as NS.
R3	[Inaudible] not CE	`[inaudible]` markers are transcription artifacts, not articulatory failure. Do not classify CE solely because of them. Audible hedges still trigger IU.
R4	CE requires visible breakdown	CE requires at least one of: repeated self-corrections, incomplete "because" statement, trailing off, restarting and collapsing, circular explanation, or explicitly stating they guessed or can't explain.
R5	Filler vs. hedge	"Like" and "kind of" in comparisons are filler, not hedges. "Like" at the start of a turn or mid-sentence IS a hedge signal.
R6	Reporting vs. reasoning	"I think he said X" is reporting → NS. "I think she thinks 6 plus 5 equals 11" is reasoning about math → IU.
R7	Non-mathematical content	Non-mathematical utterances are always NS.
R8	Self-correction phrase	"I don't know why I said that" is self-correction, not struggle.
R9	Guess: CE vs. IU	"I guessed" (past, action taken) + breakdown = CE. "I guess it's X" (present hedge) = IU.
R10	Trailing reasoning	A "Because…" statement that becomes incoherent, restarts, or ends abruptly is CE, not NS.
R11	Past tense counts	Past-tense struggle markers ("I got stuck," "I couldn't explain") count the same as present-tense ones.
R12	Use context	Read surrounding turns. Fluent-sounding utterances may reveal struggle when teacher re-explains; apparent NS may follow heavy scaffolding.
R13	Tautological explanations	A "because" clause that merely restates the problem without logical justification is CE, even if delivered confidently.
R14	Resolving self-corrections	A dash followed by a completed, corrected thought is normal self-repair → NS. A dash followed by stalling, restarting, or collapse → CE.

Table 4: Classification rules (R1–R14) embedded in the Stage 2 system prompt.

B.4 Few-Shot Examples

Table 5 presents the 50 few-shot examples supplied in the system prompt, organized by label. Examples were drawn from the annotation codebook developed for this project and selected to cover the most commonly confused category boundaries: IU vs. NS (hedge vs. comparison filler), CE vs. NS (self-correction that resolves vs. breakdown), and Im vs. IU (cover-up test cases).

Label	Utterance	Key signal
IU	"Maybe to find all the prime factors?"	hedge (*maybe*)
IU	"Oh, so it's minus five because it's less than?"	CSQ, claim remains
IU	"If the 2 is a whole, it would be before the decimal point?"	CSQ, claim remains
IU	"I tried to do a number line."	*tried to*
IU	"And we used the towers?"	past-tense CSQ
IU	"I thought it was going to be different."	past-tense hedge
IU	"He could multiply again 58 times 3."	modal tentativeness
IU	"Like, you have nine seconds and it's . . . "	*like* as filler
IU	"It's like 10 between 20 is 15."	*like* as hedge
IU	"Like, um, one-third, because [inaudible]. . . "	*like, um*
IU	"For this one, I did like [inaudible] order of operation . . . I thought the X equals one"	*like* filler + *I thought*
IU	"She would be the best, because [inaudible] would get like [inaudible]."	vague + *like* hedge
IU	"She mostly didn't—she actually kinda did get jipped . . . "	*kinda*, hedged reasoning
IU	"Except this one. You can divide by one. Right?"	*Right?* confirmation
IU	"A possibility . . . I think it's less [inaudible]."	*I think* + audible hedge
Im	"Okay. So what do I do up here?"	direct help request
Im	"Could we do one sixth and five sixths?"	open question, no answer
Im	"Could there be a thing that could weigh five grams?"	open question
Im	"Yes. Like, it's kind of confusing."	explicit confusion
Im	"Will the teacher be able to help us?"	help-seeking
Im	"Mr. F, how does that add up to eight sides. . . ?"	teacher-directed confusion
Im	"But what about the bottom part? So would the bottom be seven?"	*what about* + unresolved
Im	"What about four and three—four times three."	restart, tentative
Im	"No matter what you multiply it . . . will you still get the same answer?"	fundamental clarification
Im	"Or you could have 3 times—3 times—3 times [inaudible]."	repeated restarts
Im	"I don't get it, 'cause 70 percent is—"	explicit + trailing off
Im	"But the right one still look bigger though."	unresolved doubt
Im	"This was hard because it's impossible."	*hard*, *impossible*
Im	"Hold on, but what's—"	mid-sentence stop
Im	"That if all the edges, the number of edges are the number of vertices."	confused conflation
Im	"I got stuck on the part where you had to divide."	past-tense impasse
CE	"I guessed eight—I guessed six, because . . . "	self-correction + weak logic
CE	"Because it's one and four tenths, because I think—I can't explain it."	explicit give-up
CE	"Because 2,000—wait. Because 3,237 marched [inaudible]"	restart + collapse
CE	"Because 800—I mean 812—I—because I know. . . "	multiple corrections
CE	"Mm-hmm, because if you look at them—right now . . . "	trails off
CE	"Because . . . because four times seven is 28, and we—"	repeated *because* + cut-off
CE	"Because I would need the half and half to build the other half."	circular
CE	"Because half of 10—'cause half of 10 [inaudible]."	restart, no advance
CE	"That can be 20 and that can be 30 because that's not a half."	vague assertion
CE	"Because the leftovers, you can add them [inaudible]."	incomplete logic
CE	"It's negative because—it's negative because . . . "	repeated restart
CE	"Because you only put the square . . . that's the area."	circular
CE	"I multiplied 6 times 2 . . . but I don't know the reason why I did 6. . . "	explicit CE statement
CE	"I couldn't explain why I chose that answer."	past-tense CE
NS	"Four-eighths is kind of like one-half."	comparison filler
NS	"I think he said seven and then 10."	reporting speech
NS	"Because if you would take a rectangle and cut it in the middle . . . "	complete reasoning
NS	"Halves can work 'cause I know that twelve is half of twenty-four . . . "	clear justification
NS	"I just knew it was remainder 2 because I knew that 3 times 5 is 15. . . "	confident + full logic

Table 5: All 45 few-shot examples provided in the Stage 2 system prompt, with the primary signal that motivated each label assignment.