IDEAlign: Comparing Ideas of Large Language Models to Domain Experts

Hyunji Nam and Lucia Langlois and James Malamut
Mei Tan and Dorottya Demszky
Stanford University
{hjnam, ddemszky}@stanford.edu

Abstract

Large language models (LLMs) are increasingly used to produce open-ended, interpretive annotations, yet there is no validated, scalable measure of *idea-level similarity* to expert annotations. We (i) introduce the content evaluation of LLM annotations as a core, understudied task, (ii) propose IDEAlign for capturing expert similarity judgments via *pick-the-odd-one-out* tasks, and (iii) benchmark various similarity methods (text embeddings, topic models, and LLM-as-a-judge) against these human ratings. Applying this approach to two real-world educational datasets (e.g., interpreting math reasoning and feedback generation), we find that most metrics fail to capture the nuanced dimensions of similarity meaningful to experts. LLM-as-a-judge performs best (11–18% improvement over other methods) but still falls short of expert alignment, making it useful as a triage tool rather than a substitute for human review. Our work demonstrates the difficulty of evaluating open-ended LLM annotations at scale, and positions IDEAlign as a reusable protocol for benchmarking on this task to help guide responsible deployment of LLMs.[1]

1 Introduction

Comparing model predictions to human labels has long been central to model evaluation. When outputs are categorical, assessment is relatively straightforward (Angelopoulos et al., 2023; Zhou et al., 2024; Strachan et al., 2024; Bojić et al., 2025; Schroeder and Wood-Doughty, 2025). By contrast, large language models (LLMs) are increasingly used to generate unstructured, open-ended annotations involving reasoning and interpretation grounded in domain-specific knowledge (Tan et al., 2024b; Yu et al., 2023). In socially impactful domains, like education, LLM-based tools like KhanMigo and Brisk observe student thinking and writing to provide insights to students and teachers (Wen et al., 2024; Graesser et al., 2004; Khan, 2023; Google, 2024; Wang et al., 2024b). Because these interpretive model outputs can shape instruction and learning, expert-grounded evaluation of LLMs is essential. However, doing so is non-trivial: the tasks require domain and context knowledge (e.g., standards and learning objectives), and rarely admit a single "correct" response.

To evaluate and improve LLMs on such tasks, we need to assess their alignment with experts in terms of ideas: whether they express the same underlying interpretations or recommended actions, rather than imitating surface features (tone, length, phrasing). Measuring idea-level similarity is difficult because (i) expert annotations legitimately vary across perspectives and communities (Santurkar et al., 2023); and (ii) absolute scales for subjective constructs are difficult to elicit reliably (Annett, 2002). These challenges motivate a relative judgment approach for eliciting human judgments, which can serve as benchmarks.

Existing evaluations of interpretive LLM annotations remain limited: many focus on predefined code sets (Wadhwa et al., 2024; Larionov et al., 2019) or prediction accuracy over reasoning quality (Yu et al., 2023). Prior work on general long-form texts have targeted measuring and improving quality of text generation along dimensions of self-consistency (Huang et al., 2023) and human values, such as creativity and truthfulness (Ismayilzada et al., 2025; Spangher et al., 2024; Tan et al., 2024a; Lin et al., 2021), but these approaches are not suited for assessing conceptual alignment between LLM and expert annotations. To our knowledge, alignment of ideas on open-ended, interpretive annotations has not been systematically studied.

Automated text comparison metrics have complementary limitations: lexical overlap (e.g., BLEU (Papineni et al., 2002) ignores paraphrase; embedding-based (Reimers and Gurevych, 2019;

[1] https://github.com/EduNLP/IDEAlign

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 3908–3925

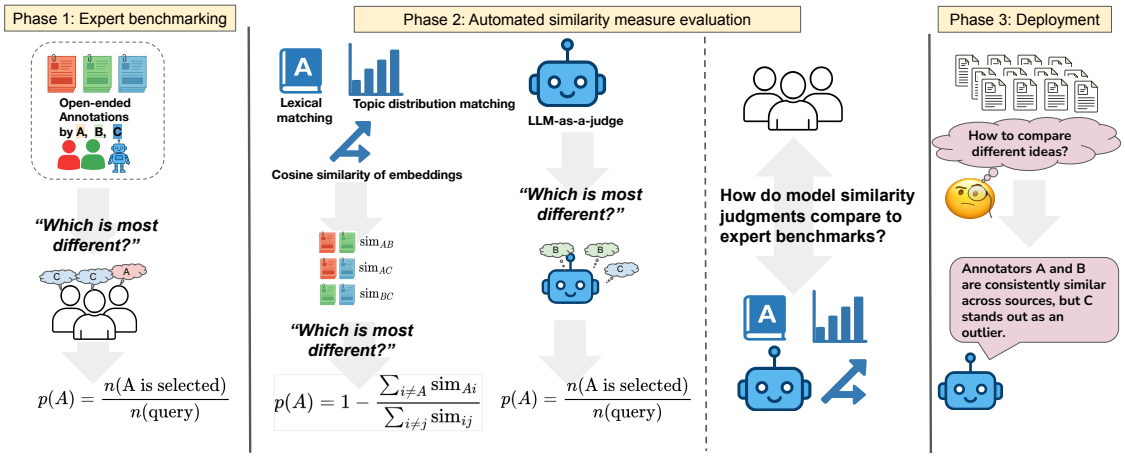

Figure 1: IDEAlign is a framework for evaluating alignment between LLM-generated and expert annotations on open-ended, interpretive tasks. It consists of three stages: **(1) Benchmarking** expert similarity judgments via *odd-one-out* tasks, **(2) Validating** automated (model) similarity methods (e.g., lexical, embedding-based, topic-based, and LLM-as-a-judge) against expert benchmarks by comparing answer distributions, and **(3) Deploying** the best-validated model to assess similarity of ideas generated by LLMs and domain experts at scale.

Radford et al., 2019; Greene et al., 2022; Neelakantan et al., 2022; Pillutla et al., 2021) and topic-based measures (Grootendorst, 2022; Blei et al., 2001) conflate content with style/length and offer limited control over the dimension of comparison. LLM-as-a-judge (Zheng et al., 2023) can provide a promising alternative but requires careful evaluation, as LLMs show sensitivity to prompt templates (Wei et al., 2025) and stylistic biases (Feuer et al., 2025). However, these methods have yet to be validated against human similarity judgments for open-ended, interpretive tasks.

We introduce IDEAlign, a benchmarking framework that elicits expert similarity judgments via *pick-the-odd-one-out* tasks. Given three annotations of the same source document, experts select the most dissimilar one along a specified, domain-relevant criterion (e.g., "What change to the student's writing is being recommended?"). This triplet design follows prior comparative judgment and "topic intruder" work (Chang et al., 2009), leveraging the well-established reliability of relative judgments, which dates back from Thurstone's *Law of Comparative Judgment* (Thurstone, 1927) and the influential Bradley-Terry model (Bradley and Terry, 1952) to more recent research and applied studies involving subjective evaluation (Laming, 2003; Chang et al., 2009; Tarricone and Newhouse, 2016; Routh et al., 2023), such as the widely adopted Adaptive Comparative Judgment (Pollitt, 2012) in educational settings, where teachers or peer evaluators compare the relative quality of two pieces of work rather than judging each one individually. Pollitt (2012) makes an argument that relying on implicit judgments of quality through pairwise comparisons with "intrinsically simple" task can give reliable results without having to specify individual criteria or calibrate subjective scores across many evaluators. In IDEAlign, expert selections yield a probability distribution over the "odd" item; model predictions are compared to this distribution using Hellinger distance.

Using IDEAlign, we evaluate families of automated similarity measures: (i) lexical overlap (BLEU), (ii) cosine similarity from 16 embedding models with/without PCA-based post-processing, (iii) topic-distribution divergence, where the topics are found using topic modeling methods like BERTopic and MALLET-LDA, and (iv) LLM-as-a-judge. Methods other than LLM-as-a-judge can only output pairwise similarity scores rather than directly answering the odd-one-out task, so we convert the models' pairwise scores to the probability of each item being selected in a triplet (Figure 1).

We study two education tasks where expert, idea-level interpretation matters: (1) assessing students' mathematical reasoning from classroom transcripts, and (2) providing inline feedback to student essays. Across both settings, most similarity measures fail to produce answers that agree with experts, and in particular, they show sensitivity to confounders like style and length. By contrast, **prompting LLMs to answer the odd-one-out task directly yields the strongest agreement with experts (11–18% improvement over other methods)**. Since LLM-as-a-judge still falls short of expert agreement, we

position it as a helpful tool that can reduce human workload rather than a substitute for expert review.

Our work demonstrates the difficulty of evaluating open-ended LLM annotations at scale, and positions IDEAlign as a reusable protocol for comparing model and human similarity judgments. Our goal is to inform responsible deployment of LLMs in education and beyond.

2 Related Work

Comparing LLMs to humans in open-ended annotations. LLMs are moving beyond tasks like storytelling and dialogue generation (Ismayilzada et al., 2025; Wiegreffe et al., 2022; Ouyang et al., 2022) toward more complex, interpretive work that requires data synthesis and reasoning based on domain knowledge (Tan et al., 2024b; Yu et al., 2023; Spangher et al., 2024). For example, in educational settings, LLM-based tools are used to assess student thinking and generate feedback as teachers (Wen et al., 2024; Graesser et al., 2004; Khan, 2023; Google, 2024; Wang et al., 2024b). However, most evaluation of LLM-generated annotations is designed for classification tasks with pre-defined coding schemes (Wadhwa et al., 2024; Larionov et al., 2019) or for assessing tone (Tan et al., 2024a) or writing quality along dimensions, like helpfulness and creativity (Ismayilzada et al., 2025; Spangher et al., 2024; Wiegreffe et al., 2022). However, for open-ended annotations, there is no single correct answer to compare against, and the quality of ideas is challenging to evaluate with ratings. Other works on evaluation focus on lexical features, length, and grammar (Muñoz-Ortiz et al., 2024; Martínez et al., 2025), but ignore content alignment, which is critical for assessing the reliability of LLM-generated annotations for tasks that require domain expertise. As an alternative to assuming a single ground truth, MAUVE (Pillutla et al., 2021) compares the distributions of human and LLM-generated texts using KL divergence. However, their method relies on text embeddings to estimate these distributions and is therefore constrained by the performance of embedding models, which are sensitive to length and style. Crucially, to our knowledge, the alignment of ideas between LLMs and experts on open-ended, interpretive annotations has not been investigated.

Automated similarity metrics. Many standard NLP metrics have been developed for text comparison such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004). However, their focus on lexical overlap fails to capture similarity in underlying ideas. **Cosine similarity of text embeddings** is also a popular method, and different embedding models have been developed and studied, including Word2Vec (Ng and Abrecht, 2015), BERT (Devlin et al., 2019; Zhao et al., 2019; Zhang et al., 2020), Sentence-BERT (Reimers and Gurevych, 2019), GPT-2 (Radford et al., 2019), GTE (Li et al., 2023), and proprietary models like ada (Greene et al., 2022) and cpt text (Neelakantan et al., 2022). Mu et al. (2018) proposed post-processing based on PCA (removing projections of the top principal components from word embeddings) to improve performance on NLP semantic similarity benchmarks. However, it remains unclear how well these embedding-based similarity scores align with expert judgments. In fact, Fabbri et al. (2021) shows that many models show poor correlations with human evaluation. Muennighoff et al. (2023) also shows that no single embedding model consistently achieves the best performance across NLP tasks.

An alternative to embeddings is to represent annotations as **topic distributions** using methods like LDA (Blei et al., 2001) and BERTopic (Grootendorst, 2022). However, as our experiments will show, depending on the number of topics, the representations may be too coarse to yield meaningful comparisons, especially when most annotations in our datasets discuss the same broad topic.

3 IDEAlign

We propose IDEAlign, a benchmarking method for eliciting expert similarity judgments and evaluating automated measures against human benchmarks.

3.1 Eliciting expert similarity judgments via *odd-one-out* tasks

A naive way of collecting expert data is to ask them for continuous similarity ratings on a fixed scale for every pair of annotations. However, absolute judgments are cognitively difficult and may yield noisy, hard-to-calibrate labels. Prior work has therefore relied on relative judgments as an alternative to absolute scoring (Laming, 2003; Jones and Alcock, 2012; Pollitt, 2012; Tarricone and Newhouse, 2016; Routh et al., 2023). Our task design builds on this line of work, including the "topic intruder/odd-one-out" paradigm introduced by Chang et al. (2009).

In the *odd-one-out* task, evaluators are provided a group of three annotations A, B, C written about

the same source text. They select which annotation is most different along domain-specific similarity criteria established by experts. We repeat this evaluation step with many triplets and multiple evaluators to aggregate answers. For each triplet (A, B, C), this yields an empirical distribution that represents how likely each item $x \in \{A, B, C\}$ is to be selected as most dissimilar:

$$P(x|A, B, C) = \frac{\#(x \text{ is picked})}{\#(A, B, C \text{ occurs in triplet})} \quad (1)$$

This allows us to construct an evaluation dataset that accounts for human uncertainty in identifying an outlier. We compare model predictions to this distribution using Hellinger distance. We additionally ask the evaluators to rate the difficulty of picking the odd one out from each triplet on a scale of easy, moderate, and difficult.

We ensured that the similarity criteria being used were meaningful and grounded in educational practices by running a co-design process with 2 experts for the math reasoning and 4 experts for the feedback data. Through this process, we iterated on the similarity definition and evaluator instructions using 3-6 task samples. This led to the task guidelines and UI in Figure 2. The template includes a section for metadata about the source text and expert-guided similarity criteria.

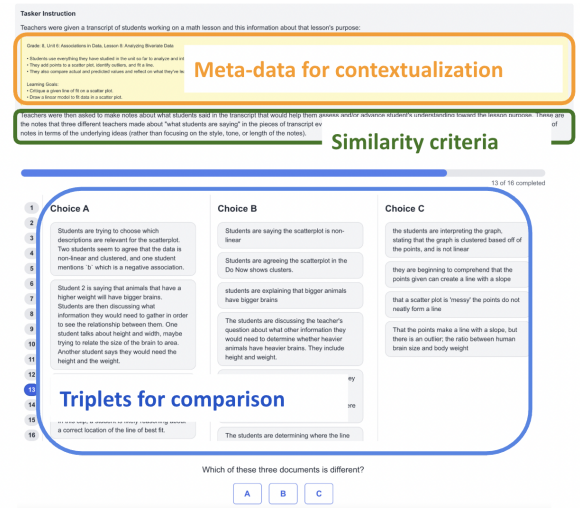

Figure 2: Evaluator interface for the odd-one-out task.

3.2 Automated similarity measures

Expert judgments are the gold standard for comparing annotations, but they are expensive to collect and therefore difficult to scale. In contrast, automated similarity methods offer a scalable alternative, but they require validation to ensure agreement with expert judgments. To this end, we evaluate 36 automated methods falling into one of four approaches: lexical matching; cosine similarity of text embeddings; divergences of topic distributions, and LLM-as-a-judge.

Lexical matching: BLEU (?Lin and Och, 2004) is a widely used metric for lexical similarity based on n-grams. Since BLEU is asymmetric (measuring how much of a prediction overlaps with a target), we convert this to a symmetric metric between annotations A and B as follows:

$$\frac{\text{BLEU}(A, B) + \text{BLEU}(B, A)}{2}. \quad (2)$$

We use a maximum n-gram order of 4.

Cosine similarity of text embeddings: Texts can be embedded as vectors, and their similarity is measured using cosine distance. We examine 16 different text embedding models (Table 1). For models with limited context windows, we split annotations into sentences and average sentence embeddings. We also implement PCA-based post-processing of the text embeddings proposed by Mu et al. (2018) which removes projections of the top principal components from the text embeddings.

Text embedding model
Sentence-BERT (Reimers and Gurevych, 2019)
MiniLM L6, 12 (Wang et al., 2020)
MPNet (Song et al., 2020)
SimCSE (Gao et al., 2021)
GTR-T5 (large, xl) (Ni et al., 2021)
ModernBERT (Warner et al., 2024)
E5-Base (Wang et al., 2024a)
GTE (base, large) (Li et al., 2023)
GPT2 (large, xl) (Radford et al., 2019)
cpt-text (small, large) (Neelakantan et al., 2022)
ada (Greene et al., 2022)

Table 1: **List of text embedding models used for evaluation.** Some models, like GPT2, are from the same family but have different number of parameters.

Divergences of topic distributions: Texts can be represented by topic distributions found using algorithms like BERTopic (Grootendorst, 2022) and LDA (Blei et al., 2001). Pairwise dissimilarity then becomes equivalent to measuring divergence between topic distributions of annotation pairs, T_A and T_B. We use Hellinger distance as a divergence metric because it can allow for cases where some

topics appear in only one of the annotations. We define similarity as one minus the distance:

$$\text{sim}(A,B) = 1 - \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^{k \text{ topics}} (\sqrt{T_A[i]} - \sqrt{T_B[i]})^2} \quad (3)$$

Our hyperparameter search across different numbers of topics k shows that k affects agreement because it controls whether representations are too coarse (small k) or too fine-grained (large k). Specifically, we searched through minimum cluster sizes $2-15$ in BERTopic (HDBSCAN) and k topic numbers through $(50, 75, 100, 125, 150)$ in LDA, and reported the best result.[2]

LLM-as-a-judge for similarity: Unlike previous approaches that output pairwise similarity scores between annotations A and B, LLM-as-a-judge (Zheng et al., 2023) can directly answer the odd-one-out tasks. The model is prompted with the same triplet multiple times ($n = 5$), so each item's probability of being selected can be aggregated from multiple model answers. We test this method with different model types (GPT-4.1 (OpenAI et al., 2024) and Claude-Sonnet-4 (Anthropic, 2025)), temperatures (0.2 and 0.8), and prompt templates (with or without additional domain information). We used the prompt below for the feedback data (reasoning prompt in Appendix H.6):

***Odd-One-Out* selection with LLM-as-a-Judge.**

You are a helpful high school English Language Arts teaching assistant. The student was instructed to write an essay to the prompt: `Include meta data about the essay topic`, and three teachers were asked to give the student feedback for revising their writing. Select which feedback is most different in terms of what it asks the student to change in their writing. Focus on the feedback content, rather than judging based on the delivery (e.g., teacher's writing style, tone, length, use of questions versus directives in the feedback). Respond with ##A## if feedback A is most different, ##B## if feedback B is most different, and ##C## if feedback C is most different.

We also experimented with prompting LLMs for direct (pairwise) similarity scores, and found that this approach performs worse than prompting for the odd-one-out tasks (Appendix A), so we keep only the triplet task setup for our main experiments.

[2]BERTopic is implemented with open source packages BERTopic, UMAP and HDBSCAN, and LDA is implemented by maria-antoniak/little-mallet-wrapper on top of MALLET (McCallum, 2002).

3.3 Evaluation of model similarity judgments with expert benchmarks

Since most automated similarity methods, except for LLM-as-a-judge, cannot directly pick the odd-one-out, we transform the pairwise similarity scores produced by the model into the probability of each item in a triplet being the most different. We assume that the probability of an item not being selected as the odd one out is proportional to that item's similarity with the other two items in the triplet. This leads to the model-predicted distribution that represents how likely each item $x \in \{A, B, C\}$ is to be selected:

$$\widehat{P}(x|A,B,C) = 1 - \frac{\sum_{j \neq x, j \in \{A,B,C\}} \text{sim}(x,j)}{\sum_{i \neq j, i,j \in \{A,B,C\}} \text{sim}(i,j)}. \quad (4)$$

This is compared to the human empirical distribution from Eq. 1 using Hellinger distance. Hellinger distance ranges from 0 (perfect agreement) to 1. Our experiment results report the average distance between the model predictions and the expert answers across all triplets in the evaluation dataset.

We chose Hellinger distance over other metrics, such as KL divergence, because KL asymmetrically penalizes under-estimation more than over-estimation, leading to counter-intuitive results. For example, given expert distribution $[0.95, 0.05, 0]$, KL divergence is lower for model prediction $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ than for $[1, 0, 0]$, even though the latter correctly identifies the most dissimilar item as experts and therefore we would expect this to get a smaller penalty than uniform guessing.

4 Datasets

We applied IDEAlign to two educational annotation tasks: assessing student mathematical reasoning and feedback generation.

Student mathematical reasoning. Eleven expert math teachers and four models each annotated four deidentified transcripts[3] (ChatGPT and Claude Sonnet 3.5 each generated two annotations per transcript) of 6th-8th grade math lessons with inline comments. These comments analyzed students' mathematical reasoning, focusing on how specific pieces of student dialogue revealed their understanding of the key mathematical concepts. Annotations averaged 139.38 words (maximum: 556

[3]These transcripts were collected from middle schools in a large urban public school district in California under an IRB for human subject research #75294.

words). Ten expert math educators (with more than 8 years of experience) then completed a total of 640 odd-one-out tasks, evaluating similarity based on the annotator's assessment of the students' understanding about the lesson's goals.

Essay feedback. We used 18 student essays from open-source datasets (Hamner et al., 2012), with inline feedback from 32 experienced English Language Arts (ELA) educators (Mah et al., 2025). We additionally included four LLM-generated feedback per student essay (ChatGPT, Gemini 2.5 Pro, Claude Sonnet 3.5, and DeepSeek V2.5). Annotations averaged 142.98 words (maximum: 477 words). These comments focused on various aspects of student writing that needed revision (e.g., grammar, word choice, and organization). Seven ELA experts (one with 6 years of teaching and the rest with more than 8 years of experience) completed a total of 458 odd-one-out tasks, evaluating similarity based on the types of revisions requested by the teachers.

Expert evaluators on both domains were compensated at an hourly rate of $50, and each contributed between 3-5 hours to the odd-one-out tasks. Further data collection details are in Appendix H.

5 Results

Using these 36 similarity measures (including 16 text embeddings from Table 1 and their post-processing variants) and two datasets, we quantify consistency across the measures (RQ1). Next, evaluate similarity measures against expert judgments (RQ2), and probe when/why they diverge by looking at task difficulty, style sensitivity, and lexical overlap (RQ3). Finally, we assess how the best-performing method can triage human work through outlier detection and annotator-pair ranking (RQ4).

RQ 1: ***How much do automated similarity measures correlate with each other?*** We computed Pearson correlations between pairwise similarities from (i) BLEU (4-gram), (ii) 16 text-embedding models, (iii) topic distributions (LDA, k=100 topics), and (iv) LLM-as-a-judge (prompting for both triplet judgments and continuous pairwise scores). While the triplet judgment setup (*LLM-as-a-judge triplet*) is our preferred specification for LLM-as-a-judge, here we also include a variant where GPT-4.1 is prompted to produce continuous pairwise similarity scores for comparability (*LLM-as-a-judge direct*); we sampled five outputs per pair and averaged them.

On the reasoning data (Fig. 3), 56 out of 190 cross-method pairs show non-significant correlations (based on $p < 0.05$). Embedding-based measures show moderate to high correlations with one another, but they do not correlate significantly with BLEU, LDA, and both variants of LLM-as-a-judge. LDA also correlates weakly with the LLM-as-a-judge. LLM-as-a-judge direct and triplet correlate strongly with each other. On the feedback data, correlations of cross-method pairs are generally higher (Appendix 5).

Inconsistency among measures means that they can lead to conflicting conclusions about which annotations are "similar." This highlights the need to validate automated measures against human judgments and to do so *by domain*, as reliability of different methods may vary across tasks and datasets.

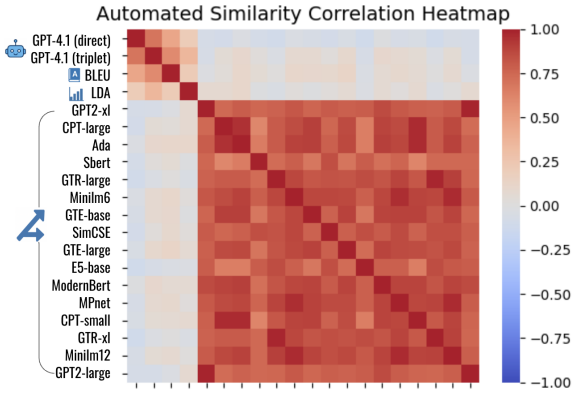

Figure 3: Pearson correlation coefficients of different similarity measures on the reasoning data. Cross-family pairs show no to weak correlation, while correlations within embedding-based measures are higher.

RQ 2: ***How well do automated similarity methods align with expert similarity judgments?*** We quantify alignment by comparing each method's predictions on the odd-one-out triplets to the expert answer distributions using Hellinger distance (lower is better; 0 = perfect match). A uniform guess serves as a naïve baseline. As shown in Table 2, BLEU and embedding-based similarities perform near this baseline (especially on reasoning), while *LLM-as-a-judge* is best in both domains, improving alignment by 18% over alternatives in feedback and 11% improvement in reasoning. This trend holds across model types (GPT-4.1, Claude Sonnet 4); lower temperature (0.2) outperforms 0.8; and surprisingly, adding extra education-related context does not help (Appendix E). Extensive embedding variants, with and without PCA post-processing, remain close to uniform on the reason-

ing dataset (Appendix D).

Method	Reasoning	Feedback
Uniform	0.650 (± 0.000)	0.493 (± 0.015)
Lexical matching (BLEU)	0.650 (± 0.000)	0.449 (± 0.026)
Text embedding (gpt2-xl)	0.653 (± 0.009)	0.481 (± 0.004)
Text embedding (ada)	0.650 (± 0.000)	0.482 (± 0.004)
Topic distribution (LDA)	0.639 (± 0.002)	0.476 (± 0.015)
Topic distribution (BERTopic)	0.619 (± 0.006)	0.495 (± 0.016)
LLM-as-judges (GPT-4.1)	**0.541 (± 0.006)**	**0.374 (± 0.029)**

Table 2: Average Hellinger distances between the model outputs and the expert answers to the odd-one-out tasks (640 expert labels from reasoning and 458 from feedback). Values range between 0 and 1, with 0 indicating perfect agreement. We report the mean and the standard error.

Does model performance vary by task difficulty? Some triplets are inherently more difficult for experts to identify the odd one out from. We captured task difficulty by collecting ratings of each triplet on a three-point scale: difficult (-1), neutral (0), and easy (1). We averaged these ratings across experts and compared model performance on the easy (mean rating > 0) and difficult (mean < 0) subsets. This yielded 38% easy and 11% difficult triplets for reasoning, and 65% easy and 16% difficult for feedback. We computed average Hellinger distances within each subset and tested for significant differences using two-tailed independent samples t-tests. Table 3 shows that only LLM-as-a-judge achieves significantly better performance on the easy tasks consistently across two domains.

Method	Reasoning (Easy)	Reasoning (Hard)
Lexical matching (BLEU)	0.65 (0.65)	0.65 (0.65)
Text embedding (gpt2-xl)	0.65 (0.65)	0.65 (0.65)
Text embedding (ada)	0.65 (0.65)	0.65 (0.65)
Topic distribution (LDA)	**0.62 (0.63)***	**0.64 (0.64)***
Topic distribution (BERTopic)	0.58 (0.61)	0.63 (0.65)
LLM-as-judges (GPT-4.1)	**0.49 (0.39)****	**0.58 (0.56)****
LLM-as-judges (Sonnet 4)	**0.49 (0.39)****	**0.61 (0.63)****
Method	**Feedback (Easy)**	**Feedback (Hard)**
Lexical matching (BLEU)	0.42 (0.43)	0.50 (0.54)
Text embedding (gpt2-xl)	0.50 (0.48)	0.46 (0.43)
Text embedding (ada)	0.51 (0.48)	0.46 (0.43)
Topic distribution (LDA)	0.49 (0.49)	0.47 (0.44)
Topic distribution (BERTopic)	0.49 (0.51)	0.51 (0.47)
LLM-as-judges (GPT-4.1)	**0.36 (0.32)****	**0.56 (0.55)****
LLM-as-judges (Sonnet 4)	**0.35 (0.32)****	**0.58 (0.55)****

Table 3: Model performance on easy and difficult task subsets. We report the mean Hellinger distance (and the median in the parentheses) for each subset. P values, based on a two-tailed t-test: $*p < 0.05$, $**p < 0.01$. Only LLM-as-a-judge shows significant performance difference on both datasets.

To provide a different lens on this gap, we also report model prediction accuracy: 1 if the model matches the majority expert answer, and 0 otherwise. GPT-4.1 as a judge improves from 44.1% (hard) to 70.6% (easy) in reasoning, and Claude Sonnet 4 from 39.7% to 71.4%. Similar trends are observed in feedback (Appendix F). Overall, these evaluation results show that LLM-as-a-judge aligns most closely with expert similarity judgments, *especially when clear outliers exist,* but there remains considerable room for improvement for all automated methods.

RQ 3: ***What might cause models to disagree with experts?*** We hypothesize two possible factors contributing to model misalignment, especially for embedding-based and lexical similarities: (i) sensitivity to style, and (ii) lack of lexical overlap with the source text due to the interpretive nature of the annotations.

Style. We noticed a discrepancy between the annotations that embedding models identified as most dissimilar and those chosen by experts. We hypothesized that this was due to differences in writing style (formal vs casual). To test this, we took two pieces of annotations from the reasoning dataset with the lowest average similarity scores and rewrote them in a more formal style without changing their content (see Appendix G for changes). Table 4 shows that the embedding similarities increase by 20–85% after this rewriting, whereas LLM-as-a-judge is more stable, showing only 2-5% change.

Text embedding model	**% change in similarity scores**
Sentence-BERT	17.35 ↑
ModernBERT	23.45 ↑
GPT-XL	26.08 ↑
MPNet	68.16 ↑
MiniLM-L12	82.30 ↑
LLM-as-a-judge	**% change in similarity scores**
GPT-4.1	5.00 ↑
Claude Sonnet 4	2.03 ↓

Table 4: We report the percentage difference (%) in similarity scores of different models after the style manipulation. Robust models should not result in dramatic score changes, because we controlled the content to be the same.

Lack of grounding in the source text. Table 2 shows that BLEU's average Hellinger distances are close to uniform guessing. BLEU is uninformative when annotations have little to no lexical overlap. This is expected for open-ended, interpretive tasks, where annotators (teachers) paraphrase rather than directly quote the student text. BLEU scores between annotations and source texts are extremely low: 0.0127 (SD 0.022) for feedback and 0.0004 (SD 0.002) for reasoning. Given this little overlap with the source text, low BLEU scores among annotations are unsurprising. While BLEU

is useful for summarization and translation, where lexical similarity to a target is desirable and expected, it performs poorly for tasks that involve interpreting and paraphrasing from one text form (e.g., student dialogue or essays) to another (e.g., teacher's assessment of student reasoning or feedback comments).

RQ 4: ***What insights do automated similarity methods provide that can reduce human workload?*** The goal of the previous section was to evaluate and select reliable automated similarity methods for comparing LLM interpretations to domain experts. While agreement with expert judgments can be improved across all methods, LLM-as-a-judge may still be useful for answering some data-driven questions that would otherwise require manual human evaluation. We explore this potential through two tasks: outlier annotation detection and ranking annotator pairs by similarity.

LLM-as-judges	Reasoning %	Feedback %
GPT-4.1	86.67	61.90
Claude 4 Sonnet	80.00	71.43

Table 5: Outlier detection rate by LLM-as-a-judge using the top quartile of the expert-selected odd ones as the true outliers.

Outlier detection. One natural question to ask about LLM and human-generated annotations is whether there exist clear outliers. We define outliers as annotations that had the highest possibility of being selected as the odd-one across different triplets. We used the top quartile from the expert data to establish the outlier thresholds and assessed whether the model's top quartile could detect the same outliers. Table 5 shows that GPT-4.1 and Sonnet 4 detect over 80% of the outliers in reasoning and over 60% in feedback.

LLM-as-judges	Reasoning ρ	Feedback ρ
GPT-4.1	0.683**	0.618**
Claude Sonnet 4	0.589**	0.613**

Table 6: Spearman rank correlation ρ between LLM and expert produced rankings of pairwise similarities. **: $p < 0.01$. We convert the answers to the odd-one-out tasks by LLMs and human experts into pairwise similarity scores by up-voting pairs that appeared in the same triplet but were not chosen (higher similarity) and down-voting pairs where either one was chosen (lower similarity). We generate rankings of the most and least similar pairs according to these pairwise scores and correlate the rankings generated by LLMs vs humans.

Ranking annotator pairs. Researchers may be interested in comparing annotator pairs (LLM-LLM, human-human, human-LLM) to see whether LLMs are more similar to each other than to humans, and whether human-to-human variability is greater than differences among LLMs. Doing so requires ranking all annotator pairs, which is costly for humans to evaluate because the number of pairs grows as n choose 2. As an alternative, LLM-as-a-judge can be used to generate pairwise similarity rankings. We demonstrate this by converting odd-one-out task responses into relative similarity scores and ranking annotator pairs using these derived scores. We repeat the same steps with expert relative judgments and compare the resulting LLM and expert rankings using Spearman rank correlation. Table 6 shows that LLM-as-a-judge achieves moderately high correlations with experts ($\rho : 0.58 - 0.68$, and $p < 0.01$) on both datasets. These results demonstrate that automated similarity methods can support scalable data exploration (e.g., identifying outliers and ranking similar annotator pairs) providing insights that would otherwise require costly human evaluation.

6 Discussion & Conclusion

As interest grows in evaluating LLM outputs against experts' for open-ended tasks (Spangher et al., 2024; Muñoz-Ortiz et al., 2024; Ismayilzada et al., 2025), we need evaluation that goes beyond surface features like word overlap or style, and capture alignment of underlying ideas. Our evaluation shows that most existing similarity measures show poor agreement with experts. In contrast, LLM-as-a-judge achieves 11-18% improvement in agreement with experts than lexical and vector-based methods, and performs especially well when clear outliers exist. Among the models evaluated, we conclude that prompting LLMs for relative judgments offers the most promising path for automated idea-level similarity evaluation, but we recommend domain-specific validation against human benchmarks. For NLP and social science researchers, unstructured data exploration, such as clustering similar ideas and comparing shared themes, is crucial but laborious and time-intensive. Automated similarity methods can reduce human workload and support large-scale data analysis by highlighting similarities and differences of ideas in unstructured annotations. We encourage researchers from different domains to consider applying IDEAlign and other validation approaches to domains beyond education.

Future direction. An interesting future direction to explore is whether similarity metrics/models behave differently when judging LLM-generated versus human-generated text pairs. For example, it would be worthwhile to investigate if models are more or less effective at judging similarity in ideas generated by LLMs (either from the same or different model families) than at comparing human-generated pairs. Furthermore, applying our evaluation framework to LLM-generated text pairs can help provide insights into models' output homogeneity highlighted in recent work (Jiang et al., 2025; Zhang et al., 2025).

Limitations

Challenge of directly quantifying pairwise similarity of ideas. While it would be ideal to collect pairwise similarity scores from humans given a text pair, doing so reliably at scale is difficult. We believe that there is no well-established method for humans to estimate the "similarity of ideas behind texts," and propose it as a crucial challenge as open-ended evaluations of LLM-generated texts become increasingly important and popular. Instead of collecting pairwise similarities from humans, this work proposes "odd-one-out" evaluation as an alternative to reduce human cognitive load and increase consistency. While this differs theoretically from absolute pairwise distance, it is a relatively easy and intuitive task for humans to perform. We demonstrate how to build an evaluation protocol for models and existing semantic similarity measures that can match human outputs. Our insight is to compare humans and models based on how well the models' similarity estimates match humans' triplet answer distributions.

Human guidance & validation. Defining dimensions of similarity with domain experts is crucial to our analysis. For instance, even within math education, the criteria for measuring similarity may vary depending on the downstream task and the data from which the annotations are generated. Therefore, collaborating with human experts is essential, though it can be expensive compared to fully automated evaluation pipelines.

Focus of our work on education. In this paper, we focus on two educational tasks, assessing student math reasoning and providing essay feedback, but there are many other tasks, both within and beyond education that also require domain expertise to generate meaningful interpretations and measure similarity. While our general evaluation framework can be applied to various open-ended, interpretive text-generation tasks, we recommend that validating with domain-specific expert oversight is crucial.

Challenges of aligning LLMs with domain experts. While LLM-as-a-judge shows promise compared to other similarity measures, our experimental results suggest that they still lack domain expertise appropriate for full automation, and prior work has questioned the reliability of their outputs (Feuer et al., 2025). Therefore, we recommend careful validation of all methods discussed in our work, including LLM-as-a-judge, before applying them to any new dataset.

Ethical considerations

Math classroom transcript data was collected under an IRB for human subject research (approval number: 75294) in which teachers, students, and student guardians consented to their participation. The feedback data was collected in prior work (Mah et al., 2025; Tan et al., 2024a). Recruited and compensated experts for similarity assessments only used deidentified data.

Regarding the intended use of our method, we propose a benchmarking task for measuring the similarity of ideas as a scalable way to assess LLM-generated responses against those of expert humans. However, high similarity scores do not imply endorsing indiscriminate use of LLMs in education. Other social and educational risks of using LLMs in classrooms or to interact with students should be considered holistically, regardless of their idea-level similarity to human teachers.

References

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. 2023. Prediction-powered inference. *Science*, 382(6671):669–674.

John Annett. 2002. Subjective rating scales: science or art? *Ergonomics*, 45(14):966–987. PMID: 12569049.

Anthropic. 2025. System card: Claude opus 4 & claude sonnet 4.

David Blei, Andrew Ng, and Michael Jordan. 2001. Latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press.

Ljubiša Bojić, Olga Zagovora, Asta Zelenkauskaite, Vuk Vuković, Milan Čabarkapa, Selma Veseljević Jerković, and Ana Jovančević. 2025. Comparing large language models and human annotators in latent content analysis of sentiment, political leaning, emotional intensity and sarcasm. *Scientific reports*, 15(1):11477.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-graber, and David Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Benjamin Feuer, Micah Goldblum, Teresa Datta, Sanjana Nambiar, Raz Besaleli, Samuel Dooley, Max Cembalest, and John P Dickerson. 2025. Style outweighs substance: Failure modes of LLM judges in alignment benchmarking. In *The Thirteenth International Conference on Learning Representations*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

LearnLM Team Google. 2024. Learnlm: Improving gemini for learning. *arXiv preprint arXiv:2412.16429*.

Arthur C Graesser, Shulan Lu, George Tanner Jackson, Heather Hite Mitchell, Mathew Ventura, Andrew Olney, and Max M Louwerse. 2004. Autotutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers*, 36(2):180–192.

Ryan Greene, Ted Sanders, Lilian Weng, and Arvind Neelakantan. 2022. New and improved embedding model. `https://openai.com/index/new-and-improved-embedding-model/`.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Ben Hamner, Jaison Morgan, lynnvandev, Mark Shermis, and Tom Vander Ark. 2012. The hewlett foundation: Automated essay scoring. `https://kaggle.com/competitions/asap-aes`. Kaggle.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1051–1068, Singapore. Association for Computational Linguistics.

Mete Ismayilzada, Claire Stevenson, and Lonneke van der Plas. 2025. Evaluating creative short story generation in humans and large language models. *Preprint*, arXiv:2411.02316.

Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten Sap, Alon Albalak, and Yejin Choi. 2025. Artificial hivemind: The open-ended homogeneity of language models (and beyond). *Preprint*, arXiv:2510.22954.

Ian Jones and Lara Alcock. 2012. Summative peer assessment of undergraduate calculus using adaptive comparative judgement. *Mapping university mathematics assessment practices*, pages 63–74.

Sal Khan. 2023. Harnessing gpt-4 so that all students benefit. a nonprofit approach for equal access. *Khan Academy Blog*.

Donald Laming. 2003. *Human judgment: The eye of the beholder*. Cengage Learning EMEA.

Daniil Larionov, Artem Shelmanov, Elena Chistova, and Ivan Smirnov. 2019. Semantic role labeling with pretrained language models for known and unknown predicates. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 619–628, Varna, Bulgaria. INCOMA Ltd.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Chin-Yew Lin and Franz Josef Och. 2004. ORANGE: a method for evaluating automatic evaluation metrics for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 501–507, Geneva, Switzerland. COLING.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Christopher Mah, Mei Tan, Lena Phalen, Alexa Sparks, and Dora Demszky. 2025. From sentence-corrections to deeper dialogue: Qualitative insights from llm and teacher feedback on student writing. *Available at SSRN 5213040*.

Gonzalo Martínez, José Alberto Hernández, Javier Conde, Pedro Reviriego, and Elena Merino-Gómez. 2025. Beware of words: Evaluating the lexical diversity of conversational llms using chatgpt as case study. *ACM Transactions on Intelligent Systems and Technology*, 16(6):1–15.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. Http://mallet.cs.umass.edu.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2018. All-but-the-top: Simple and effective post-processing for word representations. *Preprint*, arXiv:1702.01417.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez, and David Vilares. 2024. Contrasting linguistic patterns in human and llm-generated news text. *Artificial Intelligence Review*, 57(10).

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, Madeleine Thompson, Tabarak Khan, Toki Sherbakov, Joanne Jang, Peter Welinder, and Lilian Weng. 2022. Text and code embeddings by contrastive pre-training. *Preprint*, arXiv:2201.10005.

Jun-Ping Ng and Viktoria Abrecht. 2015. Better summarization evaluation with word embeddings for ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1925–1930, Lisbon, Portugal. Association for Computational Linguistics.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y. Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. 2021. Large dual encoders are generalizable retrievers. *Preprint*, arXiv:2112.07899.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov,

Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Preprint*, arXiv:2102.01454.

Alastair Pollitt. 2012. The method of adaptive comparative judgement. *Assessment in Education: principles, policy & practice*, 19(3):281–300.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Vikas Raunak, Vaibhav Kumar, Vivek Gupta, and Florian Metze. 2020. On dimensional linguistic properties of the word embedding space. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 156–165, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Preprint*, arXiv:1908.10084.

Jennifer Routh, Sharmini Julita Paramasivam, Peter Cockcroft, Sarah Wood, John Remnant, Cornélie Westermann, Alison Reid, Patricia Pawson, Sheena Warman, Vishna Devi Nadarajah, et al. 2023. Rating and ranking preparedness characteristics important for veterinary workplace clinical training: a novel application of pairwise comparisons and the elo algorithm. *Frontiers in Medicine*, 10:1128058.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. 2023. Whose opinions do language models reflect? In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 29971–30004. PMLR.

Kayla Schroeder and Zach Wood-Doughty. 2025. Can you trust llm judgments? reliability of llm-as-a-judge. *Preprint*, arXiv:2412.12509.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Alexander Spangher, Nanyun Peng, Sebastian Gehrmann, and Mark Dredze. 2024. Do LLMs plan like human writers? comparing journalist coverage of press releases with LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21814–21828, Miami, Florida, USA. Association for Computational Linguistics.

James WA Strachan, Dalila Albergo, Giulia Borghini, Oriana Pansardi, Eugenio Scaliti, Saurabh Gupta, Krati Saxena, Alessandro Rufo, Stefano Panzeri, Guido Manzi, et al. 2024. Testing theory of mind in large language models and humans. *Nature Human Behaviour*, 8(7):1285–1295.

Mei Tan, Christopher Mah, and Dorottya Demszky. 2024a. Reframing authority: A computational measure of power-affirming feedback on student writing. In *Proceedings of the Eleventh ACM Conference on Learning@ Scale*, pages 417–421.

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024b. Large language models for data annotation and synthesis: A survey. *Preprint*, arXiv:2402.13446.

Pina Tarricone and C Paul Newhouse. 2016. Using comparative judgement and online technologies in the assessment and measurement of creative performance

and capability. *International Journal of Educational Technology in Higher Education*, 13(1):16.

L. Thurstone. 1927. A law of comparative judgment. *Psychology Review*, 34:273–86.

Somin Wadhwa, Silvio Amir, and Byron C. Wallace. 2024. Revisiting relation extraction in the era of large language models. *Preprint*, arXiv:2305.05003.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024a. Text embeddings by weakly-supervised contrastive pre-training. *Preprint*, arXiv:2212.03533.

Rose E. Wang, Qingyang Zhang, Carly Robinson, Susanna Loeb, and Dorottya Demszky. 2024b. Bridging the novice-expert gap via models of decision-making: A case study on remediating math mistakes. *Preprint*, arXiv:2310.10648.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Preprint*, arXiv:2002.10957.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

Hui Wei, Shenghua He, Tian Xia, Fei Liu, Andy Wong, Jingyang Lin, and Mei Han. 2025. Systematic evaluation of llm-as-a-judge in llm alignment tasks: Explainable metrics and diverse prompt templates. *Preprint*, arXiv:2408.13006.

Qingsong Wen, Jing Liang, Carles Sierra, Rose Luckin, Richard Tong, Zitao Liu, Peng Cui, and Jiliang Tang. 2024. Ai for education (ai4edu): Advancing personalized education with llm and adaptive learning. New York, NY, USA. Association for Computing Machinery.

Sarah Wiegreffe, Jack Hessel, Swabha Swayamdipta, Mark Riedl, and Yejin Choi. 2022. Reframing human-AI collaboration for generating free-text explanations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 632–658, Seattle, United States. Association for Computational Linguistics.

Xinli Yu, Zheng Chen, Yuan Ling, Shujing Dong, Zongyi Liu, and Yanbin Lu. 2023. Temporal data meets llm – explainable financial time series forecasting. *Preprint*, arXiv:2306.11025.

Jiayi Zhang, Simon Yu, Derek Chong, Anthony Sicilia, Michael R. Tomz, Christopher D. Manning, and Weiyan Shi. 2025. Verbalized sampling: How to mitigate mode collapse and unlock llm diversity. *Preprint*, arXiv:2510.01171.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *Preprint*, arXiv:1909.02622.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Ruiyang Zhou, Lu Chen, and Kai Yu. 2024. Is LLM a reliable reviewer? a comprehensive evaluation of LLM on automatic paper reviewing tasks. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9340–9351, Torino, Italia. ELRA and ICCL.

A Prompting LLM-as-a-judge with relative judgments versus direct score assignment.

Method	Reasoning	Feedback
Direct	0.624 (0.003)	0.452 (0.015)
Triplet	**0.541 (0.006)**	**0.374 (0.029)**

Table 7: We added a variant of prompting LLMs for direct similarity scores (*direct*) rather than answering the relative judgment tasks (*triplet*). We sampled five outputs per pair to average and converted the pairwise similarity scores into the probability of each item in a triplet being selected as the most different, following the steps discussed in Section 3.3. We report the mean and the standard error across triplets. Smaller Hellinger distances (closer to 0) indicate better agreement with experts.

B Correlations of different automated similarity measures with each other.

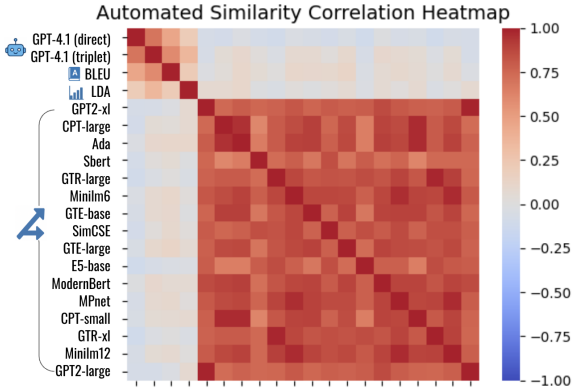

Figure 4: **Pearson correlation coefficients ρ of different similarity methods / models in math reasoning.** Negative correlations suggest that different metrics may yield different conclusions about whether the annotations are similar or not. Under one metric, the same pair may be considered to have high similarity score, but under another metric, considered to have low similarity. Pairwise scores are computed for the math data using different similarity models / methods and their correlations are computed with Pearson.

C Text embedding models used in evaluation.

See Table 8.

D Evaluation of text embedding models against expert similarity judgments

See Table 9 and 10.

E Ablations with different prompting hyperparameters.

See Table 12.

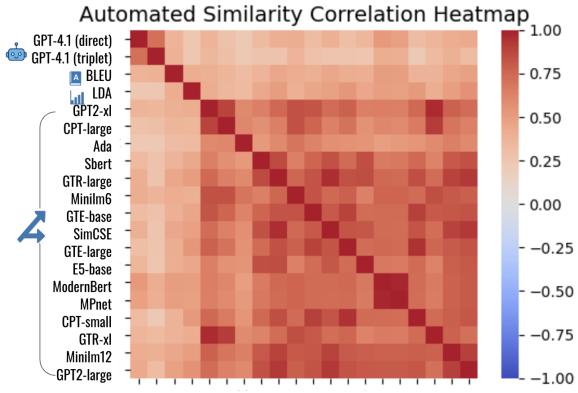

Figure 5: **Pearson correlation coefficients ρ of different similarity methods / models in essay feedback.** Negative correlations suggest that different metrics may yield different conclusions about whether the annotations are similar or not. Under one metric, the same pair may be considered to have high similarity score, but under another metric, considered to have low similarity. Pairwise scores are computed for the math data using different similarity models / methods and their correlations are computed with Pearson.

Text embedding model	Context length	Embedding dim	Open-sourced?
Sentence-BERT (Reimers and Gurevych, 2019)	128	768	✓
MiniLM L6, 12 (Wang et al., 2020)	256	384	✓
MPNet (Song et al., 2020)	382	768	✓
SimCSE (Gao et al., 2021)	512	1024	✓
GTR-T5 (large, xl) (Ni et al., 2021)	512	768	✓
ModernBERT (Warner et al., 2024)	8192	1024	✓
E5-Base (Wang et al., 2024a)	512	768	✓
GTE (base, large) (Li et al., 2023)	8192	768, 1024	✓
GPT2 (large, xl) (Radford et al., 2019)	1024	1280, 1600	✓
cpt-text (small, large) (Neelakantan et al., 2022)	8192	1536, 3072	✗
ada (Greene et al., 2022)	8192	1536	✗

Table 8: These are the text embedding models we used in our experiments to compute cosine similarity scores. We evaluated a total of 16 available models, both open-sourced and proprietary. For models with a maximum context length of less than 1024 tokens, we embedded each sentence in the document and used the average of the sentence embeddings to represent the document. If the model allowed longer contexts, we embedded the full document.

Embedding Model	Original	Post-processing by Mu et al. (2018)
gtr-large	0.651 (± 0.006)	0.688 (± 0.009)
gtr-xl	0.652 (± 0.007)	0.696 (± 0.009)
e5-base	0.650 (± 0.002)	0.678 (± 0.009)
sbert	0.652 (± 0.002)	0.682 (± 0.010))
gte-base	0.651 (± 0.002)	0.669 (± 0.009)
gte-large	0.650 (± 0.002)	0.684 (± 0.009)
simcse	0.651 (± 0.001)	0.681 (± 0.010)
mpnet	0.650 (± 0.002)	0.676 (± 0.009)
mini-l6	0.652 (± 0.006)	0.665 (± 0.009)
gpt2-large	0.653 (± 0.009)	0.673 (± 0.010)
gpt2-xl	0.653 (± 0.009)	0.699 (± 0.009)
mini-l12	0.652 (± 0.003)	0.669 (± 0.009)
modern-bert	0.650 (± 0.001)	0.667 (± 0.009)
cpt-text-small	0.651 (±0.002)	0.687 (± 0.010)
cpt-text-large	0.652 (±0.002)	0.669 (± 0.010)
ada	0.650 (±0.000)	0.672 (± 0.010)

Table 9: **Average Hellinger distances** between the model outputs and the expert answers to the odd-one-out tasks (640 expert labels from the reasoning dataset). Values range between 0 and 1, with 0 indicating perfect agreement. We report the mean and the standard deviation across the triplets.

Embedding Model	Original	Post-processing by Mu et al. (2018)	Post-processing by Raunak et al. (2020)
gtr-large	0.481 (± 0.004)	0.529 (± 0.024)	0.440 (± 0.017)
gtr-xl	0.489 (± 0.015)	0.563 (± 0.021)	0.456 (± 0.016)
e5-base	0.490 (± 0.011)	0.586 (± 0.022)	0.449 (± 0.018)
sbert	0.489 (± 0.009)	0.552 (± 0.022)	0.480 (± 0.022)
gte-base	0.481 (± 0.005)	0.532 (± 0.022)	0.468 (± 0.016)
gte-large	0.488 (± 0.007)	0.522 (± 0.021)	0.469 (± 0.015)
simcse	0.485 (± 0.006)	0.498 (± 0.022)	0.415 (± 0.018)
mpnet	0.484 (± 0.006)	0.510 (± 0.021)	0.453 (± 0.017)
mini-l6	0.482 (± 0.005)	0.552 (± 0.021)	0.463 (± 0.017)
gpt2-large	0.482 (± 0.004)	0.545 (± 0.023)	0.416 (± 0.019)
gpt2-xl	0.481 (± 0.004)	0.511 (± 0.022)	0.430 (± 0.020)
mini-l12	0.481 (± 0.004)	0.540 (± 0.020)	0.453 (± 0.018)
modern-bert	0.481 (± 0.004)	0.517 (± 0.024)	0.432 (± 0.017)
cpt-text-small	0.485 (± 0.007)	0.492 (± 0.022)	0.447 (± 0.014)
cpt-text-large	0.482 (± 0.005)	0.525 (± 0.017)	0.445 (± 0.015)
ada	0.482 (± 0.004)	0.485 (± 0.022)	0.425 (± 0.016)

Table 10: **Average Hellinger distances** between the model outputs and the expert answers to the odd-one-out tasks (458 from the feedback datasets). Values range between 0 and 1, with 0 indicating perfect agreement. We report the mean and the standard deviation across the triplets.

Temperature	**Low (0.2)**	**High (0.8)**
	0.541 (0.006)	0.542 (0.006)
Prompt template	**No metadata**	**With metadata**
	0.541 (0.006)	0.544 (0.006)

Table 11: Ablations on the reasoning dataset using GPT-4.1 as a judge. Average Hellinger distance and standard error across the triplets. Unless stated otherwise, we used the temperature of 0.2 and no metadata as the default prompting condition, which is reported in the main text.

Temperature	**Low (0.2)**	**High (0.8)**
	0.374 (0.029)	0.444 (0.022)
Prompt template	**No metadata**	**With metadata**
	0.374 (0.029)	0.455 (0.023)

Table 12: Ablations on the feedback dataset using GPT-4.1 as a judge. Average Hellinger distance and standard error across the triplets. Unless stated otherwise, we used the temperature of 0.2 and no metadata as the default prompting condition, which is reported in the main text.

F Performance of LLM-as-judges methods by task difficulty.

See Fig. 6 and Table 13.

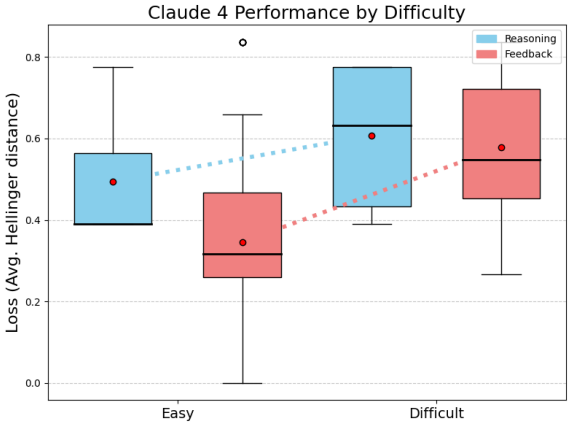

Figure 6: **LLM-as-judges (Claude Sonnet 4) performance variation on easy and difficult task subsets of reasoning and feedback datasets.**

Method	**Reasoning (Hard)**	**Reasoning (Easy)**
LLM-as-judges (GPT-4.1)	**0.350 (0.107)**	**0.734 (0.050)**
LLM-as-judges (Sonnet 4)	**0.350 (0.107)**	**0.722 (0.050)**
Method	**Feedback (Hard)**	**Feedback (Easy)**
LLM-as-judges (GPT-4.1)	**0.25 (0.097)**	**0.696 (0.052)**
LLM-as-judges (Sonnet 4)	**0.25 (0.097)**	**0.646 (0.054)**

Table 13: **Model performance in terms of accuracy on the easy and difficult task subsets.** Higher accuracy means better agreement with majority expert answers. When there are ties in the expert majority answers, we give the model a score of 1 if it matches either of the majority answers, but if the there are ties in the model answers, and there's a single expert majority answer, then we give a score of 0. We report the mean and the standard error. No methods other than LLM-as-a-judge show significant performance difference between the two subgroups split by the expert-perceived difficulty.

G Style perturbation experiment.

- Original document #1: Students are debating if 10x10 is 100 or 20., Ultimately they say that it is 100, and that 10x10x10 is 1000. Then they state that 1000 times 100 is 'one thousand one hundred', That 10^0 is 10., Students are counting 10s by 1's, 2's and 3's. The number that they count is the exponent.

- After style change #1: Students are debating if 10 multiplied by 10 should be 100 or 20. They conclude that it should be 100. Similarly, they discuss that 10 multiplied by 10, multiplied by another 10 is a thousand. Then, they reach the conclusion that 1000 times 100 is 'one thousand and one hundred.', Students argue that 10 to the power of 0 is 10., Students are counting the powers of 10.

- Original document #2: Student debate over 10 raised to 2 being 10 times 2 = 20 and 10 times 10 = 100., That after expanding the expressions you would have to add instead of multiplying.

- After style change #2: Students are debating whether 10 to the power of 2 should be interpreted as $10 \times 2 = 20$ or as $10 \times 10 = 100$., Afterwards, students are discussing whether the correct step is to add rather than multiply.

H Details about the educational tasks and datasets

H.1 Assessing students' mathematical reasoning

Annotation process Eleven expert math teachers were provided with four deidentified transcripts of 6-8th grade math lessons[4] along with the purpose of the math problems on which students worked. Each transcript contained between 300-600 lines of student and teacher utterances. Annotators marked specific lines of student utterances from each transcript to answer the following questions: *(1) What are students saying in the selected piece(s) of evidence? (2) What does this piece (or pieces) of evidence tell you about students' understanding and/or progress towards the lesson's purpose?* We also prompted ChatGPT and Claude 3.5 Sonnet to complete the same task and included two outputs per model. The average annotation length is 139.38 words, with a maximum length of 556.

***Odd-one-out* tasks with experts** The dataset of open-ended transcript annotations was then labeled for relative similarity by 10 additional math educators. This resulted in a total of 640 triplet comparisons which we used as benchmarks for evaluating automated measures. The recruited and compensated experts annotated and labeled only deidentified classroom transcript data.

H.2 Providing feedback to student writing

Annotation process We used 18 middle and high school student essays from open-source datasets (Hamner et al., 2012), with in-line feedback comments written by 32 experienced English Language Arts (ELA) educators, collected in prior work (Mah et al., 2025; Tan et al., 2024a). We additionally prompted four state-of-the-art models (Chat GPT, Gemini 2.5 Pro, Claude 3.5 Sonnet, and Deepseek-V2.5) to *provide in-line feedback to help students revise their current essay based on the teacher's suggestions*. The average combined length of all feedback comments from a feedback provider was 142.98 words, with a maximum length of 477.

***Odd-one-out* tasks with experts** The dataset of 458 points was labeled by seven ELA teachers. Teachers used the web-based interface, where they were presented with triplets of deidentified feedback either generated by human experts, LLMs, or a mix of both, and instructed to identify the most different one based on the content focus of the substantive changes recommended, rather than the tone, mechanics or style of the comments.

H.3 Expert recruitment for annotation and labeling of the de-identified artifacts

We recruited expert ELA and math educators for labeling the similarity data through a graduate school of education alumni mailing list and snowball sampling based on teacher networks affiliated with the school of education. The recruited and compensated experts annotated and labeled only deidentified classroom transcript data. Annotators were compensated at an hourly rate of $50 for their expertise. Our instructions explained that the collected annotations would be used for research purposes. Demographics of the expert labelers for the student reasoning data are: 4 White or Caucasian female, 2 Asian male, 1 Black or African American male, 1 White or Caucasian male, 1 Multi-racial female, and 1 Multi-racial prefer-not-to-say. Demographics of the expert labelers for the feedback data are: 4 White or Caucasian female, 1 White or Caucasian male, 1 Hispanic or Latine female, and 1 Black or African American female. One ELA educator has 5-6 years of teaching experience and the rest have 8 or more years of experience. They are all based in the U.S.

H.4 Data accessibility

We share the full annotations for the math reasoning data which we collected under an IRB for human subject research, and a subset of the feedback collected by Mah et al. (2025); Tan et al. (2024a) who consented to sharing a subset of their feedback data to help the reviewers understand the similarity evaluation tasks. The data files are organized as: `data` which includes the expert and LLM annotations, `metadata` which includes metadata about the math lesson or the student's writing assignment, `tasker2task mapping` which contains a mapping of the expert IDs to the task IDs, and `task2annotation mapping`, which contains a mapping of the task IDs used during the similarity evaluations to the annotation texts. `similarity judgments` include the labels from the *pick-the-odd-one* evaluations linked to each tasker ID. Aggregated similarity scores are in

[4] These transcripts were collected from middle schools in a large urban public school district in California, under an IRB for human subject research #75294. The teachers, students, and student guardians consented to their participation.

human-eval scores.

H.5 Web-based task interfaces

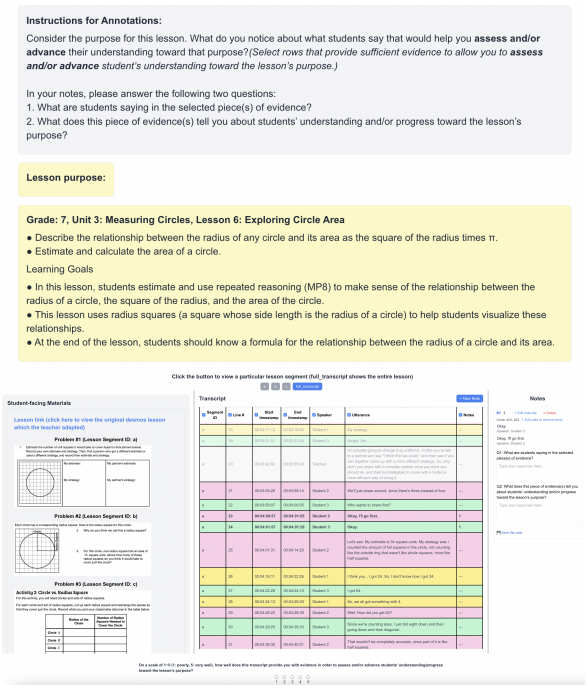

Figure 7: **Task interface for assessing and interpreting student's mathematical understanding from classroom transcripts.** We collected annotations from expert math educators using a web-based interface. Participants were presented with a classroom transcript, metadata about the lesson's purpose, learning goals, and math activities. Their task was to identify specific lines of student talk that signaled meaningful moments of mathematical reasoning and answer two questions regarding the observed student's understanding and progress towards the learning goals.

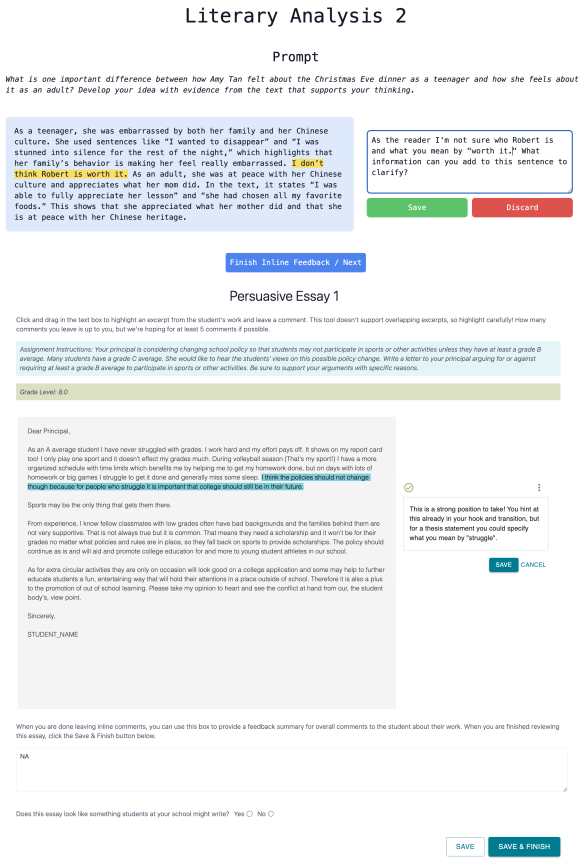

Figure 8: **Task interface for writing feedback about student's essay by Mah et al. (2025); Tan et al. (2024a).**

H.6 Expert instruction for the *odd-one-out* task

LLM-as-a-judge system message started with either "You are a helpful high school English Language Arts teaching assistant" or "You are a helpful math teacher assistant." The relative judgment task was given in the system prompt and the triplet samples were given in the user message. We added "Do not include any explanation in your answer" to the system message to enable easier model output processing.

Math expert evaluation instruction

Teachers were asked to make notes about what students said in the transcript that would help them assess and/or advance students' understanding towards the lesson purpose. These are the notes that three different teachers made about "what students are saying" in the pieces of transcript. Select which note is most different in terms of the underlying ideas (rather than focusing on the style, tone, or length of the notes).

ELA expert evaluation instruction

{Essay instruction / writing prompt given to the student.} Three teachers asked to give the student feedback for revising their writing. Select which feedback is most different in terms of what it asks the student to change in their writing. Focus on the feedback content, rather than judging based on the delivery (e.g., teacher's writing style, tone, length, use of questions versus directives in the feedback).

I Hyperparameters for topic distribution modeling.

We used BERTopic package for topic modeling, UMAP for dimension reduction, and HDBSCAN for clustering. The parameters for UMAP are: n neighbors = 3, n components = 100, min distance = 0, metric = cosine. The parameters for HDBSCAN are min samples = 1, metric = euclidean. We set the min cluster size of HDBSCAN as a tunable hyperparameter and enumerated between $\{2, 3, .., 15\}$ and reported the best result from hyperparameter search.

We used an open-sourced Python wrapper for MALLET implementation of LDA by maria-antoniak/little-mallet-wrapper. For both the math and the feedback domains, we enumerated the number of topics to find from $\{50, 75, 100, 125, 150\}$ and reported the best result. With the math data, LDA topic modeling could not find more than 125 unique topics.

Our experiments do not rely on GPU resources.