

# Hierarchical Analytic Writing Feedback with Fine-Tuned Transformers

Michael P. Hemenway and Justin O. Barber and Martha Bellows and Susan M. Lottridge  
Pearson

{michael.hemenway, justin.barber, martha.bellows, sue.lottridge}@pearson.com

## Abstract

We describe a production system that uses frontier LLMs to bootstrap deployable, deterministic scoring models — using LLMs for *dataset enhancement* rather than production scoring. This division of labor resolves the cost, latency, and psychometric validity challenges that make direct LLM scoring unsuitable for high-volume formative assessment: the generation cost is amortized across the deployment lifetime, stochasticity is resolved at labeling time, and the resulting fine-tuned models are fully owned, reproducible artifacts.

Applied to Grade 5 opinion essays in a single-prompt case study, the system scores 5 competency dimensions and 31 fine-grained binary feedback codes. Fine-tuned transformers achieve strong performance from remarkably small labeled datasets: competency models trained on just  $n=400$  essays achieve QWK of 0.65–0.74, and component models trained on as few as 300–500 examples achieve  $\kappa$  up to 0.92, with several matching inter-rater agreement. The system was deployed in May 2026 and forms the foundation for broader release across grades and genres.

## 1 Introduction

Automated writing feedback systems have the potential to dramatically expand access to formative assessment in K-12 education (Taghipour and Ng, 2016; Misgna et al., 2024). Unlike summative scoring, formative feedback must be *actionable*: students and teachers need to know not just that a response scored poorly, but *which* specific writing skills contributed to that score and where in the text those skills are (or are not) demonstrated. This requires simultaneously modeling dozens of fine-grained traits from a single essay — a multi-task prediction problem with heterogeneous label quality, severe class imbalance on many traits, and production constraints that rule out direct LLM inference.

NAVY Writing is a production system within the Pearson ecosystem that delivers hierarchical analytic writing feedback for K-12 students. For the May 2026 pilot, the system targets Grade 5 opinion writing across five competency dimensions and 31 binary feedback codes, serving approximately 50 teacher participants. We present this as a *single-prompt case study*: all results derive from one Grade 5 opinion prompt, one administration year, and  $n = 500$  labeled responses. Findings establish proof-of-concept but should not be interpreted as generalizable claims across grades, genres, or prompts without further validation. Throughout this paper we use *competency* to refer to the five standards-aligned holistic scores (W.5.1, W.5.4, L.5.1, L.5.2, L.5.3) and *component* to refer to the fine-grained binary feedback codes nested within each competency. This two-level hierarchy — 5 holistic scores with 31 analytic components beneath them — distinguishes the system from typical analytic rubrics with 2–6 traits at a single level of granularity.

This paper addresses three practical challenges that arise when building hierarchical analytic writing feedback systems at scale:

**Small-sample modeling.** Obtaining reliable human labels for 31 binary codes and 5 ordinal competency scores requires substantial investment in rubric development, scorer training, and double-scoring with adjudication. Rather than treating small labeled datasets as a limitation to overcome before modeling can begin, we demonstrate that fine-tuned transformers achieve strong performance from remarkably modest annotation budgets: competency models trained on  $n=400$  essays and component models trained on as few as 300–500 examples routinely approach or match human inter-rater agreement.

**LLMs for data enhancement, not scoring.** A natural question is why not score directly with a

frontier LLM? The target deployment scale (~1M responses/year across 36 traits) makes direct LLM inference impractical due to cost, latency, and — critically — psychometric validity concerns: stochastic outputs from generative models cannot be validated against the reproducibility standards required for educational measurement. We instead use LLMs *upstream* to generate silver-labeled training data, extending coverage to rule-amenable constructs without additional human annotation. This positions the LLM as a *dataset enhancement* tool whose cost is amortized across the full deployment lifetime.

**Independent granular and holistic models.** Because each component and competency is modeled independently, predictions can conflict. We describe the production post-processing that reconciles these predictions.

**Contributions.** Specifically, we describe:

1. Empirical evidence that fine-tuned transformers achieve  $\kappa$  up to 0.92 from training sets of 300–500 labeled essays (§6).
2. A hybrid gold/silver labeling strategy in which a multi-agent LLM pipeline (Claude Sonnet 4.6 via AWS Bedrock) generates training data, enabling coverage of 31 binary feedback codes without proportional human annotation cost (§3).
3. A production architecture with a competency override mechanism that reconciles component-level and competency-level predictions (§5).

A companion paper (Barber et al., 2026) describes the multi-agent LLM annotation pipeline in further detail.

## 2 Related Work

Automated essay scoring (AES) has progressed from feature-engineered systems (Attali and Burstein, 2006) through neural approaches (Taghipour and Ng, 2016; Rodriguez et al., 2019; Yang et al., 2020) to transformer-based models fine-tuned on rubric-aligned labels (Crossley et al., 2024). The PERSUADE 2.0 corpus (Crossley et al., 2024) provides a large-scale benchmark for argumentative writing assessment, though it targets secondary students and holistic scoring rather than hierarchical formative feedback.

Recent work has explored LLMs as direct annotators (Ding et al., 2023; He et al., 2024; Tan et al., 2024), with mixed results on subjective tasks. Our approach follows this line but uses LLM outputs as *training data* for smaller, deployable models rather than as production-time scorers — a distillation strategy that preserves the LLM’s labeling capacity while meeting production latency and cost constraints.

The Longformer architecture (Beltagy et al., 2020) provides linear-complexity attention suited to essay-length documents (1024–2048 tokens). While more recent efficient transformers exist (e.g., ModernBERT (Warner et al., 2025)), Longformer remains competitive in our evaluations and benefits from extensive operational validation.

Prior work on analytic essay scoring (Persing and Ng, 2016) typically addresses 4–6 traits at a single level of granularity. Our system operates on a two-level hierarchical structure — 5 ordinal competency scores and 31 binary feedback codes nested within them — with the explicit goal of actionable formative feedback rather than summative evaluation alone.

## 3 Training Data and Label Quality

### 3.1 Assessment Context and Data

The pilot targets a Grade 5 opinion writing prompt. Five hundred responses were randomly sampled; the same set was used across all scored competencies and components.

### 3.2 Label Acquisition Strategy

The system employs two labeling approaches, chosen per-component based on construct complexity:

**Gold-standard labels (human-scored).** Five competency scores and 13 component codes received full double-scoring by trained raters ( $n = 500$ ). These include all ordinal competency scores and components requiring holistic qualitative judgment (e.g., coherence, logical reasoning, appropriate style). The scoring protocol used a three-point ordinal scale (0–2) for competencies rather than direct binary scoring, as binary scales were found to produce unreliable inter-rater agreement on complex constructs. Range-finding sessions with content experts, 30 anchor papers, and back-reading quality control ensured calibrated labels.

**Silver-standard labels (LLM-generated).** The remaining 16 component codes were scored by a

multi-agent LLM pipeline (Claude Sonnet 4.6 via AWS Bedrock), validated against a small human-scored benchmark ( $n = 50$ , single expert rater). The pipeline employs three specialized agents: (1) an *annotator* that scores each component using rubric-grounded chain-of-thought prompts, (2) a *backreader* that independently re-scores a stratified subset, and (3) an *adjudicator* that resolves disagreements between the first two agents. This design reduces LLM stochasticity at labeling time rather than at inference time. Target components are amenable to rule-based evaluation (e.g., detecting run-on sentences, checking capitalization, identifying paragraph structure). A companion paper (Barber et al., 2026) describes the pipeline architecture in full detail; here we report only the downstream model performance on human-scored test data. A limitation is that the  $n = 50$  benchmark uses a single expert rater, precluding inter-rater agreement computation for silver-labeled codes.

### 3.3 Label Characteristics

Table 1 reports inter-rater agreement for gold-standard labels, which validate label quality and establish performance ceilings for the trained models.

| Competency / Component                              | QWK/ $\kappa$ | SMD   | EA (%) |
|-----------------------------------------------------|---------------|-------|--------|
| <i>Ordinal competency scores (QWK)</i>              |               |       |        |
| W.5.1                                               | 0.81          | -0.02 | 78.4   |
| W.5.4                                               | 0.75          | -0.01 | 73.6   |
| L.5.1                                               | 0.76          | 0.00  | 74.6   |
| L.5.2                                               | 0.74          | -0.01 | 76.8   |
| L.5.3                                               | 0.78          | -0.05 | 79.8   |
| <i>Binary component codes (<math>\kappa</math>)</i> |               |       |        |
| W51-C4 Support Purpose                              | 0.75          | 0.01  | 88.2   |
| W51-C5 Logical Reasons                              | 0.75          | 0.01  | 88.8   |
| W51-C6 Facts & Details                              | 0.76          | -0.03 | 91.8   |
| W51-C10 Conclusion                                  | 0.66          | 0.09  | 84.8   |
| W54-C2 Coherence                                    | 0.76          | 0.02  | 88.4   |
| W54-C3 Development                                  | 0.83          | 0.04  | 94.2   |
| W54-C4 Organization                                 | 0.80          | 0.02  | 91.2   |
| W54-C6 Approp. Style                                | 0.61          | 0.06  | 82.0   |
| W54-C7 Purpose of Writing                           | 0.90          | 0.04  | 97.4   |
| L51-C5 Var. Times                                   | 0.61          | -0.03 | 94.0   |
| L51-C7 Var. States                                  | 0.74          | 0.00  | 93.2   |
| L51-C8 Var. Conditions                              | 0.68          | -0.07 | 88.4   |
| L53-C2 Sent. Structure                              | 0.66          | 0.03  | 84.8   |
| L53-C6 Formality                                    | 0.59          | -0.09 | 79.8   |

Table 1: Inter-rater agreement for gold-standard labels ( $N = 500$ ). QWK for ordinal scores; Cohen’s  $\kappa$  for binary codes. Four zero-variance codes excluded (L51-C6 Sequences, L53-C3 Dialects, L53-C4 Registers, L53-C5 Var. of English — all responses scored 0).

Key observations relevant to modeling:

- **Class imbalance is pervasive.** Positive-class prevalence ranges from 8.8% (L51-C5 Var. Times) to 84.6% (W54-C7 Purpose), with 6 of 13 gold-labeled codes above 65% prevalence.
- **Agreement ceilings vary widely.** Inter-rater  $\kappa$  spans 0.59 (Formality) to 0.90 (Purpose of Writing). Models cannot be expected to exceed these ceilings when training labels are drawn from the same raters.
- **Zero-variance codes exist.** Four codes showed no variation (100% scored 0) — the prompt did not elicit those skills. These were excluded from modeling.

### 3.4 Data Splits

The 500 gold-labeled responses were split using stratified random sampling on the Cartesian product of the five competency scores:

- **Held-out test set (20%,  $n \approx 100$ ):** Final evaluation only. Never exposed to training.
- **Training pool (80%,  $n \approx 400$ ):** Split 2:1 into training (67%) and validation (33%) for model selection.

## 4 Modeling Approach

For the five competency scores and the component feedback codes with fully human-scored responses, models were trained exclusively on hand-labeled data using a staged hyperparameter search grounded in Pearson’s operational scoring experience.

### 4.1 Staged Search Protocol

Training proceeds through two escalating stages:

1. **Stage 1 — Longformer-base, 8 configurations.** Each scoring target is first trained across 8 hyperparameter combinations derived from configurations with established strong performance in prior operational deployments. If the best trial meets acceptable metric thresholds, training stops here.
2. **Stage 2 — Longformer-base, 48 configurations.** Scoring targets that did not meet threshold in Stage 1 are escalated to a broader search over the same encoder backbone.

This staged approach avoids exhaustive search on targets that converge quickly (many competency models resolve at Stage 1).

## 4.2 Hyperparameter Search Space

Table 2 shows the search space for the Stage 2 Longformer-base recipe as a representative example.

ModernBERT (Warner et al., 2025) was evaluated as an alternative backbone for a subset of scoring targets but did not outperform Longformer in any case and was not selected for deployment.

## 4.3 Model Selection and Deployment

The model with the highest validation-set performance across all trials and stages was selected for deployment. The HPO objective, early-stopping criterion, and checkpoint-selection criterion are all the same metric: QWK for ordinal scores, Cohen’s  $\kappa$  for binary codes. Final test metrics used the held-out partition ( $n = 91$ ) never exposed to training or model selection.

## 4.4 Deployment Outcomes

In total, 5 competency models and 14 component models were trained on gold-standard human labels; an additional 26 component models were trained on silver-standard LLM-generated labels. All five competency models have been deployed. Of the 40 component models evaluated, 31 were selected for deployment; the remaining 9 were suppressed due to insufficient performance or removal from the operational rubric.

## 5 Score Reconciliation

Because competency and component models are trained independently and deliver predictions on different scales, post-processing is required to produce coherent student-facing feedback.

### 5.1 Competency Score Scaling

Competency models produce ordinal 0–2 predictions. A post-processor maps to binary delivery: score 2 maps to 1 (competent), scores 0–1 map to 0 (not yet competent). This high threshold reflects a deliberate content decision — in a formative context, labeling a student “competent” at score 1 risks signaling no further improvement is needed. As Table 4 shows, this transformation reduces agreement for both humans and models, an expected consequence that the override mechanism (§5.2) partially mitigates.

### 5.2 Competency Override Logic

Because competency and component models are trained independently, a student can receive

“strength” on all evaluated components while the competency model predicts “not competent.” To resolve this, a post-processor defines a subset of component codes per competency as a composite override: if all components in the subset predict “strength,” the competency score is overridden to “competent.” For example, strength on W54-C1, C2, C3, and C4 triggers a W.5.4 override. This reconciles local and global predictions without requiring a joint model.

## 6 Results

### 6.1 Evaluation Protocol

**Primary metrics.** Quadratic Weighted Kappa (QWK) (Cohen, 1968) for ordinal competency scores; Cohen’s  $\kappa$  (Cohen, 1960) for binary component codes. Both are chance-corrected agreement measures appropriate for imbalanced label distributions.

**Supplementary metrics.** Exact agreement (EA), Standardized Mean Difference (SMD), positive-class precision, and macro F1 are reported to provide a complete picture of model behavior. No single threshold on any one metric serves as an acceptance gate; metrics are evaluated jointly.

**Test set.** All results use the held-out partition ( $n = 91$ ) with final human-adjudicated scores, never exposed to training or model selection. For silver-labeled components, a subset ( $n = 50$ ) received single expert scores for validation.

### 6.2 Competency Scores (Ordinal 0–2)

Table 3 reports model performance alongside inter-rater agreement on the same test set. QWK ranges from 0.645 (W.5.4) to 0.740 (L.5.1). Four of five competencies approach or match inter-rater agreement levels. SMD values are small ( $|\text{SMD}| \leq 0.15$ ) for four of five competencies, indicating no systematic scoring bias.

Notably, L.5.2 is the only competency where the model’s QWK numerically exceeds inter-rater QWK (0.720 vs. 0.714). More broadly, four of five competencies achieve QWK within 0.03–0.07 of the inter-rater ceiling, indicating that model agreement closely matches the agreement achievable between trained human raters.

**Binary competency delivery.** The operational system maps ordinal predictions to binary (competent/not-competent) via integer truncation at the midpoint (score 2  $\rightarrow$  1; scores 0–1  $\rightarrow$  0).

| Hyperparameter          | Values / Range                                       | Type     |
|-------------------------|------------------------------------------------------|----------|
| Model artifact          | longformer-base-regression:1.0.0                     | Fixed    |
| Max sequence length     | {1024, 2048}                                         | Discrete |
| Learning rate           | $\{1.01, 1.51, 2.01, 2.99\} \times 10^{-5}$          | Discrete |
| Weight decay            | {0.000, 0.011, 0.021, 0.031}                         | Discrete |
| Warmup ratio            | {0.06, 0.08, 0.10, 0.12}                             | Discrete |
| Epochs                  | {3, 4, 5, 6}                                         | Discrete |
| LR scheduler            | {C, L, CW}                                           | Discrete |
| Balance weight          | {0.1, 0.5, 0.6, 0.7, 0.8}                            | Discrete |
| Train batch size        | {4, 6, 8} (seq 1024); {2, 4} (seq 2048) <sup>a</sup> | Discrete |
| Predict batch size      | {1, 2} (seq 1024); {1} (seq 2048) <sup>a</sup>       | Discrete |
| torch.compile (train)   | {True, False}                                        | Discrete |
| torch.compile (predict) | False                                                | Fixed    |
| Gradient checkpointing  | False                                                | Fixed    |

Table 2: Hyperparameter search space for Longformer-base (48 random trials). Scheduler codes: C = cosine, L = linear, CW = constant\_with\_warmup. <sup>a</sup>Batch sizes are sequence-length-dependent.

| Comp. | $n$ | QWK <sub>H1H2</sub> | QWK <sub>model</sub> | EA <sub>model</sub> | SMD <sub>model</sub> |
|-------|-----|---------------------|----------------------|---------------------|----------------------|
| W.5.1 | 91  | 0.766               | 0.720                | 0.736               | 0.144                |
| W.5.4 | 91  | 0.717               | 0.645                | 0.604               | -0.029               |
| L.5.1 | 91  | 0.752               | 0.740                | 0.736               | 0.031                |
| L.5.2 | 91  | 0.714               | 0.720                | 0.769               | -0.051               |
| L.5.3 | 91  | 0.823               | 0.698                | 0.747               | -0.154               |

Table 3: Competency score results ( $n = 91$ , ordinal 0–2). QWK<sub>H1H2</sub> = inter-rater agreement on same test set; QWK<sub>model</sub> = model vs. human final score; EA<sub>model</sub> = exact agreement; SMD<sub>model</sub> = model – human standardized mean difference.

Table 4 shows that this transformation substantially reduces agreement for both humans and models — an expected consequence of collapsing a three-point scale to binary, particularly when the “developing” category (score 1) contains 35–50% of responses.

| Comp. | $n$ | $\kappa_{H1H2}$ | $\kappa_{model}$ | EA    | $p_+$ |
|-------|-----|-----------------|------------------|-------|-------|
| W.5.1 | 91  | 0.535           | 0.607            | 0.802 | 0.451 |
| W.5.4 | 91  | 0.488           | 0.403            | 0.791 | 0.209 |
| L.5.1 | 91  | 0.592           | 0.468            | 0.824 | 0.198 |
| L.5.2 | 91  | 0.470           | 0.412            | 0.879 | 0.143 |
| L.5.3 | 91  | 0.696           | 0.556            | 0.912 | 0.143 |

Table 4: Competency scores after binary transformation ( $n = 91$ ).  $p_+$  = human positive-class prevalence (proportion rated competent). Low prevalence on L.5.2 and L.5.3 inflates kappa volatility.

### 6.3 Component Feedback Code Performance

Table 5 reports performance for all deployed component models evaluated against the human-scored held-out test set. Gold-labeled (G) models were trained on double-scored human labels ( $n = 91$ ); silver-labeled (S) models were trained on LLM-generated annotations and validated against expert scores (mostly  $n = 50$ ).

**Summary statistics.** Among the 28 deployed components with computable  $\kappa$  (3 of 31 deployed models did not have computable  $\kappa$ ): 10 achieve  $\kappa \geq 0.70$  and 17 achieve  $\kappa \geq 0.50$ . Three gold-labeled models numerically match or exceed their respective inter-rater  $\kappa$ : W54-C2 (0.843 vs. 0.76), W54-C4 (0.804 vs. 0.80), and W51-C5 (0.788 vs. 0.75), though at  $n = 91$  these differences may fall within sampling variability (bootstrap confidence intervals are recommended for future reporting). Among silver-labeled models, L52-C18 (Quotation Marks,  $\kappa = 0.922$ ) demonstrates that rule-amenable constructs can be scored with near-perfect agreement using LLM-generated training data alone.

## 7 Discussion

### 7.1 LLMs as Data Enhancement, Not Scoring

The decision to use LLMs for *training data generation* rather than *production scoring* reflects three categories of concern:

**Cost and latency.** At 1M responses/year and 36 traits per response, direct LLM scoring would require  $\sim 36$ M inference calls annually — exceeding fine-tuned model compute by 2–3 orders of magnitude. Fine-tuned Longformer models produce all

| Code                                   | Description              | Lbl | $n$ | $\kappa_{\text{H1H2}}$ | $\kappa$ | EA    | F1 <sub>+</sub> | Prec <sub>+</sub> | Rec <sub>+</sub> | $p_+$ |
|----------------------------------------|--------------------------|-----|-----|------------------------|----------|-------|-----------------|-------------------|------------------|-------|
| <i>W.5.1 — Opinion Writing</i>         |                          |     |     |                        |          |       |                 |                   |                  |       |
| W51-C1                                 | Intro Topic              | S   | 50  | —                      | 0.569    | 0.800 | 0.844           | 0.900             | 0.794            | 0.600 |
| W51-C2                                 | State Opinion            | S   | 50  | —                      | 0.623    | 0.920 | 0.955           | 0.933             | 0.977            | 0.900 |
| W51-C3                                 | Org Structure            | S   | 50  | —                      | 0.827    | 0.920 | 0.938           | 0.909             | 0.968            | 0.660 |
| W51-C4                                 | Support Purpose          | G   | 91  | 0.75                   | 0.751    | 0.879 | 0.897           | 0.857             | 0.941            | 0.615 |
| W51-C5                                 | Logical Reasons          | G   | 91  | 0.75                   | 0.788    | 0.901 | 0.922           | 0.883             | 0.964            | 0.659 |
| W51-C6                                 | Facts/Details            | G   | 91  | 0.76                   | 0.656    | 0.890 | 0.932           | 0.971             | 0.895            | 0.769 |
| W51-C7                                 | Link Opinion             | S   | 50  | —                      | 0.330    | 0.740 | 0.831           | 0.711             | 1.000            | 0.900 |
| W51-C8                                 | Conclusion               | S   | 50  | —                      | 0.545    | 0.760 | 0.769           | 0.625             | 1.000            | 0.640 |
| W51-C9                                 | Paragraphs               | S   | 50  | —                      | 0.786    | 0.900 | 0.921           | 0.879             | 0.967            | 0.660 |
| W51-C10                                | Logical Conclusion       | G   | 91  | 0.66                   | 0.413    | 0.747 | 0.596           | 0.630             | 0.567            | 0.297 |
| W51-C11                                | Avoid Repetition         | S   | 50  | —                      | 0.369    | 0.740 | 0.817           | 0.806             | 0.829            | 0.720 |
| <i>W.5.4 — Task, Purpose, Audience</i> |                          |     |     |                        |          |       |                 |                   |                  |       |
| W54-C1                                 | Clear                    | S   | 50  | —                      | 0.781    | 0.900 | 0.923           | 0.882             | 0.968            | 0.680 |
| W54-C2                                 | Coherent                 | G   | 91  | 0.76                   | 0.843    | 0.923 | 0.933           | 0.961             | 0.907            | 0.560 |
| W54-C3                                 | Approp. Dev.             | G   | 91  | 0.83                   | 0.494    | 0.846 | 0.905           | 0.893             | 0.918            | 0.824 |
| W54-C4                                 | Approp. Org.             | G   | 91  | 0.80                   | 0.804    | 0.912 | 0.933           | 0.933             | 0.933            | 0.659 |
| W54-C5                                 | On Task                  | S   | 50  | —                      | 0.408    | 0.900 | 0.945           | 0.896             | 1.000            | 0.960 |
| W54-C6                                 | Approp. Style            | G   | 91  | 0.61                   | 0.328    | 0.692 | 0.562           | 0.514             | 0.621            | 0.385 |
| <i>L.5.1 — Grammar and Usage</i>       |                          |     |     |                        |          |       |                 |                   |                  |       |
| L51-C1                                 | Conjunctions             | S   | 50  | —                      | 0.393    | 0.740 | 0.822           | 0.698             | 1.000            | 0.860 |
| L51-C9                                 | Verb Tense               | S   | 50  | —                      | 0.432    | 0.880 | 0.932           | 0.932             | 0.932            | 0.880 |
| L51-C11                                | Sentence Variety         | S   | 50  | —                      | 0.594    | 0.800 | 0.833           | 0.735             | 0.962            | 0.680 |
| L51-C12                                | Run-ons                  | S   | 50  | —                      | 0.532    | 0.760 | 0.760           | 0.905             | 0.655            | 0.420 |
| L51-C13                                | Fragments                | S   | 50  | —                      | 0.458    | 0.820 | 0.886           | 0.875             | 0.897            | 0.800 |
| L51-C14                                | Grade Level              | S   | 50  | —                      | 0.680    | 0.840 | 0.857           | 0.774             | 0.960            | 0.620 |
| <i>L.5.2 — Conventions</i>             |                          |     |     |                        |          |       |                 |                   |                  |       |
| L52-C15                                | Comma Usage <sup>†</sup> | S   | 50  | —                      | —        | 0.580 | —               | —                 | —                | 0.000 |
| L52-C16                                | End Punctuation          | S   | 50  | —                      | 0.783    | 0.920 | 0.947           | 1.000             | 0.900            | 0.720 |
| L52-C17                                | Upper/Lower Case         | S   | 50  | —                      | 0.711    | 0.860 | 0.881           | 0.867             | 0.897            | 0.600 |
| L52-C18                                | Quotation Marks          | S   | 50  | —                      | 0.922    | 0.980 | 0.933           | 0.875             | 1.000            | 0.160 |
| L52-C19                                | Grade Level <sup>‡</sup> | S   | —   | —                      | —        | —     | —               | —                 | —                | —     |
| <i>L.5.3 — Grade-Level Style</i>       |                          |     |     |                        |          |       |                 |                   |                  |       |
| L53-C2                                 | Sent. Structure          | G   | 91  | 0.66                   | 0.351    | 0.725 | 0.528           | 0.700             | 0.424            | 0.220 |
| L53-C6                                 | Formality <sup>a</sup>   | S   | 91  | 0.59 <sup>b</sup>      | 0.168    | 0.582 | 0.568           | 0.532             | 0.610            | 0.516 |
| L53-C7                                 | Grade Level <sup>‡</sup> | S   | —   | —                      | —        | —     | —               | —                 | —                | —     |

Table 5: Component feedback model performance on the held-out test set. **Lbl:** G = gold-standard (human double-scored,  $n = 91$ ); S = silver-standard (LLM-annotated,  $n = 50$  unless noted).  $\kappa_{\text{H1H2}}$  = inter-rater agreement from Table 1 (shown for gold-labeled codes to contextualize model performance; — indicates no human double-scoring available). <sup>a</sup> $n = 91$ . <sup>b</sup>Gold inter-rater  $\kappa$  available for L53-C6 despite silver-labeled model training. <sup>†</sup>Zero positive-class instances in validation sample;  $\kappa$ , F1<sub>+</sub>, Prec<sub>+</sub>, and Rec<sub>+</sub> undefined. <sup>‡</sup>Deployed but not validated against human scores; see Ethics Statement.  $\kappa$  = Cohen’s kappa; EA = exact agreement; F1<sub>+</sub> = positive-class F1; Prec<sub>+</sub> = positive-class precision; Rec<sub>+</sub> = positive-class recall;  $p_+$  = positive-class prevalence.

36 scores in <2 seconds on a single GPU vs. 10–30 seconds for sequential LLM inference.

**Psychometric validity.** Educational measurement requires *reproducibility*: the same response must receive the same score across administrations. Fine-tuned models produce deterministic outputs, enabling the regression testing and fairness auditing that operational assessment demands.

**The upstream alternative.** Using LLMs for dataset enhancement resolves these concerns: the generation cost is amortized across the deployment lifetime, stochasticity is resolved through multi-agent adjudication at labeling time, and the resulting fine-tuned models are fully owned, deterministic artifacts.

## 7.2 Small-Sample Effectiveness

The gold-labeled competency models are trained on just  $n \approx 400$  essays, yet four of five approach or match inter-rater QWK. Component models trained on 300–500 examples achieve  $\kappa \geq 0.70$  for 10 of 28 evaluated codes — including distributed, essay-level phenomena (coherence, organization, logical reasoning) that require integrating signal across the full document.

That several models numerically match or exceed inter-rater agreement point estimates is noteworthy, though the small test set ( $n = 91$ ) means these differences are not statistically robust. The pattern is plausibly explained by training on adjudicated labels that resolve rater disagreement, producing a more consistent scoring function than either individual rater.

Among silver-labeled models, L52-C18 (Quotation Marks,  $\kappa = 0.922$ ) and W51-C3 (Org Structure,  $\kappa = 0.827$ ) demonstrate that for rule-amenable constructs, LLM-generated training data can produce models matching or exceeding what human labeling achieves on more subjective traits.

## 7.3 Where Modeling Struggles: Label Quality as the Binding Constraint

Model performance is bounded by label quality and construct definition, not model capacity. Three patterns illustrate this:

**Subjective constructs with low inter-rater agreement.** Appropriate Style (W54-C6,  $\kappa = 0.33$ ; inter-rater 0.61) and Formality (L53-C6,  $\kappa = 0.17$ ; inter-rater 0.59) demonstrate that when raters cannot reliably agree, no model can exceed the signal

quality in the labels.

### Localized phenomena in full-essay models.

Codes requiring detection of a specific feature in a narrow essay region exhibit low signal-to-noise ratios. Logical Conclusion (W51-C10,  $\kappa = 0.41$ ) and Link Opinion (W51-C7,  $\kappa = 0.33$ ) exhibit this pattern; sentence-level decomposition (Barber et al., 2026) provides mitigation.

**Prompt design as the binding constraint.** Four codes showed zero variance across all 500 responses — the prompt did not elicit those skills. For practitioners, this suggests that prompt/task design is the binding constraint for trait coverage.

## 7.4 The Binary Transformation Problem

Collapsing the ordinal 0–2 competency scale to binary delivery (Table 4) degrades agreement for *both* humans and models — an information-theoretic consequence of forcing the “developing” category (score 1, 35–50% of responses) into the “not competent” bin. The override mechanism (§5.2) partially mitigates this.

## 7.5 The Independent Modeling Problem

Each component and competency is modeled independently — a deliberate production choice that enables progressive deployment and isolates failure modes, but prevents low-resource components from borrowing statistical strength from related traits. The override mechanism (§5.2) addresses prediction conflicts; multi-task learning across correlated traits is planned for Phase 2.

## 7.6 Limitations

**Evaluation sample size.** At  $n = 91$ , a single misclassification can shift  $\kappa$  by 0.08–0.12 for low-prevalence codes. Bootstrap confidence intervals are needed for production monitoring.

**Silver-label validation.** The 16 silver-labeled components were validated against a single expert rater on  $n = 50$  essays, precluding performance ceiling estimates. Double-scoring a subset is planned for Phase 2.

**Unvalidated deployed models.** Two component models (L52-C19, L53-C7) are deployed without human validation; one (L52-C15) has undefined  $\kappa$  due to zero positive-class instances. All are flagged for priority validation.

**Single grade and genre.** All results derive from a single Grade 5 opinion prompt. Transfer to other grades, genres, and prompts requires fresh validation.

## 8 Conclusion

Three findings emerge from this single-prompt case study. First, fine-tuned transformers achieve agreement closely matching inter-rater ceilings ( $\kappa$  up to 0.92, QWK up to 0.74) from training sets of 300–500 labeled essays. Second, using LLMs for dataset enhancement rather than production scoring resolves the cost, latency, and validity tensions inherent in generative scoring. Third, the binding constraints on trait coverage are label quality and prompt design, not model capacity.

Phase 2 will extend coverage to additional grades and genres, explore joint training across correlated traits, and incorporate span-level feedback highlights.

## Ethics Statement

This system produces scores and feedback that real students and teachers see. Model performance is validated against human-scored benchmarks, and known limitations are documented explicitly. Automated scores supplement, not replace, teacher judgment; the system is deployed in a formative context where scores inform instruction but do not determine grades or advancement.

Two component models (L52-C19, L53-C7) are deployed without human validation, and one (L52-C15) has undefined  $\kappa$ . These are flagged for priority validation. Fairness analysis across demographic subgroups will be conducted when enrollment data becomes available.

## Acknowledgments

The authors thank the NAVVY ELA content team and the hand scoring team for their contributions to rubric development, range finding, and scorer training. We also thank the Pearson TDS and infrastructure teams for support in model deployment.

## References

Yigal Attali and Jill Burstein. 2006. [Automated essay scoring with e-rater® v.2](#). *The Journal of Technology, Learning and Assessment*, 4(3).

Justin O. Barber, Michael P. Hemenway, Martha Bellows, and Susan M. Lottridge. 2026. Multi-agent

LLM annotation and scoring for training fine-grained K-12 writing-feedback models. In *Proceedings of the AIME-Con*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. [Longformer: The long-document transformer](#). *arXiv preprint arXiv:2004.05150*.

Jacob Cohen. 1960. [A coefficient of agreement for nominal scales](#). *Educational and Psychological Measurement*, 20(1):37–46.

Jacob Cohen. 1968. [Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit](#). *Psychological Bulletin*, 70(4):213–220.

S.A. Crossley, Y. Tian, P. Baffour, A. Franklin, M. Benner, and U. Boser. 2024. [A large-scale corpus for assessing written argumentation: PERSUADE 2.0](#). *Assessing Writing*, 61:100865.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. [Is GPT-3 a good data annotator?](#) In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada. Association for Computational Linguistics.

Xingwei He, Zhenghao Lin, Yeyun Gong, A-Long Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2024. [AnnoLLM: Making large language models to be better crowdsourced annotators](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 165–190, Mexico City, Mexico. Association for Computational Linguistics.

Haile Misgna, Byung-Won On, Ingyu Lee, and Gyu Sang Choi. 2024. [A survey on deep learning-based automated essay scoring and feedback generation](#). *Artificial Intelligence Review*, 58(2):36.

Isaac Persing and Vincent Ng. 2016. [End-to-end argumentation mining in student essays](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1384–1394, San Diego, California. Association for Computational Linguistics.

Pedro Uria Rodriguez, Amir Jafari, and Christopher M. Ormerod. 2019. [Language models and automated essay scoring](#). *arXiv preprint arXiv:1909.09482*.

Kaveh Taghipour and Hwee Tou Ng. 2016. [A neural approach to automated essay scoring](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas. Association for Computational Linguistics.

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansoor Karami, Jundong Li, Lu Cheng, and Huan Liu.

2024. [Large language models for data annotation and synthesis: A survey](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 930–957, Miami, Florida, USA. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. [Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547, Vienna, Austria. Association for Computational Linguistics.

Ruosong Yang, Jiannong Cao, Zhiyuan Wen, Youzheng Wu, and Xiaodong He. 2020. [Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1560–1569, Online. Association for Computational Linguistics.