ALIGNED TEXTUAL SCORING RULES

Anonymous authors
Paper under double-blind review

ABSTRACT

A reference-based text scoring system scores the quality of a reported text by comparing it to a ground truth text. Existing scoring systems, including BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), BERTScore (Zhang et al., 2020), and LLM judges (Zheng et al., 2023), can be susceptible to strategic manipulations that increase scores without improving the textual information. Conversely, scoring systems designed for strategic robustness (Wu & Hartline, 2024; Lu et al., 2024; Xu et al., 2025) need not align well with target scores assigned by human or LLM judges (Xu et al., 2026). We develop a framework for learning scoring systems that are robust to strategic manipulations. Our approach obtains a numerical representation of reported and ground-truth texts and learns a proper scoring rule over the numerical representation. Our main assumption is that text representations encode linear properties of expressed uncertainty over an underlying semantic space. Under this assumption, truthful reporting maximizes the expected score of the proper scoring rule, leading to the strategic robustness of the scoring system. When the assumption holds approximately, we bound the strategic gain by a multiple of the squared representation error. Our experiment evaluates both alignment and strategic robustness of our learned scores. Experiment results show that our learned scores improves the alignment with a given target score from a judge, and are robust to strategic manipulations. These findings support learning within proper-scoring-rule families as an approach to aligning textual scoring systems with judge scores, while retaining strategic robustness.

1 INTRODUCTION

Scoring a reported text by comparing it with a ground truth text is central to applications such as peer grading, question answering, and language model training. In peer grading, for example, students are asked to grade each other's homework submissions. An instructor may compare a peer review against the instructor's ground truth review of the same submission and score the peer review. A scoring system can be implemented by human judges or scalable automatic scoring systems. Automatic scoring systems include lexical-overlap-based scoring systems such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004), embedding-based scoring systems such as BERTScore (Zhang et al., 2020), and LLM judges. Prior work has shown LLM judges align well with human judgements of text quality (Zheng et al., 2023; Fu et al., 2024). These automatic scoring systems enable scalable text evaluation.

However, these scoring systems can be susceptible to strategic manipulations. Prior work on reward hacking and bias in scoring systems has shown such manipulations can increase scores without improving the underlying information (Amodei et al., 2016; Krakovna et al., 2020). Recent work on the evaluation of scoring systems proposes to test both the alignment and the strategic robustness of a scoring system (Xu et al., 2026). The test framework integrates several existing strategic manipulations as robustness tests, including rephrasing to probe sensitivity to presentation (Chen et al., 2024), meaningless elongation to probe length bias (Singhal et al., 2023), positive and negative opinion shifts, and hedged and strong opinion shifts to probe sensitivity to expressed confidence (Zhou et al., 2024; Leng et al., 2025).

The design of strategically robust scoring systems draws on the information elicitation literature from game theory. The game-theoretic approach models the source of the reported text as a strategic reporter, whose goal is to maximize expected score. With access to a ground truth text for scoring, the information elicitation literature formalizes this strategic robustness as proper scoring rules (McCarthy, 1956; Savage, 1971; Gneiting & Raftery, 2007). A scoring rule is called *proper* if truthful

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

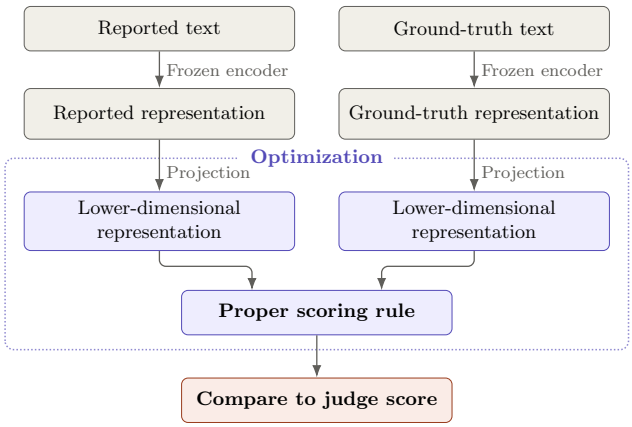


Figure 1: Learning pipeline for textual scoring. A shared frozen encoder maps the reported text and ground-truth text to numerical representations. A shared projection reduces their dimension, and a proper scoring rule produces a score for comparison with the judge score. The dotted box encloses the projection and scoring rule, whose parameters are jointly optimized to align with the judge score.

reporting of information maximizes the expected score. Thus, the strategic reporter has no incentive to manipulate the report. With access to a ground truth text, Wu & Hartline (2024) apply this to scoring systems for text by breaking texts down to semantic summary points, and applying numerical proper scoring rules. Without a ground truth text, Lu et al. (2024); Xu et al. (2025); Robertson & Koyejo (2025) construct strategically robust scoring systems, based on the literature of peer prediction mechanism design (Miller et al., 2005; Shnayder et al., 2016; Kong & Schoenebeck, 2018). These scoring systems produce explicit incentive guarantees, instead of unconstrained evaluation.

These approaches prioritize strategic robustness but are shown not to align well with target scores, such as from human judges or LLM judges (Xu et al., 2026). Generally, incentive guarantees can be orthogonal to the quality a judge cares about. Motivated by this limitation, we ask the following question.

Can we design strategically robust scoring systems that align with a given judge score?

We solve this question by optimizing over the space of proper scoring rules for alignment with a target score. The target score is generated by a human judge or an LLM judge comparing a reported text with a ground truth text. Such an aligned scoring rule thus serves as a strategically robust scoring proxy for judges. Our approach, detailed in Section 3, has three steps, illustrated in Figure 1. First, we separately obtain numerical representations of a reported text and a ground-truth text using a frozen encoder. Second, the high-dimensional representations are linearly projected to a lower-dimensional space. Finally, we apply a proper scoring rule over the lower-dimensional representations. Both the projection and the proper scoring rules are learned to minimize the mean squared error (MSE) from the target score.

To apply a proper scoring rule, our main assumption is that the representation is a vector of *linear properties* of the expressed belief over semantic outcomes in text (Section 3.1). We assume there exists a numerical space of semantic meaning in text. We treat the representation of a ground-truth text as a fixed linear transformation of this semantic space. For a reported text that expresses semantic uncertainty, we assume its representation equals the average of the corresponding ground-truth representations, under the expressed belief. For example, in a peer review, a text saying "I'm not sure" should lie on the linear interpolation between a text of "This homework submission is correct" and "This homework is wrong". Linear projection preserves these linear properties. Prior work has studied the linear representation of semantic features in language models (Park et al., 2024; Jiang et al., 2024; Garg et al., 2026). Our assumption is specialized to uncertainty expressed in text, requiring representations to encode expectations of semantic features.

When the assumption is perfect, the strategic robustness of the scoring system follows directly from the properness of the scoring rule we use. We use scoring rules that are proper for a reported linear property. For such a scoring rule, reporting the expected value of the corresponding outcome maximizes the expected score. We optimize over two hypothesis classes, both of which are proper scoring rules for linear properties: sums of Brier scores and sums of Log-Cosh scores over the projected representations (Section 3.2).

We relax the linearity assumption by showing the scoring system is approximately proper when the assumption is imperfect. When the linearity assumption holds only approximately, we bound the possible gain from strategic reporting in terms of representation error. For both the Brier and log-cosh

hypothesis, we show that the strategic gain can be bounded by a constant multiple of the squared error of the projected representation (Section 3.3). We also empirically test the approximate properness of the learned Log-Cosh scoring rule under imperfect linearity. We use texts that express numeric or verbal uncertainty between two semantic outcomes. Our experiment shows that the imperfect linearity assumption only leads to a small gain by strategic manipulation (Section E.5).

Our experiments follow the alignment and strategic-robustness test framework of Xu et al. (2026) on real and simulated peer grading and two question-answering datasets (Section 4). We compare our Brier and Log-Cosh scoring rules with ROUGE-L, BLEU, BERTScore, raw LLM judges, the strategically robust TVD-FVar-Statement baseline from Xu et al. (2026), and a simple linear regression trained with the same pipeline in our paper. For alignment, we report Pearson and Spearman correlations with the target score. Log-Cosh exceeds TVD-FVar-Statement on both correlations for Peer Grading and LFQA-E; MedAESQA is the exception, with lower correlations except for Haiku-target Pearson and overlapping reported confidence intervals. For strategic robustness, we adopt the six manipulation strategies from Xu et al. (2026): rephrasing, meaningless elongation, positive and negative opinion shifts, and hedged and strong opinion shifts. Brier, Log-Cosh, and TVD-FVar-Statement exhibit no statistically significant score increases under their evaluated manipulations, whereas ROUGE-L, BLEU, BERTScore, raw LLM judges, and linear regression are susceptible to strategic manipulations (Table 2). Notably, the contrast between linear regression's 23/40 significant increases and Log-Cosh's 0/40 supports the contribution of proper-scoring-rule structure to robustness beyond the shared representation and regularization (Section F.2).

1.1 Related Work

Our work connects textual elicitation (Wu & Hartline, 2024; Lu et al., 2024; Kimpara et al., 2023), statistical and strategic evaluation of scoring systems (Xu et al., 2026), and optimization of proper scoring rules (Li et al., 2022; Hartline et al., 2023; Chen et al., 2023). We learn judge-relevant linear properties while retaining proper-score structure. Section A discusses these connections and automated mechanism design in detail.

2 Preliminaries

We introduce proper scoring rules for linear properties and connect them to textual scoring with ground truth texts.

Notation. Bold symbols denote column vectors, uppercase letters denote matrices, and $\Delta(\mathcal{A})$ is the set of distributions on $\mathcal{A}$. Full notation and further scoring details appear in Section B.

2.1 Proper Scoring Rules for a Scalar Linear Property

A scoring rule scores a probabilistic report by comparing it to the realized outcome. It is *proper* if reporting one's true belief distribution maximizes expected score (Gneiting & Raftery, 2007). More generally, a rule can elicit a property of the belief rather than the entire distribution (Abernethy & Frongillo, 2012). We first consider a scalar linear property; the later hypotheses combine several such properties.

Let $\boldsymbol{z} \in \mathcal{Z} \subseteq \mathbb{R}^d$ be an outcome, fix a direction $\boldsymbol{v} \in \mathbb{R}^d$, and let $q \in \Delta(\mathcal{Z})$ be a belief. The realized *linear statistic* and its expected value, the *linear property*, are

$$\theta(\boldsymbol{z}) = \boldsymbol{v}^\top \boldsymbol{z} \in \mathbb{R}, \qquad \mu_{q,\boldsymbol{v}} = \mathbf{E}_{\boldsymbol{z}\sim q}\left[\theta(\boldsymbol{z})\right] = \boldsymbol{v}^\top \mathbf{E}_{\boldsymbol{z}\sim q}\left[\boldsymbol{z}\right] \in \mathbb{R}. \tag{1}$$

It is linear in the outcome distribution. We will focus on scoring rules for eliciting linear properties. A *report*, or *reported linear property*, is a scalar $r \in \mathbb{R}$; it is truthful when $r = \mu_{q,\boldsymbol{v}}$. A scoring rule $S : \mathbb{R} \times \mathcal{Z} \to \mathbb{R}$ takes the report and outcome as inputs, and gives a real-valued score.

Definition 2.1 (Properness for a Scalar Linear Property)**.** A scoring rule S is *ε-approximately proper* for the linear property defined by $\boldsymbol{v}$ if, for every belief q with finite relevant expectations and every report $r \in \mathbb{R}$,

$$\mathbf{E}_{\boldsymbol{z}\sim q}\left[S(\mu_{q,\boldsymbol{v}}, \boldsymbol{z})\right] \geq \mathbf{E}_{\boldsymbol{z}\sim q}\left[S(r, \boldsymbol{z})\right] - \varepsilon. \tag{2}$$

where $\varepsilon \geq 0$. The rule is *proper* when this holds with $\varepsilon = 0$.

We use the following two proper scoring rules for eliciting a linear property.

Table 1: Notation table for the textual scoring pipeline in Section 3.

Notation	Math Domain	Purpose	Implementation
$\boldsymbol{z}(y)$	$\mathbb{R}^d$	Semantic outcome	Assumed; not observed or implemented
W	$\mathbb{R}^{d\times p}$	Semantic-to-representation map	Assumed; not observed or implemented
$\boldsymbol{h}(t)$	$\mathbb{R}^p$	text representation	From a frozen encoder
A	$\mathbb{R}^{p\times K}$	Linear projection of $\boldsymbol{h}$	Learned

Brier (quadratic) score. The Brier score for the scalar linear property is

$$S_{\text{quad}}(r, \boldsymbol{z}) = -(r - \theta(\boldsymbol{z}))^2. \tag{3}$$

When $\theta(\boldsymbol{z}) \in \{0, 1\}$ and r is its reported probability, this is the negative binary Brier loss. We write scores with high scores being good, in accordance with the literature on scoring rules. For a general linear statistic, it elicits $\mu_{q,\boldsymbol{v}}$ under finite second moments.

Log-Cosh score. For a fixed threshold $b \in \mathbb{R}$, define

$$S_{\text{lc},b}(r, \boldsymbol{z}) = \log\cosh(r - b) + \tanh(r - b)(\theta(\boldsymbol{z}) - r). \tag{4}$$

The basic rule has $b = 0$; any fixed b elicits the same linear property.

2.2 Textual Scoring Problem

Problem Statement. We seek a scoring rule over text that aligns with a given target score while preserving truthful-reporting incentives. In our dataset, each sample consists of a reported text x_i, a ground truth text $y_i \in \mathcal{Y}$, and a target score s_i. The target score can be generated from human judges, LLM judges, or a known synthetic rule. Given n such samples, our objective is to find a scoring rule that minimizes squared error against the target scores:

$$\min_{S\in\mathcal{S}} \sum_{i=1}^{n} \big(S(x_i, y_i) - s_i\big)^2, \tag{5}$$

where $\mathcal{S}$ is a hypothesis class of textual scoring rules.

3 Learning Textual Scoring Rules

The main difficulty in applying a scoring rule to text is constructing a shared numerical space for the reported text and ground truth text. We construct the scoring rule in three steps.

1. Text representation: we obtain a shared numerical representation $\boldsymbol{h}$ from LLM hidden states or text embeddings. Our main assumption is that $\boldsymbol{h}$ is a vector of linear properties of the text's expressed belief over the semantic space.
2. Dimension reduction: we linearly project $\boldsymbol{h}$ to a lower-dimensional space. Every such projection is again a vector of linear properties.
3. Scoring: we apply a proper scoring rule to the projected representations of the reported and ground truth texts.

We jointly learn the projection matrix in step 2 and the scoring-rule from a hypothesis class in step 3 to minimize squared error against the target score. The two hypothesis classes use sums of scalar Brier scores or positively weighted scalar Log-Cosh scores.

Section 3.1 states our assumption that text representations encode linear properties. Section 3.2 presents the hypothesis space and learning problem. Section 3.3 analyzes approximate properness when the representation assumption holds only approximately.

3.1 Main Assumption: Text Representations as Linear Properties

For a reported text x and ground truth text y, let $\boldsymbol{h}(x), \boldsymbol{h}(y) \in \mathbb{R}^p$ be their shared numerical representations. Table 1 summarizes the main pipeline quantities. We assume that each coordinate of $\boldsymbol{h}$ is a linear property, in the sense of Equation (1), of the expressed belief over semantic outcomes.

Each ground truth text y determines a semantic vector $\boldsymbol{z}(y) \in \mathcal{Z} \subseteq \mathbb{R}^d$. In peer grading, its dimensions may encode criteria such as clarity, statement correctness, and proof correctness. A reported text x expresses a belief $Q_x \in \Delta(\mathcal{Z})$ over the semantic outcome. The report is truthful when this expressed belief matches the reporter's posterior given their information. We associate a ground truth text y with the certainty belief $Q_y = \delta_{\boldsymbol{z}(y)}$. We allow $\boldsymbol{z}$ to include a constant coordinate so that fixed representation offsets, including preprocessing centering, can be included in the direction matrix below.

For example, let $z_1 \in \{0, 1\}$ indicate whether a proof is correct. A report expressing a 75% probability of correctness has $\mathbf{E}_{\boldsymbol{z}\sim Q_x}[z_1] = 0.75$. We assume that the text representation encodes fixed linear combinations of such expected semantic features.

Assumption 3.1 (Linear-Property Representation)**.** There exists a fixed direction matrix $W = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_p] \in \mathbb{R}^{d\times p}$ such that, for every reported text or ground truth text t with associated belief Q_t of finite first moment,

$$\boldsymbol{h}(t) = \mathbf{E}_{\boldsymbol{z}\sim Q_t}\left[W^\top \boldsymbol{z}\right] = W^\top \mathbf{E}_{\boldsymbol{z}\sim Q_t}[\boldsymbol{z}]\,. \tag{6}$$

Thus $h_j(t) = \mu_{Q_t,\boldsymbol{w}_j}$: the representation is a vector of linear properties over the semantic space. For a ground truth text, $\boldsymbol{h}(y) = W^\top \boldsymbol{z}(y)$ is the corresponding realized statistic. Linearity is with respect to the expressed belief, not the text string. If reports x_1 and x_2 express beliefs Q_1 and Q_2, a report x_λ expressing $\lambda Q_1 + (1-\lambda)Q_2$, for $\lambda \in [0, 1]$, satisfies

$$\boldsymbol{h}(x_\lambda) = \lambda \boldsymbol{h}(x_1) + (1-\lambda)\boldsymbol{h}(x_2). \tag{7}$$

The matrix W is implicit in the fixed text representation and is not estimated. The key closure property is that linear maps commute with expectation. Under Assumption 3.1,

$$A^\top \boldsymbol{h}(t) = A^\top W^\top \mathbf{E}_{\boldsymbol{z}\sim Q_t}[\boldsymbol{z}] = \mathbf{E}_{\boldsymbol{z}\sim Q_t}\left[(WA)^\top \boldsymbol{z}\right]. \tag{8}$$

Thus, linear projection preserves the interpretation of each coordinate as an expectation. Under this assumption, the scoring rules depend on the expressed belief Q_x only through its semantic mean $\boldsymbol{m}_x = \mathbf{E}_{\boldsymbol{z}\sim Q_x}[\boldsymbol{z}]$, and ultimately through the learned projection $A^\top W^\top \boldsymbol{m}_x$. They therefore elicit these projected means rather than the full belief distribution.

Assumption 3.1 is a special case of the linear representation hypothesis studied in prior work. Prior work has formulated and theoretically analyzed linear representations of semantic features (Park et al., 2024; Jiang et al., 2024; Garg et al., 2026), alongside empirical validations (Li et al., 2021; Gurnee & Tegmark, 2024; Kim et al., 2025). We specialize this hypothesis by allowing all task-relevant dimensions of semantic meaning to be expressed with uncertainty: their feature values are expectations of semantic outcomes under the belief conveyed by the text, as in Equation (6). Text representations may also encode nonsemantic presentation features, such as the style in which a report is written, that do not themselves express uncertainty about the semantic outcome. Our learned projection aims to select the semantic meaning dimensions while discarding such nonsemantic directions. A training regularizer encourages this selection by penalizing differences between representations of the reported and ground truth texts, as detailed in Section E.

Imperfect Assumption Need not Affect Properness We relax Assumption 3.1 theoretically and empirically. Section 3.3 derives guarantees relating representation error to reporting incentives; Section E.5 measures the resulting gaps and bounds for our learned Log-Cosh scoring rule. Despite imperfect linearity, mean incentive gaps are modest in five of six conditions (0.0614–0.0750 in the evaluated score's units). Thus, imperfect linearity need not imply a large loss of properness.

3.2 Learning Problem

From judge-scored pairs (x_i, y_i, s_i), we jointly learn the linear projection and scoring-rule parameters by minimizing squared prediction loss. The same frozen encoder produces the representations $\boldsymbol{h}$ for both text roles. Let K be the number of scalar linear properties and $A = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_K] \in \mathbb{R}^{p\times K}$ the learned dimension-reduction matrix. Define

$$\boldsymbol{r}_A(x) = A^\top \boldsymbol{h}(x), \qquad \boldsymbol{\theta}_A(y) = A^\top \boldsymbol{h}(y). \tag{9}$$

By Assumption 3.1 and Equation (8), each $r_{A,k}(x)$ is a reported linear property and $\theta_{A,k}(y)$ is its realized outcome statistic.

We introduce the two hypothesis classes here. We calculate the score's mean squared error (MSE) from the target score and select the minimizer from each class.

Brier (quadratic) Family. We sum the scalar Brier scores in Equation (3) across the K projected coordinates and add a learned intercept $\beta \in \mathbb{R}$:

$$\widehat{s}^{\text{Brier}}_{A,\beta}(x, y) = \ell_{A,\beta}(x, y) = \beta - \|\boldsymbol{r}_A(x) - \boldsymbol{\theta}_A(y)\|_2^2. \tag{10}$$

The parameters are the projection matrix A and the intercept β.

Log-Cosh Family. We sum the scalar Log-Cosh scores in Equation (4) across the K projected coordinates, with learned positive weights $\alpha_k > 0$ and thresholds $b_k \in \mathbb{R}$, and add a term $\beta + \boldsymbol{\gamma}^\top \boldsymbol{h}(y)$ that depends only on the ground truth text:

$$\begin{aligned} S_{\text{lc}}(x, y) = \beta + \boldsymbol{\gamma}^\top \boldsymbol{h}(y) + \sum_{k=1}^{K} \alpha_k \log\cosh(r_{A,k}(x) - b_k) \\ + \sum_{k=1}^{K} \alpha_k \tanh(r_{A,k}(x) - b_k)(\theta_{A,k}(y) - r_{A,k}(x)). \end{aligned} \tag{11}$$

The hypothesis outputs $\widehat{s}^{\text{Log-Cosh}}_{A,\boldsymbol{\alpha},\boldsymbol{b},\beta,\boldsymbol{\gamma}}(x, y) = S_{\text{lc}}(x, y)$. The parameters are the projection matrix A, the positive weights $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$ over scalar log-cosh scores, the thresholds $\boldsymbol{b} = (b_1, \ldots, b_K)$, the intercept $\beta \in \mathbb{R}$, and the ground truth text coefficient $\boldsymbol{\gamma} \in \mathbb{R}^p$.

3.3 Approximate Properness under Representation Error

We now relax the assumption that a truthful text's representation equals the conditional mean of the ground truth representations. We call the discrepancy *representation error* and analyze its effect on truthful-reporting incentives. Here $\boldsymbol{h}$ may be nonlinear and need not satisfy Assumption 3.1; the projection $A^\top$ remains linear. The error measures failure to preserve the relevant linearity assumption. Fix $\boldsymbol{h}$, A, and all scoring parameters before reporting. For $f \in \{\text{Brier}, \text{lc}\}$, let $S_{f,A}(\boldsymbol{r}, y)$ denote the rule in Equations (10) and (11) with $\boldsymbol{r}$ substituted for $\boldsymbol{r}_A(x)$, suppressing the other fixed parameters.

The representation error. Let Y denote the random ground truth text and $\mathcal{I}_x$ the information available to the reporter when producing x. This information decides the reporter's belief over Y. For a truthful report, Q_x is the corresponding distribution over semantic outcomes. Define

$$\overline{\boldsymbol{h}}_x = \mathbf{E}[\boldsymbol{h}(Y) \mid \mathcal{I}_x], \qquad \boldsymbol{\delta}_h(x) = \boldsymbol{h}(x) - \overline{\boldsymbol{h}}_x. \tag{12}$$

This error is defined without Assumption 3.1. When the assumption holds and x is truthful, $\boldsymbol{\delta}_h(x) = \mathbf{0}$. For a truthful report x_λ expressing probabilities $1 - \lambda$ and λ over ground truth texts y_0, y_1, the discrepancy is directly computable:

$$\boldsymbol{\delta}_h(x_\lambda) = \boldsymbol{h}(x_\lambda) - \big[(1 - \lambda)\boldsymbol{h}(y_0) + \lambda \boldsymbol{h}(y_1)\big]. \tag{13}$$

This is the controlled setting evaluated in Section E.5. For either scoring family, define the posterior-mean target and reporting error in its learned scoring space by

$$\boldsymbol{\mu}_{A,x} = \mathbf{E}[\boldsymbol{\theta}_A(Y) \mid \mathcal{I}_x], \qquad \boldsymbol{\delta}_A(x) = \boldsymbol{r}_A(x) - \boldsymbol{\mu}_{A,x} = A^\top \boldsymbol{\delta}_h(x). \tag{14}$$

Its k-th coordinate is $\delta_{A,k}(x) = \boldsymbol{a}_k^\top \boldsymbol{\delta}_h(x)$, where $\boldsymbol{a}_k$ is the k-th column of A. Log-Cosh thresholds cancel from this error because the same shift applies to the reported coordinate and its posterior mean.

Theorem 3.2 (Approximate properness under representation error)**.** *Fix a truthful text x and either scoring family $f \in \{\text{Brier}, \text{lc}\}$. Suppose the relevant conditional expectations are finite. The numerical report $\boldsymbol{\mu}_{A,x}$ maximizes conditional expected score. Consequently, for every alternative report $\widetilde{\boldsymbol{r}} \in \mathbb{R}^K$,*

$$\mathbf{E}[S_{f,A}(\boldsymbol{r}_A(x), Y) \mid \mathcal{I}_x] \geq \mathbf{E}[S_{f,A}(\widetilde{\boldsymbol{r}}, Y) \mid \mathcal{I}_x] - \varepsilon_f, \tag{15}$$

where the exact maximum gain over the truthful representation is

$$\varepsilon_f = \mathbf{E}[S_{f,A}(\boldsymbol{\mu}_{A,x}, Y) - S_{f,A}(\boldsymbol{r}_A(x), Y) \mid \mathcal{I}_x]. \tag{16}$$

This gain is nonnegative and is attained by $\widetilde{\boldsymbol{r}} = \boldsymbol{\mu}_{A,x}$.

Every alternative text induces a numerical report, so the same error bounds the gain from textual deviations, even when no text realizes the optimal numerical report. Terms depending only on the ground truth text cancel from these comparisons.

Corollary 3.3 (Log-Cosh approximate properness)**.** *Under the conditions of Theorem 3.2, the Log-Cosh rule in Equation (11) has approximation error bounded by*

$$\varepsilon_{\mathrm{lc}} \le \frac{1}{2}\boldsymbol{\delta}_A(x)^\top \Lambda \boldsymbol{\delta}_A(x) = \frac{1}{2}\sum_{k=1}^{K} \alpha_k \delta_{A,k}(x)^2 \le \frac{L}{2}\|\boldsymbol{\delta}_h(x)\|_2^2, \tag{17}$$

where $\Lambda = \mathrm{diag}(\alpha_1, \ldots, \alpha_K)$ *and* $L = \lambda_{\max}(A\Lambda A^\top)$*. The diagonal matrix* Λ *collects the positive scoring weights, and* $\lambda_{\max}$ *denotes the largest eigenvalue.*

Corollary 3.4 (Brier approximate properness)**.** *Fix a truthful text* x *and suppose* $\mathbf{E}[\|\boldsymbol{\theta}_A(Y)\|_2^2 \mid \mathcal{I}_x] < \infty$*. The Brier rule satisfies Equation (15) with the exact approximation error*

$$\varepsilon_{\mathrm{Brier}} = \|\boldsymbol{\delta}_A(x)\|_2^2 = \|A^\top \boldsymbol{\delta}_h(x)\|_2^2. \tag{18}$$

Both corollaries use the same representation error, with different bounds for the two scoring rules. The Brier gap is exactly its squared Euclidean norm, whereas the Log-Cosh gap is bounded by its amplitude-weighted squared norm. Errors in the null space of $A^\top$ do not affect either guarantee. In particular, $\boldsymbol{\delta}_A(x) = \mathbf{0}$ recovers exact properness, even if $\boldsymbol{\delta}_h(x) \neq \mathbf{0}$. Proofs appear in Section D.

4 Experiments

We ask two empirical questions. **Alignment:** How well do the two scoring-rule families reproduce human, synthetic, and LLM target scores? **Strategic robustness:** Are learned scoring rules robust to strategic manipulations?

We compare the Brier (quadratic) and Log-Cosh scoring rules. Both use MSE training, four-seed ensembles, and clean-validation selection, but differ in score parameterization, regularizers, representations, and capacity grids, as described below. The main text summarizes test alignment and manipulation counts; validation correlations, every manipulation effect, selected configurations, and complete 95% intervals appear in Section E.

4.1 Datasets and Target Scores

We evaluate on four datasets spanning peer grading and question answering. Each example pairs a reported text (candidate) with a ground truth text and a scalar target score.

Peer Grading pairs a student's peer review with an instructor review of the same algorithms submission. Its targets are a scalar human annotation and independent 1–5 ratings from Claude Haiku 4.5 and GPT-5-mini. Simulated Peer Grading uses controlled reviews that encode three binary attributes: answer correctness, proof correctness, and writing quality. Its target is the fraction of these attributes that match the ground truth text.

MedAESQA pairs generated answers to consumer-medical questions with expert-curated ground truth texts (Gupta et al., 2025). For LFQA-E, we compare long-form candidate answers with ground truth texts in a grouped English subset of the dataset (Fan et al., 2026). These two datasets do not provide a scalar human target suited to our regression setup, so their targets are ratings from the same two LLM judges.

We group the training, validation, and test splits by submission, simulated ground truth text instance, medical question, or LFQA-E question–ground truth text group, so that no reference group appears in more than one split. Table 3 in the appendix reports the split sizes; Section E gives the full protocol.

4.2 Evaluation Protocol

Alignment metrics. We measure alignment on clean test data using Pearson and Spearman correlations between predicted scores and target scores. Pearson measures linear association between score values, while Spearman measures agreement between their rankings. Higher positive correlations indicate better alignment. We report both metrics separately for each target and summarize them with target-macro averages that weight the eight targets equally. Section E.3 gives the confidence-interval procedures and baseline evaluation details.

Strategic manipulations. Following Xu et al. (2026), we evaluate six strategies: rephrasing (R) changes presentation; meaningless elongation (M) adds uninformative filler; positive and negative opinion shifts (P/N) make evaluations more favorable or unfavorable; and hedged and strong opinion

shifts (H/S) weaken or intensify claims. Peer Grading and Simulated Peer Grading use all six strategies; MedAESQA and LFQA-E use H/M/R/S. Claude Haiku 4.5 rewrites only test reported texts, keeping ground truth texts fixed; all locally evaluated scoring rules use the same cached rewrites. We measure score changes with standardized effects and count a manipulation effect as significantly positive when its 95% confidence interval lies entirely above zero. For effects evaluated on our splits, intervals use 50,000 bootstrap resamples of whole reference groups, preserving clean/manipulated pairs. The retained Brier results use a different effect normalization from Log-Cosh, so their effect magnitudes are not directly comparable. Section E.4 describes the strategies and effect estimation in detail.

4.3 Implementation

Brier (quadratic). The retained Brier pipeline uses frozen Llama-3.1-8B-Instruct, concatenates mean-pooled layers 12–15, and fits shared standardization and PCA on training data. It fits the Brier scoring rule in Equation (10) with squared error and a training-distribution penalty of coefficient $\rho = 0.5$. Each configuration averages seeds 0–3, retaining the minimum clean-validation-MSE checkpoint per seed; minimum ensemble validation MSE selects capacity. The complete grid, objective, and optimizer settings are in Section E.

Log-Cosh. We fit a separate Log-Cosh scoring rule to each frozen Qwen3.5-27B (Qwen Team, 2026) layer 28/32/36/40/44/48. Qwen interleaves three linear-attention blocks with one full-attention block, motivating this regular full-attention sequence. Each layer uses train-fitted standardization, group-balanced mean-gap decomposition, and unwhitened PCA. The role coordinate has penalty 20, and the linear term in the ground truth text representation has unit ridge. Unit axis directions are normalized separately from positive scales and amplitudes. We average four fixed seeds per layer, then six scalar layer scores.

For Log-Cosh, the grid is $p \in \{16, 32, 64, 128, 256, 384, 512\}$ and $K \in \{4, 8, 16, 32, 64\}$. Checkpoints minimize clean validation MSE plus the role penalty. Among capacities within half a group-bootstrap standard error of minimum ensemble validation MSE, we choose the fewest parameters, breaking size ties by validation MSE (the half-SE rule). These clean-validation criteria select the scoring rule used for both evaluations below. Section E gives details.

4.4 Alignment and Strategic Robustness

Alignment. Log-Cosh improves alignment over the strategically robust TVD-FVar-Statement baseline on Peer Grading and LFQA-E, while MedAESQA gives mixed results. We report Pearson and Spearman correlations with the target score in Table 2. Log-Cosh achieves target-macro Pearson/Spearman $.629/.606$ and $.933/.929$ on Simulated Peer Grading. On Peer Grading, both correlations exceed TVD-FVar-Statement for all three targets; for example, alignment with the human target improves from $.408/.468$ to $.618/.548$. Both correlations also improve for the two LFQA-E targets. On MedAESQA, Log-Cosh has higher Haiku-target Pearson ($.669$ versus $.643$), but lower Spearman ($.618$ versus $.645$); for the GPT target, both correlations are lower ($.623/.546$ versus $.659/.663$), with overlapping reported confidence intervals. Log-Cosh also exceeds ROUGE-L, BLEU, and BERTScore on both correlations for every target. These comparisons use source-reported TVD-FVar-Statement results on different evaluation subsets, and the Brier and Log-Cosh pipelines differ in representations, preprocessing, regularization, capacity selection, and QA judge rubrics. Complete alignment estimates and intervals appear in Table 5.

Strategic robustness. Brier, Log-Cosh, and TVD-FVar-Statement exhibit no statistically significant score increases under the evaluated manipulations, whereas the other baselines remain susceptible. We evaluate the scoring systems selected for alignment on clean validation data, without using manipulated texts for model selection. Across the six manipulation strategies adopted from Xu et al. (2026), Brier and Log-Cosh each have 0/40 significantly positive effects; the source-reported TVD-FVar-Statement results have 0/14 on the overlapping datasets. In contrast, raw LLM judges have 16/28 significantly positive effects, and ROUGE-L, BLEU, and BERTScore have 3/20, 4/20, and 4/20, respectively. Notably, linear regression has 23/40 despite using the Log-Cosh representations, matched regularization, and the same model-selection procedure. This contrast supports the contribution of proper-scoring-rule structure to robustness beyond the shared representation and regularization, although the controls do not isolate properness from all differences in expressiveness (Section F.2).

Table 2: Baseline comparison. (a) Test Pearson / Spearman; bold and underline mark the highest and second-highest reported point estimates for each metric and target, not statistical significance. TVD denotes TVD-FVar-Statement† from Xu et al. (2026), on different evaluation subsets; "–" is unavailable. Brier is the retained pipeline with different representations and QA rubrics; LinReg uses the Log-Cosh representations and matched regularization (Section F.2). (b) Positive 95% manipulation intervals / evaluated effects; denominators differ by available targets and attacks. Raw-judge scores define the LLM targets; independent judge-to-human alignment results are not reported. Full estimates and intervals are in Tables 4 to 7 and 15.

(a) Test alignment: Pearson / Spearman

Dataset	Target	ROUGE-L	BLEU	BERTScore	TVD†	Brier	LinReg	Log-Cosh
Peer	Human	.215 / .229	.307 / .290	.096 / .063	.408 / .468	.536 / .452	**.640 / .622**	.618 / .548
Peer	Haiku 4.5	.274 / .373	.370 / .490	.293 / .335	.564 / .568	.559 / .602	.650 / **.686**	**.651** / .674
Peer	GPT-5-mini	.310 / .358	.396 / .462	.210 / .221	.626 / .619	.663 / .700	**.819 / .826**	.649 / .665
Simulated	Outcome score	.108 / .072	.137 / .122	.192 / .177	–	.731 / .767	-.035 / -.055	**.933 / .929**
MedAESQA	Haiku 4.5	.381 / .318	.414 / .450	.453 / .421	.643 / **.645**	.444 / .389	.524 / .411	**.669** / .618
MedAESQA	GPT-5-mini	.413 / .320	.404 / .420	.425 / .357	.659 / .663	.596 / .529	**.755 / .697**	.623 / .546
LFQA-E	Haiku 4.5	.164 / .174	.098 / .108	.176 / .186	.351 / .326	.204 / .219	**.557 / .530**	.425 / .415
LFQA-E	GPT-5-mini	.100 / .079	.082 / .017	.058 / .061	.367 / .342	.147 / .106	**.550 / .537**	.467 / .449

(b) Manipulation: positive-interval counts

	ROUGE-L	BLEU	BERTScore	TVD†	Brier	LinReg	Log-Cosh	Raw Haiku 4.5	Raw GPT-5-mini
Sig. ↑	3/20	4/20	4/20	0/14	0/40	23/40	0/40	7/14	9/14

The zero counts indicate no statistically detectable average gains under these tests. Complete manipulation effects and intervals appear in Tables 6 and 7; further ablations in Section F examine how the scoring form, role regularization, and capacity selection affect the alignment–robustness tradeoff.

5 Conclusion, Limitations, and Future Work

We learn textual scoring systems that align with judge scores while retaining the structure of proper scoring rules. Under a linear representation assumption, truthful reporting maximizes expected score; when the assumption is imperfect, we bound the possible strategic gain by squared projected representation error. Experiments on peer grading and question answering demonstrate that the learned scoring systems combine alignment with resistance to the tested manipulations. Both Brier and Log-Cosh exhibit no statistically significant score increases under these tests, while matched linear regression and raw LLM judges remain susceptible. These results support proper scoring rules as a basis for designing judge-aligned, strategically robust scoring systems.

We identify adaptive manipulations as an important direction for future work. Our empirical evaluation follows the test framework of Xu et al. (2026), which spans a representative set of strategic manipulations identified in the reward-hacking literature. These tests use fixed manipulation strategies and therefore do not establish robustness against attacks tailored to a particular scoring system. A more general test framework could adaptively generate and refine manipulations in response to the system's scores, searching for profitable deviations beyond the predefined strategies. Such evaluation would strengthen the empirical assessment of strategic robustness.

Another direction for future work is to improve the strategic robustness of scoring systems by improving the linearity of their representations with respect to expressed beliefs. Prior work identifies linear directions in language-model activations that mediate verbal uncertainty and shows that interventions along these directions can control the uncertainty expressed in generated text (Ji et al., 2025). Motivated by these findings, future work could train or calibrate text representations to better preserve mixtures of expressed beliefs, so that an uncertain report's representation more closely matches the belief-weighted average of the corresponding ground truth text representations. This would target the projected representation error that governs our approximate-properness bounds and could thereby reduce the potential gain from strategic reporting (Section 3.3).

AI USE STATEMENT

In this work we used generative AI tools to generate synthetic data sets (synthetic text and reports), formulate mathematical claims, design or provide feedback on research methodology or experiments, and implement methods. In particular, we used AI in coding and running experiments, and to polish and edit the writing, including mathematical writing. We asked AI for feedback to improve our experiments. LLMs are an essential part of the methods presented in this paper. Humans have independently verified the writing and experiments. Additionally, we used generative AI as a search tool for literature review. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The scoring hypotheses, representation assumption, and incentive bounds are specified in Section 3; complete proofs appear in Section D. Section E records grouped splits, preprocessing, training objectives, capacity selection, and confidence-interval procedures. Section F reports the retained ablations and their per-target results. The limits of the evaluation are discussed in Section 4.

REFERENCES

Jacob D. Abernethy and Rafael M. Frongillo. A characterization of scoring rules for linear properties. In *Conference on Learning Theory*, pp. 27–1, 2012.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Maria-Florina Balcan, Avrim Blum, Jason D. Hartline, and Yishay Mansour. Reducing mechanism design to algorithm design via machine learning. *Journal of Computer and System Sciences*, 74(8): 1245–1270, 2008.

Maria-Florina Balcan, Travis Dick, and Ellen Vitercik. Dispersion for data-driven algorithm design, online learning, and private optimization. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science*, pp. 603–614, 2018a.

Maria-Florina Balcan, Tuomas Sandholm, and Ellen Vitercik. A general theory of sample complexity for multi-item profit maximization. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 173–174, 2018b.

Maria-Florina F. Balcan, Tuomas Sandholm, and Ellen Vitercik. Sample complexity of automated mechanism design. *Advances in Neural Information Processing Systems*, 29, 2016.

Noah Burrell and Grant Schoenebeck. Measurement integrity in peer prediction: A peer assessment case study. *arXiv preprint arXiv:2108.05521*, 2021.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. Humans or llms as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8301–8327, 2024.

Siyu Chen, Jibang Wu, Yifan Wu, and Zhuoran Yang. Learning to incentivize information acquisition: Proper scoring rules meet principal-agent model. In *International Conference on Machine Learning*, pp. 5194–5218, 2023.

Yiling Chen and Fang-Yi Yu. Optimal scoring rule design. *arXiv preprint arXiv:2107.07420*, 2021.

Vincent Conitzer and Tuomas Sandholm. Automated mechanism design: Complexity results stemming from the single-agent setting. In *Proceedings of the 5th International Conference on Electronic Commerce*, pp. 17–24, 2003a.

Vincent Conitzer and Tuomas Sandholm. Automated mechanism design for a self-interested designer. In *Proceedings of the 4th ACM Conference on Electronic Commerce*, pp. 232–233, 2003b.

Vincent Conitzer and Tuomas Sandholm. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, pp. 132–141, 2004.

Michael Curry, Tuomas Sandholm, and John Dickerson. Differentiable economics for randomized affine maximizer auctions. In *International Joint Conference on Artificial Intelligence*, 2022.

Michael Curry, Vinzenz Thoma, Darshan Chakrabarti, Stephen McAleer, Christian Kroer, Tuomas Sandholm, Niao He, and Sven Seuken. Automated design of affine maximizer mechanisms in dynamic settings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(9):9626–9635, 2024.

Zhijian Duan, Haoran Sun, Yurong Chen, and Xiaotie Deng. A scalable neural network for DSIC affine maximizer auction design. *Advances in Neural Information Processing Systems*, 36:56169–56185, 2023.

Abhimanyu Dubey et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C. Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning: Advances in differentiable economics. *Journal of the ACM*, 71(1):1–53, 2024.

Yuchen Fan, Chen Ling, Xin Zhong, Shuo Zhang, Heng Zhou, Yuchen Zhang, Mingyu Liang, Chengxing Xie, Ermo Hua, Zhizhou He, Cheng Huang, Gang Chen, Ning Ding, and Bowen Zhou. LFQA-E: Carefully benchmarking long-form QA evaluation. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=bJYm4v0Spr.

Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial auctions via posted prices. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 123–135, 2014.

Jinlan Fu, See Kiong Ng, Zhengbao Jiang, and Pengfei Liu. GPTScore: Evaluate as you desire. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6556–6576, 2024.

Nikhil Garg, Jon Kleinberg, and Kenny Peng. How many features can a language model store under the linear representation hypothesis? In *Proceedings of Thirty Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 5358–5376. PMLR, 2026. URL https://proceedings.mlr.press/v336/garg26a.html.

Tilmann Gneiting. Making and evaluating point forecasts. *Journal of the American Statistical Association*, 106(494):746–762, 2011.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Deepak Gupta, Davis Bartels, and Dina Demner-Fushman. A dataset of medical questions paired with automatically generated answers and evidence-supported references. *Scientific Data*, 12(1):1035, 2025. doi: 10.1038/s41597-025-05233-z. URL https://doi.org/10.1038/s41597-025-05233-z.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jE8xbmvFin.

Jason D. Hartline, Liren Shan, Yingkai Li, and Yifan Wu. Optimal scoring rules for multi-dimensional effort. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 2624–2650. PMLR, 2023.

Justin Hsu, Jamie Morgenstern, Ryan Rogers, Aaron Roth, and Rakesh Vohra. Do prices coordinate markets? In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, pp. 440–453, 2016.

Ziwei Ji, Lei Yu, Yeskendir Koishekenov, Yejin Bang, Anthony Hartshorn, Alan Schelten, Cheng Zhang, Pascale Fung, and Nicola Cancedda. Calibrating verbal uncertainty as a linear feature to reduce hallucinations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3769–3793, 2025.

Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21879–21911. PMLR, 2024. URL https://proceedings.mlr.press/v235/jiang24d.html.

Junsol Kim, James Evans, and Aaron Schein. Linear representations of political perspective emerge in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=rwqShzb9li.

Dhamma Kimpara, Rafael Frongillo, and Bo Waggoner. Proper losses for discrete generative models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17015–17040. PMLR, 2023. URL https://proceedings.mlr.press/v202/kimpara23a.html.

Yuqing Kong and Grant Schoenebeck. Water from two rocks: Maximizing the mutual information. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 177–194, 2018.

Victoria Krakovna, Jonathan Uesato, Vladimir Mikulik, Matthew Rahtz, Tom Everitt, Ramana Kumar, Zac Kenton, Jan Leike, and Shane Legg. Specification gaming: the flip side of ai ingenuity. *DeepMind Blog*, 3:40–53, 2020.

Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. Taming overconfidence in llms: Reward calibration in rlhf. In *International Conference on Learning Representations*, volume 2025, pp. 16484–16517, 2025.

Belinda Z. Li, Maxwell Nye, and Jacob Andreas. Implicit representations of meaning in neural language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1813–1827. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.143. URL https://aclanthology.org/2021.acl-long.143/.

Yingkai Li, Jason D. Hartline, Liren Shan, and Yifan Wu. Optimization of scoring rules. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pp. 988–989, 2022.

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

Yuxuan Lu, Shengwei Xu, Yichi Zhang, Yuqing Kong, and Grant Schoenebeck. Eliciting informative text evaluations with large language models. *The 25th ACM Conference on Economics and Computation*, 2024.

John McCarthy. Measures of the value of information. *Proceedings of the National Academy of Sciences of the United States of America*, 42(9):654, 1956.

Nolan Miller, Paul Resnick, and Richard Zeckhauser. Eliciting informative feedback: The peer-prediction method. *Management Science*, 51(9):1359–1373, 2005.

Jamie Morgenstern and Tim Roughgarden. Learning simple auctions. In *Conference on Learning Theory*, pp. 1298–1318, 2016.

Jamie H. Morgenstern and Tim Roughgarden. On the pseudo-dimension of nearly optimal auctions. In *Advances in Neural Information Processing Systems*, 2015.

Eric Neyman, Georgy Noarov, and S. Matthew Weinberg. Binary scoring rules that incentivize precision. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 718–733, 2021.

Vishakh Padmakumar and He He. Unsupervised extractive summarization using pointwise mutual information. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 2505–2512, 2021.

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

Maneesha Papireddygari and Bo Waggoner. Contracts with information acquisition, via scoring rules. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pp. 703–704, 2022.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39643–39666. PMLR, 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Zachary Robertson and Sanmi Koyejo. Let's measure information step-by-step: LLM-based evaluation beyond vibes. *arXiv preprint arXiv:2508.05469*, 2025. doi: 10.48550/arXiv.2508.05469. URL https://arxiv.org/abs/2508.05469.

Tim Roughgarden and Okke Schrijvers. Ironing in the dark. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 1–18, 2016.

Tuomas W. Sandholm, Vincent Conitzer, and Craig Boutilier. Automated design of multistage mechanisms. In *International Joint Conference on Artificial Intelligence*, 2007.

Leonard J. Savage. Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, 66(336):783–801, 1971.

Johannes Schneider, Robin Richner, and Micha Riser. Towards trustworthy autograding of short, multi-lingual, multi-type answers. *International Journal of Artificial Intelligence in Education*, 33: 88–118, 2023a. doi: 10.1007/s40593-022-00289-z.

Johannes Schneider, Bernd Schenk, Christina Niklaus, and Michaelis Vlachos. Towards LLM-based autograding for short textual answers. *arXiv preprint arXiv:2309.11508*, 2023b.

Weiran Shen, Pingzhong Tang, and Song Zuo. Automated mechanism design via neural networks. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2019.

Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C Parkes. Informed truthfulness in multi-task peer prediction. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 179–196, 2016.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf. *arXiv preprint arXiv:2310.03716*, 2023.

Tonghan Wang, Yanchen Jiang, and David C. Parkes. GemNet: Menu-based, strategy-proof multi-bidder auctions through deep learning. In *Proceedings of the 2024 ACM Conference on Economics and Computation*, 2024.

Yifan Wu and Jason Hartline. ElicitationGPT: Text elicitation mechanisms via language models. *Working paper*, 2024.

Shengwei Xu, Yichi Zhang, Paul Resnick, and Grant Schoenebeck. Spot check equivalence: An interpretable metric for information elicitation mechanisms. *arXiv preprint arXiv:2402.13567*, 2024.

Shengwei Xu, Yuxuan Lu, Grant Schoenebeck, and Yuqing Kong. Benchmarking LLMs' judgments with no gold standard. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uE84MGbKD7.

Shengwei Xu, Yuxuan Lu, Yifan Wu, Jason Hartline, and Grant Schoenebeck. Scoring rules! statistical and strategic alignment for text evaluation metrics. Preprint, 2026.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

Kaitlyn Zhou, Jena Hwang, Xiang Ren, and Maarten Sap. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3623–3643, 2024.

A Additional Related Work

Grading with LLMs Recent work studies the use of LLMs in grading reported texts from student reporters. Schneider et al. (2023a) grade short answers using similarity between numerical representations of reported text and ground truth text, and find that performance varies across languages and question types; the accompanying error analysis shows that semantic similarity alone can be insufficient for answers requiring logical reasoning. Schneider et al. (2023b) prompt a language model to compare reported texts with ground truth texts and find low Pearson correlation with judge scores assigned by instructors. Instead of using embedding similarity or direct prompting as the final scoring system, our approach learns a shared linear projection and optimizes a proper scoring family for alignment with judge scores.

Automated Mechanism Design and Differentiable Economics Automated mechanism design uses computational techniques to search for good mechanisms in specific problem settings. Early work uses linear programming (Conitzer & Sandholm, 2003a;b; Sandholm et al., 2007; Conitzer & Sandholm, 2004). Learning-theoretic work instead selects a high-performing mechanism from a class using samples from the type distribution (Roughgarden & Schrijvers, 2016; Morgenstern & Roughgarden, 2016; 2015; Balcan et al., 2008; Feldman et al., 2014; Hsu et al., 2016; Balcan et al., 2016; 2018b;a). Work sometimes called "differentiable economics" learns mechanisms with neural networks as general function approximators (Dütting et al., 2024), or with specialized architectures that guarantee strategyproofness in single-reporter (Shen et al., 2019; Dütting et al., 2024; Curry et al., 2024) and multi-reporter settings (Curry et al., 2022; Duan et al., 2023; Wang et al., 2024). Our work likewise searches computationally over a class of scoring systems, while imposing proper-scoring-rule structure and learning from a relatively small number of judge-scored instances. With more training data, richer differentiable parameterizations are a promising direction.

A.1 Textual Elicitation and Scoring-Rule Optimization

Textual Elicitation Several recent papers design scoring systems to elicit textual information from language models. Kimpara et al. (2023) treat a generative model and a target distribution as black boxes accessible through independent and identically distributed samples, and design proper losses for the reported distribution rather than for a single reported text. Lu et al. (2024) construct truthful peer-prediction scoring systems that compare reported texts without access to ground truth texts, whereas Wu & Hartline (2024) construct proper scoring systems using ground truth texts through LLM summarization and question answering. Related scoring systems estimate mutual information between a reported text and a ground truth text: Padmakumar & He (2021) use pointwise mutual information for extractive summarization, while Xu et al. (2025); Robertson & Koyejo (2025) develop LLM-based scoring systems that measure information without requiring a ground truth text. Prior work also observes that incentive guarantees alone need not make the resulting scores statistically meaningful measurements (Burrell & Schoenebeck, 2021; Xu et al., 2024). Xu et al. (2026) make this distinction explicit through an evaluation framework that separates statistical alignment with human ratings from strategic alignment under score-seeking behavior, tested through human-rating correlation, degradation sensitivity, and manipulation robustness. Our work addresses a complementary design problem: we jointly learn latent semantic dimensions and optimize a proper scoring family for alignment with an exogenous judge score. Our alignment and manipulation experiments instantiate the framework's human-rating-correlation and manipulation-robustness criteria.

Optimization of Scoring Rules There is an extensive literature characterizing proper scoring rules for numerical elicitation (McCarthy, 1956; Savage, 1971; Abernethy & Frongillo, 2012; Gneiting, 2011). More recent work optimizes scoring rules subject to normalization constraints such as boundedness. Li et al. (2022) optimize a scoring rule to incentivize binary effort in peer grading, and Hartline et al. (2023) generalize the objective to multidimensional effort. Additionally, Neyman et al. (2021) incentivize sequential and discrete effort, Papireddygari & Waggoner (2022) connect proper scoring rules to contract theory, Chen & Yu (2021) design robust scoring rules under partial knowledge of the principal's prior, and Chen et al. (2023) learn optimal scoring rules online when the information structure and signal costs are unknown.

B Numerical Scoring Details

Mathematical notation. We use unbolded symbols for scalar values (e.g., r and θ) and bold symbols for vectors (e.g., $\boldsymbol{r}$ and $\boldsymbol{\theta}$); all vectors are column vectors. The k-th coordinate of a vector $\boldsymbol{v}$ is v_k, and the Euclidean norm of that vector is $\|\boldsymbol{v}\|_2$. Semicolons within brackets denote vector concatenation. We use uppercase letters for matrices such as A, and the transpose of a matrix is denoted by $A^\top$. The set of probability distributions on a space $\mathcal{A}$ is $\Delta(\mathcal{A})$, and the cardinality of a finite set $\mathcal{A}$ is $|\mathcal{A}|$. For a random scalar v, the expectation is $\mathbf{E}[v]$ and the variance is $\mathrm{Var}(v)$. For indexed scalar observations v_i, the empirical mean is $\overline{v}$ and the empirical standard deviation is $\mathrm{sd}(v)$.

Exact scalar decomposition of the learned hypotheses. For each axis, let $Q_{A,k}(r, y) = -(r - \theta_{A,k}(y))^2$ and $L_{A,k,b}(r, y) = \log\cosh(r - b) + \tanh(r - b)(\theta_{A,k}(y) - r)$. Then

$$\begin{aligned} \widehat{s}^{\,\mathrm{Brier}}_{A,\beta}(x, y) &= \beta + \sum_{k=1}^{K} Q_{A,k}(r_{A,k}(x), y), \\ S_{\mathrm{lc}}(x, y) &= \beta + \boldsymbol{\gamma}^\top \boldsymbol{h}(y) + \sum_{k=1}^{K} \alpha_k L_{A,k,b_k}(r_{A,k}(x), y). \end{aligned} \tag{19}$$

Under Assumption 3.1, each projection $r_{A,k}(x) = (W\boldsymbol{a}_k)^\top \mathbf{E}_{\boldsymbol{z}\sim Q_x}[\boldsymbol{z}]$ is a scalar linear property, with realized statistic $\theta_{A,k}(y) = (W\boldsymbol{a}_k)^\top \boldsymbol{z}(y)$ as in Section 2. Thus the Brier weights are one and the Log-Cosh weights are $\alpha_k > 0$, with thresholds fixed after training. These weights need not sum to one. The terms depending only on the ground truth text do not affect reporting incentives.

C Implementation Details

C.1 Detailed Learning Objective

Our approach learns a textual scoring hypothesis from judge-scored data. Each instance consists of a reported text x, a ground truth text y, and a judge score s. A hypothesis combines scalar Brier or Log-Cosh scores evaluated on the projected representations of the reported and ground truth texts, with the intercept and terms depending only on the ground truth text specified in Section 3.2.

The learning problem searches jointly for a linear projection and scoring parameters that align with judge scores. Let the training dataset be $\mathcal{D}_{\mathrm{train}}$, the parameter space be Φ, and the per-example prediction loss be $\mathcal{L}$. Each parameter $\eta \in \Phi$ specifies a complete textual score $S_\eta(x, y)$. The supervised empirical-risk objective is

$$\min_{\eta\in\Phi} \quad \frac{1}{|\mathcal{D}_{\mathrm{train}}|} \sum_{(x_i,y_i,s_i)\in\mathcal{D}_{\mathrm{train}}} \mathcal{L}(S_\eta(x_i, y_i), s_i)\,. \tag{20}$$

Both families use squared prediction loss, with the clean-training regularizers described in Section 4.3. The Log-Cosh implementation weights reference groups equally in the empirical loss. Optimization seeks a low empirical objective; no global optimum is assumed. Fitting affects parameter estimation, whereas properness follows from the form of each frozen scoring rule.

C.2 Preprocessing Text Representations

For each family, both text roles use the same frozen encoder and fitted preprocessing. The retained Brier experiments use mean-pooled layers 12–15 of Llama-3.1-8B-Instruct (Dubey et al., 2024), concatenated and reduced by train-fitted standardization and PCA. The updated Log-Cosh experiments fit independent rules to Qwen3.5-27B layers 28/32/36/40/44/48 (Qwen Team, 2026) and average their scalar outputs.

For each Log-Cosh layer ℓ, clean training texts determine shared standardization, giving $\boldsymbol{z}_\ell(t)$. The group-balanced mean difference between ground truth text and reported text representations defines

the unit role direction $\boldsymbol{b}_\ell$ in Equation (31). Decompose

$$\begin{aligned} q_\ell(t) &= \boldsymbol{b}_\ell^\top \boldsymbol{z}_\ell(t), \\ \boldsymbol{z}_{\ell,\perp}(t) &= \boldsymbol{z}_\ell(t) - q_\ell(t)\boldsymbol{b}_\ell. \end{aligned} \tag{21}$$

Fit unwhitened PCA to the pooled orthogonal training features, then append the role coordinate $q_\ell(t)$. The experimental tables count only ordinary PCA coordinates in p; the full Log-Cosh feature dimension is therefore $p+1$. The hypothesis definitions below use p generically for the full feature dimension. Rather than deleting the role coordinate, the objective penalizes the squared overlap of each unit scoring axis with it, with coefficient 20, and applies unit ridge to the linear term in the ground truth text representation. Each layer averages four fixed seeds; the final prediction averages the six layer scores. Details and clean half-standard-error (half-SE) capacity selection appear in Section E.

The same fitted centering and linear transforms apply to both roles and all splits, preserving Equation (7) whenever it holds before preprocessing. The constant semantic coordinate described in Section 3 absorbs the shared centering, so the preprocessed representation remains a linear property of the augmented semantic space. For Log-Cosh QA encoding, each text input includes the same public question context and pooling uses the answer span; theoretical comparisons hold that context fixed. The backbone is never fine-tuned.

C.3 Complete Hypothesis Class

For each family separately, let $\zeta \in \Xi$ collect the scoring parameters: $\zeta = \beta$ for Brier and $\zeta = (\boldsymbol{\alpha}, \boldsymbol{b}, \beta, \boldsymbol{\gamma})$ for Log-Cosh. Thus $\Xi = \mathbb{R}$ for Brier and $\Xi = (0,\infty)^K \times \mathbb{R}^K \times \mathbb{R} \times \mathbb{R}^p$ for Log-Cosh. Writing $S_{A,\zeta}$ for the corresponding explicit score in Equations (10) and (11), the textual hypothesis class is

$$\mathcal{H}_S = \left\{ S_{A,\zeta} : A \in \mathbb{R}^{p\times K},\ \zeta \in \Xi \right\}. \tag{22}$$

The Log-Cosh ground truth text term can depend on the full ground truth text representation, not only on $\boldsymbol{\theta}_A(y)$; it does not affect reporting incentives. All learned parameters are frozen before evaluation.

C.4 Direction–Scale Parameterization

The implementation parameterizes each column as $\boldsymbol{a}_k = s_k \boldsymbol{u}_k$, with $\|\boldsymbol{u}_k\|_2 = 1$ and $s_k > 0$. A threshold t_k before scaling corresponds to $b_k = s_k t_k$, so

$$\begin{aligned} r_{A,k}(x) - b_k &= s_k(\boldsymbol{u}_k^\top \boldsymbol{h}(x) - t_k), \\ \theta_{A,k}(y) - b_k &= s_k(\boldsymbol{u}_k^\top \boldsymbol{h}(y) - t_k). \end{aligned} \tag{23}$$

Thus the matrix notation gives exactly the same score as the direction–scale parameterization: the threshold cancels in the difference between outcome-statistic and reported-property coordinates.

D Proofs of the Approximate-Properness Results

Proof of Theorem 3.2. Condition on $\mathcal{I}_x$ and write $\boldsymbol{\mu} = \boldsymbol{\mu}_{A,x}$. For Brier, conditional bias–variance decomposition gives

$$\mathbf{E}[S_{\mathrm{Brier},A}(\boldsymbol{r}, Y) \mid \mathcal{I}_x] = C_{\mathrm{Brier},x} - \|\boldsymbol{r} - \boldsymbol{\mu}\|_2^2, \tag{24}$$

where $C_{\mathrm{Brier},x} = \beta - \mathbf{E}[\|\boldsymbol{\theta}_A(Y) - \boldsymbol{\mu}\|_2^2 \mid \mathcal{I}_x]$ is independent of the report. This expression is maximized at $\boldsymbol{r} = \boldsymbol{\mu}$.

For Log-Cosh, linearity of conditional expectation gives

$$\begin{aligned} &\mathbf{E}[S_{\mathrm{lc},A}(\boldsymbol{r}, Y) \mid \mathcal{I}_x] \\ &\quad = C_{\mathrm{lc},x} + \sum_{k=1}^{K} \alpha_k \left[\log\cosh(r_k - b_k) + \tanh(r_k - b_k)(\mu_k - r_k) \right], \end{aligned} \tag{25}$$

where $C_{\mathrm{lc},x} = \mathbf{E}[\beta + \boldsymbol{\gamma}^\top \boldsymbol{h}(Y) \mid \mathcal{I}_x]$ is independent of the report. The derivative of the k-th summand with respect to r_k is $\alpha_k \operatorname{sech}^2(r_k - b_k)(\mu_k - r_k)$. It is positive below μ_k and negative above μ_k, so each coordinate is maximized at $r_k = \mu_k$.

Table 3: Dataset overview and the fixed train/validation/test splits used in all experiments.

Dataset	One-sentence description	Train	Val.	Test
Peer Grading	A peer review is scored against an instructor review of the same algorithms submission; targets are the human score and two LLM-judge scores.	600	150	150
Simulated Peer Grading	Controlled reviews encode three binary qualities (answer, proof, and writing), giving a known score for matching the ground truth text.	2,400	300	300
MedAESQA	Generated answers to consumer-medical questions are compared with expert-curated ground truth texts (Gupta et al., 2025); targets are two LLM judges.	600	300	300
LFQA-E	Long-form answers are evaluated against ground truth texts in a grouped English subset of LFQA-E (Fan et al., 2026); targets are two LLM judges.	2,400	300	300

Thus $\boldsymbol{\mu}$ maximizes conditional expected score for both families. Subtracting the expected score at $\boldsymbol{r}_A(x)$ gives the exact gain in Equation (16), and no alternative numerical report can exceed it. This proves Equation (15). Every alternative text induces a numerical report, so the same bound applies to textual deviations even when no text realizes $\boldsymbol{\mu}$. □

Proof of Theorem 3.3. Write $\boldsymbol{r} = \boldsymbol{r}_A(x)$, $\boldsymbol{\mu} = \boldsymbol{\mu}_{A,x}$, and $d_k = \mu_k - r_k = -\delta_{A,k}(x)$. Integrating the coordinate derivative of the expected Log-Cosh score from r_k to μ_k gives

$$\begin{aligned}
\varepsilon_{\mathrm{lc}} &= \sum_{k=1}^{K} \alpha_k d_k^2 \int_0^1 (1-t)\,\mathrm{sech}^2(r_k + t d_k - b_k)\,dt \\
&\le \frac{1}{2}\sum_{k=1}^{K} \alpha_k \delta_{A,k}(x)^2 = \frac{1}{2}\boldsymbol{\delta}_A(x)^\top \Lambda \boldsymbol{\delta}_A(x) \\
&= \frac{1}{2}\boldsymbol{\delta}_h(x)^\top A \Lambda A^\top \boldsymbol{\delta}_h(x) \\
&\le \frac{L}{2}\|\boldsymbol{\delta}_h(x)\|_2^2.
\end{aligned} \tag{26}$$

The first inequality uses $0 < \mathrm{sech}^2(u) \le 1$ and $\int_0^1 (1-t)\,dt = 1/2$. The identity in Equation (14) gives the expression in the original representation space, and the final inequality uses $L = \lambda_{\max}(A\Lambda A^\top)$. This proves Equation (17) directly from the score formula. □

Proof of Theorem 3.4. By Equation (24), the expected-score gain from replacing $\boldsymbol{r}_A(x)$ with $\boldsymbol{\mu}_{A,x}$ is exactly $\|\boldsymbol{r}_A(x) - \boldsymbol{\mu}_{A,x}\|_2^2 = \|\boldsymbol{\delta}_A(x)\|_2^2$. Substituting $\boldsymbol{\delta}_A(x) = A^\top \boldsymbol{\delta}_h(x)$ proves Equation (18). □

E Detailed Experimental Results

This appendix records the complete protocol and confidence intervals behind Table 4. The learned comparison contains two scoring-rule families: Brier (quadratic) and Log-Cosh, both fitted by squared prediction loss. The tables additionally reproduce the source-reported TVD-FVar-Statement baseline, raw LLM judges, and local ROUGE-L, BLEU, and BERTScore baselines. Controlled Log-Cosh ablations are reported in Section F.

E.1 Data Construction and Splits

The Peer Grading release contains 917 rows, of which 600, 150, and 150 form the training, validation, and test splits; 17 legacy rows without a release split are unused. Simulated Peer Grading and LFQA-E each contain 3,000 rows split 2,400/300/300. MedAESQA contains 1,200 candidate answers split 600/300/300. Splits are deterministic and grouped so that a reference group—a submission,

simulated ground truth text instance, medical question, or LFQA-E question–ground truth text group—never crosses training, validation, and test. The corresponding numbers of test reference groups are 24, 50, 10, and 75.

The simulated target is the fraction of three matching binary attributes and is the only target reported for that dataset. Claude Haiku 4.5 and GPT-5-mini assign 1–5 integer scores at temperature zero for the remaining LLM targets. The updated Log-Cosh MedAESQA rubric includes the medical question, reported text, and ground truth text, and its LFQA-E rubric includes question and context. The retained Brier QA experiments use question-omitting rubrics. Reported and ground truth texts remain separately delimited in every prompt.

E.2 Representation, Scores, and Optimization

Brier (quadratic). The retained Brier pipeline mean-pools text-span activations at layers 12–15 of frozen Llama-3.1-8B-Instruct, concatenates the four vectors, and fits feature standardization and PCA on the training split only. Its complete grid is

$$p \in \{16, 32, 64, 128, 256\}, \qquad K \in \{3, 4, 6, 9, 12, 16\}. \tag{27}$$

Every configuration uses the Brier scoring rule

$$\ell(x, y) = \beta - \|A^\top \boldsymbol{h}(x) - A^\top \boldsymbol{h}(y)\|_2^2. \tag{28}$$

For Brier, the predicted score is $\ell(x, y)$ and its mean squared prediction loss is

$$\mathcal{L}_{\mathrm{MSE}} = \frac{1}{n} \sum_{i=1}^{n} (\ell(x_i, y_i) - s_i)^2. \tag{29}$$

For each Brier axis, independently sort the reported-text and ground truth text training projections and compute their empirical one-dimensional Wasserstein distance, divided by the pooled projection standard deviation, to obtain D_k. The full training objective is

$$\begin{aligned} \mathcal{L}_{\mathrm{train}} &= \mathcal{L}_{\mathrm{MSE}} + \rho S_s D_{\mathrm{RMS}}, \\ D_{\mathrm{RMS}} &= \sqrt{\frac{1}{K} \sum_{k=1}^{K} D_k^2}, \qquad \rho = 0.5, \end{aligned} \tag{30}$$

where $S_s = \mathrm{Var}(s)$ places the penalty on the reducible scale of the supervised objective. Each configuration is trained for 1,000 AdamW steps with learning rate 0.003, weight decay 0.05, and validation every 25 steps. Seeds $0, 1, 2, 3$ each retain their minimum clean-validation-MSE checkpoint; the (p, K) minimizing the equal-weight ensemble validation MSE is selected.

Log-Cosh representation. The main scoring rule combines independent fits to Qwen3.5-27B layers 28/32/36/40/44/48, following the architecture-based choice in Section 4.3. Peer and Simulated text is encoded verbatim; for MedAESQA and LFQA-E, the public question is prepended identically to reported and ground truth texts. We independently process these six layers. Within every layer, we mean-pool text-span activations and fit coordinate-wise standardization on pooled clean training texts only. Let $\boldsymbol{z}_\ell(t)$ be the standardized representation at layer ℓ. From clean training groups $\mathcal{G}$, we estimate

$$\boldsymbol{b}_\ell = \mathrm{normalize}\left[\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \frac{1}{|g|} \sum_{i \in g} \big(\boldsymbol{z}_\ell(y_i) - \boldsymbol{z}_\ell(x_i) \big) \right]. \tag{31}$$

Writing $q_\ell(t) = \boldsymbol{b}_\ell^\top \boldsymbol{z}_\ell(t)$, we decompose $\boldsymbol{z}_\ell(t)$ into the role coordinate $q_\ell(t)$ and $\boldsymbol{z}_{\ell,\perp}(t) = \boldsymbol{z}_\ell(t) - q_\ell(t)\boldsymbol{b}_\ell$. A randomized PCA is fitted to pooled orthogonal features with a fixed 512-component fit budget. The retained PCA prefix is not whitened, and the scalar role coordinate is appended. Accordingly, the reported p counts ordinary PCA coordinates; the input to each layer rule has dimension $p + 1$.

The layers have independent PCA transforms and scoring parameters but share the selected target-specific (p, K). We average four seeds within each layer, then six scalar layer scores: 24 component scoring rules with fixed equal weights. As established in Section 3, this preserves properness when

the same report is truthful for every component, and averages the component approximation-error bounds otherwise.

The Llama-3.1-8B backbone control uses consecutive middle layers 12–15, whose blocks all use standard self-attention. It follows the same fitting procedure and reuses the Qwen-selected capacities; it is not retuned on Llama.

Log-Cosh optimization and selection. Each Log-Cosh hypothesis uses Equation (11). The target is standardized by its clean-training mean and standard deviation, and supervised squared error is weighted equally across reference/question groups. For layer ℓ, the complete objective is

$$\mathcal{L}_{\text{train},\ell} = \mathcal{L}_{\text{fit},\ell} + 20 \sum_{k=1}^{K} (\widetilde{\boldsymbol{u}}_{\ell k}^{\top} \boldsymbol{e}_q)^2 + \|\boldsymbol{\gamma}\|_2^2, \tag{32}$$

where $\widetilde{\boldsymbol{u}}_{\ell k}$ is the unit direction of scoring axis k, $\boldsymbol{e}_q$ is the unit vector of the appended role coordinate, and $\boldsymbol{\gamma}$ is the coefficient of the linear term in the ground truth text representation. This ordinary ridge prevents the flexible ground truth text term from overfitting while leaving report incentives unchanged. Each scoring axis is direction-normalized separately from its learned positive scale and amplitude. The role penalty is dimensionless and uses the same coefficient for every target.

The complete capacity grid is

$$\begin{aligned} p &\in \{16, 32, 64, 128, 256, 384, 512\}, \\ K &\in \{4, 8, 16, 32, 64\}. \end{aligned} \tag{33}$$

Each layer run uses 1,000 Adam steps at learning rate 0.003 and validation every 25 steps. Seeds 0–3 are trained independently at each layer; each seed retains the checkpoint minimizing clean standardized validation squared error plus the role penalty. For each candidate (p, K), predictions are averaged over the four seeds and then over the six layers before selection. For each target, a reference-group bootstrap estimates the standard error of the minimum ensemble validation squared error. Among capacities no more than 0.5 standard errors above that minimum, we select the fewest trainable parameters and use validation squared error to break exact size ties. The selected target-specific capacity is then shared across the six layers. Capacity selection itself therefore uses unpenalized clean validation squared error, while checkpoint selection uses clean validation MSE plus the role-overlap penalty. The ridge on the ground truth text coefficient is used in optimization, not checkpoint selection. This half-standard-error (half-SE) rule favors fewer parameters when validation errors are close; it is a parsimony heuristic, not a test of statistical equivalence.

Selection and evaluation boundary. Preprocessing and fitting use clean training data, and checkpoints and capacities use the clean validation criteria above. These choices use neither test labels/correlations nor manipulation prompts, rewrites, or robustness statistics. Seeds and layers are averaged with fixed equal weights. After capacity selection, the same frozen ensemble per target supplies both clean test alignment and manipulation results. The implementation loads rewrite embeddings only after capacity selection is complete. Thus robustness is an evaluation of the validation-selected scoring rule, not a separate selection objective or a search for the most robust checkpoint.

E.3 Complete Alignment Results

Table 5 reports the selected (p, K) and every clean validation/test Pearson and Spearman point estimate. Log-Cosh validation correlations are reported as point estimates, and test correlations include 95% confidence intervals. Brier entries show the point estimate plus or minus the half-width of a 95% interval, computed as 1.96 times the standard deviation over 50,000 reference-group bootstrap resamples. Log-Cosh entries use percentile intervals from 50,000 resamples of the same whole reference groups. All are conditional on the validation-selected configuration and do not treat training seeds as independent test samples. TVD-FVar-Statement instead uses its source report's candidate-level percentile-bootstrap intervals, with $n = 534$, 400, and 200 on Peer Grading, MedAESQA, and LFQA-E (Xu et al., 2026); we reproduce those values rather than substituting our rerun. The baseline has no Simulated result, validation result, or reported hyperparameters. Prediction MSE values are retained only as selection diagnostics because their scales differ across targets.

Table 4: Alignment and manipulation point estimates. Our Log-Cosh scoring rule averages independently fitted Qwen3.5-27B layers 28/32/36/40/44/48, with 4 seeds per layer, a restored role coordinate penalized at coefficient 20, and half-SE capacity selection. Correlations are computed on clean validation and test sets. Manipulation columns report standardized effects; bold effects have 95% intervals entirely above zero. Raw judges, ROUGE-L, BLEU, BERTScore, and our scoring rule are evaluated on our fixed test splits. A dagger marks TVD-FVar-Statement values cited from Xu et al. (2026) rather than recomputed here; those rows use Xu et al.'s evaluation subsets. The retained Brier rows use the original canonical $\rho = 0.5$ protocol and report legacy paired Cohen's d_z manipulation effects; all other manipulation rows report pooled-SD d. "–" is unavailable. Complete intervals, all local-baseline targets, and selected hyperparameters appear in Tables 5 to 7.

Target	Scoring rule	Validation		Test		Manipulation effect					
		Pearson	Spearman	Pearson	Spearman	H	M	N	P	R	S
					(a) Peer Grading						
Human	ROUGE-L	–	–	.215	.229	**+.218**	−.510	−.375	−.122	**+.449**	**+.414**
Human	BLEU	–	–	.307	.290	**+.608**	+.129	+.222	**+.361**	**+.636**	**+.601**
Human	BERTScore	–	–	.096	.063	**+.270**	−.425	−.326	−.108	**+.409**	**+.435**
Human	TVD-FVar-Statement†	–	–	.408	.468	−.040	+.050	−.190	−.140	+.000	+.000
Human	Brier	.631	.468	.536	.452	−.276	−.096	−.723	−.261	−.322	−.286
Human	Ours (Log-Cosh)	.731	.697	.618	.548	−.099	−.217	−.033	−.198	−.098	−.073
Haiku 4.5	Raw judge	–	–	–	–	**+.202**	+.015	**+.407**	**+.132**	**+.157**	**+.173**
Haiku 4.5	TVD-FVar-Statement†	–	–	.564	.568	−.040	+.050	−.190	−.140	+.000	+.000
Haiku 4.5	Brier	.485	.550	.559	.602	−.320	−.424	+.045	−.647	−.327	−.296
Haiku 4.5	Ours (Log-Cosh)	.681	.686	.651	.674	−.041	−.147	−.083	−.146	−.030	−.017
GPT-5-mini	Raw judge	–	–	–	–	**+.143**	**+.150**	**+.400**	**+.138**	**+.160**	**+.118**
GPT-5-mini	TVD-FVar-Statement†	–	–	.626	.619	−.040	+.050	−.190	−.140	+.000	+.000
GPT-5-mini	Brier	.520	.530	.663	.700	−.347	−.276	−.021	−.679	−.409	−.365
GPT-5-mini	Ours (Log-Cosh)	.722	.721	.649	.665	−.096	−.193	−.177	−.186	−.075	−.034
					(b) Simulated Peer Grading						
Outcome score	Brier	.710	.723	.731	.767	−.571	−.060	−.738	−.550	−.585	−.646
Outcome score	Ours (Log-Cosh)	.933	.928	.933	.929	−.068	+.069	−.171	−.036	−.049	−.094
					(c) MedAESQA						
Haiku 4.5	Raw judge	–	–	–	–	+.023	−.065	–	–	+.075	+.015
Haiku 4.5	TVD-FVar-Statement†	–	–	.643	.645	−.080	−.030	–	–	−.080	+.010
Haiku 4.5	Brier	.490	.511	.444	.389	+.029	−.613	–	–	−.461	+.033
Haiku 4.5	Ours (Log-Cosh)	.590	.534	.669	.618	+.008	−.198	–	–	−.266	+.009
GPT-5-mini	Raw judge	–	–	–	–	+.042	+.036	–	–	−.067	+.052
GPT-5-mini	TVD-FVar-Statement†	–	–	.659	.663	−.080	−.030	–	–	−.080	+.010
GPT-5-mini	Brier	.596	.522	.596	.529	+.049	−.416	–	–	−.663	−.087
GPT-5-mini	Ours (Log-Cosh)	.615	.544	.623	.546	−.006	−.183	–	–	−.266	−.009
					(d) LFQA-E						
Haiku 4.5	Raw judge	–	–	–	–	**+.220**	−.141	–	–	+.010	**+.110**
Haiku 4.5	TVD-FVar-Statement†	–	–	.351	.326	+.000	+.050	–	–	−.040	−.010
Haiku 4.5	Brier	.185	.158	.204	.219	−.078	−.270	–	–	−.140	−.100
Haiku 4.5	Ours (Log-Cosh)	.406	.420	.425	.415	+.012	−.009	–	–	+.008	+.012
GPT-5-mini	Raw judge	–	–	–	–	**+.276**	**+.080**	–	–	+.026	**+.170**
GPT-5-mini	TVD-FVar-Statement†	–	–	.367	.342	+.000	+.050	–	–	−.040	−.010
GPT-5-mini	Brier	.323	.294	.147	.106	−.034	−.251	–	–	−.117	−.117
GPT-5-mini	Ours (Log-Cosh)	.372	.369	.467	.449	+.006	+.002	–	–	+.006	+.004

The local baselines use ROUGE-L F1 with stemming, SacreBLEU sentence BLEU with exponential smoothing, and RoBERTa-large layer-17 BERTScore F1 without baseline rescaling or a target-dependent fit (Lin, 2004; Papineni et al., 2002; Zhang et al., 2020). Their intervals use the same 50,000 whole-reference-group percentile-bootstrap resamples as Log-Cosh.

Table 5: Complete alignment results with 95% confidence intervals where available. Our scoring rule and the report-level ROUGE-L, BLEU, and BERTScore baselines use 50,000 percentile-bootstrap resamples of whole reference groups on our fixed test splits. Our intervals are conditional on the clean-validation-selected configuration, and Log-Cosh averages 4 seeds within each of 6 independently fitted layers. Its tabulated p counts ordinary PCA coordinates; each component scoring rule additionally receives one role coordinate. Validation Log-Cosh correlations are point estimates. The retained Brier rows use the original canonical $\rho = 0.5$ protocol and report estimate $\pm$ the half-width of a 95% reference-group-bootstrap interval. A dagger marks TVD-FVar-Statement values cited from Xu et al. (2026) rather than recomputed on our fixed splits.

Target	Scoring rule	p	K	Validation		Test	
				Pearson	Spearman	Pearson	Spearman
				Peer Grading			
Human	ROUGE-L	–	–	–	–	.215 [−.001, .398]	.229 [.009, .408]
Human	BLEU	–	–	–	–	.307 [.093, .489]	.290 [.011, .518]
Human	BERTScore	–	–	–	–	.096 [−.193, .348]	.063 [−.224, .317]
Human	TVD-FVar-Statement†	–	–	–	–	.408 [.353, .463]	.468 [.397, .532]
Human	Brier	32	16	.631 ± .184	.468 ± .170	.536 ± .128	.452 ± .201
Human	Ours (Log-Cosh)	32	8	.731	.697	.618 [.528, .704]	.548 [.429, .651]
Haiku 4.5	ROUGE-L	–	–	–	–	.274 [.083, .452]	.373 [.149, .536]
Haiku 4.5	BLEU	–	–	–	–	.370 [.193, .533]	.490 [.275, .625]
Haiku 4.5	BERTScore	–	–	–	–	.293 [.097, .452]	.335 [.121, .490]
Haiku 4.5	TVD-FVar-Statement†	–	–	–	–	.564 [.502, .624]	.568 [.508, .623]
Haiku 4.5	Brier	16	16	.485 ± .125	.550 ± .157	.559 ± .103	.602 ± .136
Haiku 4.5	Ours (Log-Cosh)	32	4	.681	.686	.651 [.576, .710]	.674 [.576, .736]
GPT-5-mini	ROUGE-L	–	–	–	–	.310 [.124, .462]	.358 [.143, .518]
GPT-5-mini	BLEU	–	–	–	–	.396 [.222, .542]	.462 [.217, .634]
GPT-5-mini	BERTScore	–	–	–	–	.210 [−.022, .384]	.221 [−.020, .400]
GPT-5-mini	TVD-FVar-Statement†	–	–	–	–	.626 [.573, .673]	.619 [.560, .672]
GPT-5-mini	Brier	16	16	.520 ± .098	.530 ± .161	.663 ± .069	.700 ± .102
GPT-5-mini	Ours (Log-Cosh)	32	8	.722	.721	.649 [.538, .729]	.665 [.538, .753]
				Simulated Peer Grading			
Outcome score	ROUGE-L	–	–	–	–	.108 [.007, .201]	.072 [−.027, .171]
Outcome score	BLEU	–	–	–	–	.137 [.028, .239]	.122 [.014, .225]
Outcome score	BERTScore	–	–	–	–	.192 [.092, .287]	.177 [.069, .280]
Outcome score	Brier	256	3	.710 ± .061	.723 ± .069	.731 ± .059	.767 ± .057
Outcome score	Ours (Log-Cosh)	512	4	.933	.928	.933 [.917, .947]	.929 [.909, .943]
				MedAESQA			
Haiku 4.5	ROUGE-L	–	–	–	–	.381 [.165, .572]	.318 [.072, .547]
Haiku 4.5	BLEU	–	–	–	–	.414 [.206, .603]	.450 [.208, .653]
Haiku 4.5	BERTScore	–	–	–	–	.453 [.332, .583]	.421 [.269, .551]
Haiku 4.5	TVD-FVar-Statement†	–	–	–	–	.643 [.564, .707]	.645 [.575, .707]
Haiku 4.5	Brier	256	16	.490 ± .139	.511 ± .166	.444 ± .165	.389 ± .191
Haiku 4.5	Ours (Log-Cosh)	128	4	.590	.534	.669 [.500, .776]	.618 [.399, .768]
GPT-5-mini	ROUGE-L	–	–	–	–	.413 [.244, .578]	.320 [.108, .524]
GPT-5-mini	BLEU	–	–	–	–	.404 [.252, .556]	.420 [.226, .589]
GPT-5-mini	BERTScore	–	–	–	–	.425 [.277, .589]	.357 [.208, .521]
GPT-5-mini	TVD-FVar-Statement†	–	–	–	–	.659 [.576, .729]	.663 [.591, .726]
GPT-5-mini	Brier	256	16	.596 ± .129	.522 ± .163	.596 ± .069	.529 ± .128
GPT-5-mini	Ours (Log-Cosh)	256	4	.615	.544	.623 [.419, .775]	.546 [.323, .725]
				LFQA-E			
Haiku 4.5	ROUGE-L	–	–	–	–	.164 [.052, .270]	.174 [.052, .286]
Haiku 4.5	BLEU	–	–	–	–	.098 [−.013, .217]	.108 [−.022, .232]
Haiku 4.5	BERTScore	–	–	–	–	.176 [.060, .301]	.186 [.065, .305]
Haiku 4.5	TVD-FVar-Statement†	–	–	–	–	.351 [.226, .465]	.326 [.196, .448]
Haiku 4.5	Brier	64	12	.185 ± .120	.158 ± .117	.204 ± .129	.219 ± .116
Haiku 4.5	Ours (Log-Cosh)	128	4	.406	.420	.425 [.320, .515]	.415 [.289, .523]
GPT-5-mini	ROUGE-L	–	–	–	–	.100 [−.037, .233]	.079 [−.061, .214]
GPT-5-mini	BLEU	–	–	–	–	.082 [−.043, .203]	.017 [−.112, .147]
GPT-5-mini	BERTScore	–	–	–	–	.058 [−.091, .215]	.061 [−.088, .211]
GPT-5-mini	TVD-FVar-Statement†	–	–	–	–	.367 [.247, .478]	.342 [.214, .460]
GPT-5-mini	Brier	256	9	.323 ± .125	.294 ± .119	.147 ± .110	.106 ± .133
GPT-5-mini	Ours (Log-Cosh)	256	4	.372	.369	.467 [.339, .569]	.449 [.309, .568]

E.4 Complete Manipulation Effects

Manipulation strategies. We use the six prompt families of Xu et al. (2026). Rephrasing (R) asks for a stylistic rewrite that preserves the candidate's meaning. Meaningless elongation (M) appends generic filler without adding task-relevant information. Positive opinion shift (P) makes the evaluation more favorable, while negative opinion shift (N) makes it less favorable, retaining the topics covered. Hedged opinion shift (H) makes claims tentative and less committal. Strong opinion shift (S) amplifies claims and expresses them with greater confidence. These are prompt objectives; the generated rewrites need not preserve semantic content, particularly under opinion shifts.

Generation and effect estimation. Claude Haiku 4.5 generates each test reported-text rewrite once; all locally evaluated scoring rules and raw judges use the same cached text with ground truth texts fixed. For clean/manipulated reported texts x_i, x_i' and ground truth text y_i, we average Log-Cosh predictions over four seeds per layer and then six layers before computing the pooled-SD effect used by Xu et al. (2026):

$$d = \frac{\overline{S}_{\text{manip}} - \overline{S}_{\text{clean}}}{\sqrt{(s^2_{\text{manip}} + s^2_{\text{clean}})/2}}. \tag{34}$$

Here s_{manip} and s_{clean} are the empirical standard deviations of manipulated and clean scores. Positive d indicates a mean score increase; bold marks effects whose 95% intervals lie entirely above zero. Log-Cosh, raw judges, and report-level ROUGE-L, BLEU, and BERTScore use our fixed test splits and 50,000 reference-group percentile-bootstrap resamples, preserving clean/manipulated pairs. Daggered TVD-FVar-Statement effects and intervals come from Table 11 of Xu et al. (2026) on their evaluation subsets. Peer and Simulated use H/M/N/P/R/S; MedAESQA and LFQA-E use H/M/R/S.

The Brier rows retain paired Cohen's $d_z = \overline{\Delta}/\operatorname{sd}(\Delta)$, where $\Delta_i = S(x_i', y_i) - S(x_i, y_i)$, and the original 95% reference-group-bootstrap half-widths. Their magnitudes are not directly comparable with pooled-SD effects. Positive-interval counts are 0/40 for Brier, 0/40 for Log-Cosh, and 16/28 for raw judges. MedAESQA has only ten reference groups.

Table 6: Complete manipulation effects for Peer Grading and Simulated Peer Grading; QA effects appear in Table 7. Entries show the estimate above its 95% confidence interval or retained half-width. Bold intervals lie entirely above zero. H, M, N, P, R, and S denote hedged opinion shift, meaningless elongation, negative opinion shift, positive opinion shift, rephrasing, and strong opinion shift. Except for the retained Brier rows, unmarked rows—including report-level ROUGE-L, BLEU, and BERTScore—use our fixed test splits and 50,000 reference-group percentile-bootstrap resamples of paired clean/manipulated units. The Brier rows retain the original canonical $\rho = 0.5$ protocol and report legacy paired Cohen's d_z estimates $\pm$ the half-width of a 95% reference-group-bootstrap interval. Daggered TVD-FVar-Statement values are cited from Table 11 of Xu et al. (2026), including its reported 95% intervals, and use that paper's evaluation subsets.

Target	Scoring rule	H	M	N	P	R	S
				Peer Grading			
Xu et al.	TVD-FVar-Statement†	−.040 [−.120, +.040]	+.050 [−.010, +.110]	−.190 [−.330, −.050]	−.140 [−.240, −.040]	+.000 [−.070, +.070]	+.000 [−.070, +.070]
Local	ROUGE-L	**+.218** [**+.020**, **+.432**]	−.510 [−.783, −.254]	−.375 [−.644, −.096]	−.122 [−.368, +.136]	**+.449** [**+.271**, **+.655**]	**+.414** [**+.264**, **+.612**]
	BLEU	**+.608** [**+.354**, **+.880**]	+.129 [−.260, +.551]	+.222 [−.194, +.702]	**+.361** [**+.019**, **+.729**]	**+.636** [**+.410**, **+.884**]	**+.601** [**+.432**, **+.809**]
	BERTScore	**+.270** [**+.147**, **+.417**]	−.425 [−.582, −.296]	−.326 [−.486, −.187]	−.108 [−.268, +.042]	**+.409** [**+.290**, **+.557**]	**+.435** [**+.336**, **+.564**]
Human	Brier	−.276 ±.257	−.096 ±.280	−.723 ±.281	−.261 ±.261	−.322 ±.248	−.286 ±.226
Human	Ours (Log-Cosh)	−.099 [−.231, +.029]	−.217 [−.343, −.105]	−.033 [−.314, +.230]	−.198 [−.358, −.052]	−.098 [−.223, +.020]	−.073 [−.181, +.032]
Haiku 4.5	Raw judge	**+.202** [**+.075**, **+.332**]	+.015 [−.096, +.110]	**+.407** [**+.219**, **+.608**]	**+.132** [**+.023**, **+.242**]	**+.157** [**+.049**, **+.272**]	**+.173** [**+.068**, **+.280**]
Haiku 4.5	Brier	−.320 ±.244	−.424 ±.209	+.045 ±.298	−.647 ±.214	−.327 ±.242	−.296 ±.241
Haiku 4.5	Ours (Log-Cosh)	−.041 [−.173, +.109]	−.147 [−.280, −.009]	−.083 [−.302, +.114]	−.146 [−.311, +.028]	−.030 [−.152, +.101]	−.017 [−.134, +.111]
GPT-5-mini	Raw judge	**+.143** [**+.061**, **+.229**]	**+.150** [**+.068**, **+.231**]	**+.400** [**+.183**, **+.624**]	**+.138** [**+.053**, **+.230**]	**+.160** [**+.076**, **+.246**]	**+.118** [**+.030**, **+.206**]
GPT-5-mini	Brier	−.347 ±.257	−.276 ±.255	−.021 ±.289	−.679 ±.245	−.409 ±.221	−.365 ±.224
GPT-5-mini	Ours (Log-Cosh)	−.096 [−.258, +.073]	−.193 [−.326, −.060]	−.177 [−.396, +.039]	−.186 [−.371, +.001]	−.075 [−.207, +.065]	−.034 [−.172, +.116]
				Simulated Peer Grading			
Local	ROUGE-L	−1.631 [−1.797, −1.489]	−1.460 [−1.605, −1.336]	−1.757 [−1.919, −1.620]	−1.606 [−1.765, −1.475]	−1.455 [−1.604, −1.326]	−1.724 [−1.920, −1.556]
	BLEU	−2.582 [−2.830, −2.377]	−2.112 [−2.307, −1.951]	−2.471 [−2.708, −2.274]	−2.494 [−2.727, −2.298]	−2.433 [−2.663, −2.242]	−2.684 [−2.938, −2.474]
	BERTScore	−3.882 [−4.194, −3.622]	−3.133 [−3.405, −2.913]	−4.082 [−4.398, −3.821]	−4.087 [−4.418, −3.810]	−3.941 [−4.241, −3.688]	−4.130 [−4.459, −3.862]
Outcome score	Brier	−.571 ±.120	−.060 ±.183	−.738 ±.200	−.550 ±.152	−.585 ±.134	−.646 ±.119
Outcome score	Ours (Log-Cosh)	−.068 [−.154, +.016]	+.069 [−.016, +.156]	−.171 [−.415, +.066]	−.036 [−.118, +.045]	−.049 [−.149, +.051]	−.094 [−.176, −.013]

Table 7: Complete manipulation effects for MedAESQA and LFQA-E. Estimates and uncertainty, bolding, effect-size conventions, and source attribution follow Table 6.

Target	Scoring rule	H	M	N	P	R	S
			MedAESQA				
		$-.080$	$-.030$			$-.080$	$+.010$
Xu et al.	TVD-FVar-Statement†	$[-.150, -.010]$	$[-.080, +.020]$	–	–	$[-.140, -.020]$	$[-.060, +.080]$
		$-.248$	$-.720$			$-.211$	$-.106$
Local	ROUGE-L	$[-.448, -.067]$	$[-1.015, -.519]$	–	–	$[-.424, -.032]$	$[-.369, +.137]$
		$-.210$	$-.780$			$-.711$	$-.266$
	BLEU	$[-.370, -.069]$	$[-1.051, -.577]$	–	–	$[-.844, -.584]$	$[-.435, -.049]$
		$-.241$	$-.792$			-1.679	$-.266$
	BERTScore	$[-.343, -.167]$	$[-1.029, -.651]$	–	–	$[-2.300, -1.285]$	$[-.400, -.161]$
		$+.023$	$-.065$			$+.075$	$+.015$
Haiku 4.5	Raw judge	$[-.048, +.095]$	$[-.112, -.020]$	–	–	$[-.011, +.171]$	$[-.089, +.107]$
		$+.029$	$-.613$			$-.461$	$+.033$
Haiku 4.5	Brier	$\pm.210$	$\pm.306$	–	–	$\pm.170$	$\pm.209$
		$+.008$	$-.198$			$-.266$	$+.009$
Haiku 4.5	Ours (Log-Cosh)	$[-.017, +.038]$	$[-.431, -.032]$	–	–	$[-.487, -.123]$	$[-.032, +.049]$
		$+.042$	$+.036$			$-.067$	$+.052$
GPT-5-mini	Raw judge	$[-.004, +.088]$	$[+.000, +.079]$	–	–	$[-.152, +.014]$	$[-.018, +.125]$
		$+.049$	$-.416$			$-.663$	$-.087$
GPT-5-mini	Brier	$\pm.119$	$\pm.252$	–	–	$\pm.408$	$\pm.187$
		$-.006$	$-.183$			$-.266$	$-.009$
GPT-5-mini	Ours (Log-Cosh)	$[-.054, +.041]$	$[-.312, -.110]$	–	–	$[-.430, -.136]$	$[-.064, +.057]$
			LFQA-E				
		$+.000$	$+.050$			$-.040$	$-.010$
Xu et al.	TVD-FVar-Statement†	$[-.090, +.090]$	$[-.020, +.120]$	–	–	$[-.120, +.040]$	$[-.100, +.080]$
		$-.184$	$-.174$			$-.135$	$-.241$
Local	ROUGE-L	$[-.280, -.095]$	$[-.269, -.080]$	–	–	$[-.216, -.060]$	$[-.359, -.137]$
		$-.086$	$+.000$			$-.085$	$-.235$
	BLEU	$[-.197, +.021]$	$[-.098, +.103]$	–	–	$[-.176, +.002]$	$[-.347, -.126]$
		$+.026$	$-.129$			$+.\mathbf{136}$	$+.013$
	BERTScore	$[-.044, +.094]$	$[-.193, -.071]$	–	–	$[+.\mathbf{081}, +.\mathbf{194}]$	$[-.059, +.083]$
		$+.\mathbf{220}$	$-.141$			$+.010$	$+.\mathbf{110}$
Haiku 4.5	Raw judge	$[+.\mathbf{138}, +.\mathbf{307}]$	$[-.214, -.075]$	–	–	$[-.067, +.084]$	$[+.\mathbf{029}, +.\mathbf{195}]$
		$-.078$	$-.270$			$-.140$	$-.100$
Haiku 4.5	Brier	$\pm.138$	$\pm.153$	–	–	$\pm.127$	$\pm.124$
		$+.012$	$-.009$			$+.008$	$+.012$
Haiku 4.5	Ours (Log-Cosh)	$[-.004, +.030]$	$[-.019, +.000]$	–	–	$[-.006, +.022]$	$[-.003, +.028]$
		$+.\mathbf{276}$	$+.\mathbf{080}$			$+.026$	$+.\mathbf{170}$
GPT-5-mini	Raw judge	$[+.\mathbf{194}, +.\mathbf{364}]$	$[+.\mathbf{016}, +.\mathbf{144}]$	–	–	$[-.054, +.108]$	$[+.\mathbf{080}, +.\mathbf{265}]$
		$-.034$	$-.251$			$-.117$	$-.117$
GPT-5-mini	Brier	$\pm.139$	$\pm.137$	–	–	$\pm.134$	$\pm.140$
		$+.006$	$+.002$			$+.006$	$+.004$
GPT-5-mini	Ours (Log-Cosh)	$[-.013, +.024]$	$[-.006, +.010]$	–	–	$[-.008, +.020]$	$[-.013, +.021]$

E.5 Approximate-Properness Errors for Expressed Uncertainty

We evaluate the approximate-properness error in Equation (16) for a frozen Log-Cosh scoring rule trained on Simulated Peer Grading with Qwen3.5-27B embeddings and $K = 4$ learned coordinates. For each of two expression types, numeric and verbal uncertainty, the test uses 100 families of similarly worded sentences at mixture probabilities $\lambda \in \{0.25, 0.50, 0.75\}$, together with their certain endpoints.

Reports and posterior-mean targets. For sentence family i, let $y_{i,0}$ and $y_{i,1}$ be the two endpoint ground truth texts, and let $x_{i,\lambda}$ be the reported text expressing uncertainty between them. The controlled belief assigns probabilities $1 - \lambda$ and λ to these outcomes. Thus the numerical report and posterior-mean target in Equation (14) are

$$\begin{aligned} \boldsymbol{r}_{i,\lambda} &= \boldsymbol{r}_A(x_{i,\lambda}), \\ \boldsymbol{\mu}_{i,\lambda} &= (1-\lambda)\boldsymbol{\theta}_A(y_{i,0}) + \lambda\boldsymbol{\theta}_A(y_{i,1}). \end{aligned} \tag{35}$$

The verbal conditions use the designated mixture probability for each uncertainty level. All representation and scoring parameters are fixed during evaluation.

Approximate-properness error. Write the expected score under this two-outcome belief as

$$\overline{S}_{i,\lambda}(\boldsymbol{r}) = (1-\lambda)S_{\mathrm{lc},A}(\boldsymbol{r}, y_{i,0}) + \lambda S_{\mathrm{lc},A}(\boldsymbol{r}, y_{i,1}). \tag{36}$$

By Theorem 3.2, the exact maximum expected-score gain over the truthful representation is

$$\varepsilon_{\mathrm{lc},i,\lambda} = \overline{S}_{i,\lambda}(\boldsymbol{\mu}_{i,\lambda}) - \overline{S}_{i,\lambda}(\boldsymbol{r}_{i,\lambda}). \tag{37}$$

This quantity also bounds the gain from an alternative text, even if no text realizes the optimal numerical report $\boldsymbol{\mu}_{i,\lambda}$. We report the corresponding upper bound $B_{i,\lambda}$ from Theorem 3.3. Errors and bounds are calculated for each reported text before aggregation. Both have the units of the evaluated score; they are not divided by an endpoint distance.

Table 8: Approximate-properness errors for numeric and verbal uncertainty under the frozen four-coordinate Log-Cosh scoring rule. Each row summarizes 100 reported texts at the specified mixture probability. We report the mean and empirical 95th percentile (P95) of $\varepsilon_{\mathrm{lc}}$, and the mean upper bound B, in the evaluated model's score units.

Expression	λ	Mean $\varepsilon_{\mathrm{lc}}$	P95 $\varepsilon_{\mathrm{lc}}$	Mean B
Numeric	0.25	0.0707	0.1782	0.3323
	0.50	0.0674	0.1694	0.3024
	0.75	0.0625	0.1543	0.3061
Verbal	0.25	0.0614	0.1147	0.4344
	0.50	0.2677	0.5403	1.3058
	0.75	0.0750	0.1418	0.3870

Results and scope. In Table 8, mean approximate-properness error ranges from 0.0614 to 0.0750 in five of the six conditions. Verbal uncertainty at $\lambda = 0.50$ has the largest mean error, 0.2677, and a 95th percentile of 0.5403; the corresponding numeric condition has mean 0.0674 and 95th percentile 0.1694. The reported mean upper bounds exceed the mean errors in all six conditions. These summaries quantify the incentive gaps on the constructed beliefs. Their means and percentiles do not establish a uniform bound over all truthful reports or beliefs. This evaluation concerns the specified four-coordinate Log-Cosh scoring rule; it does not report corresponding errors for Brier or the full six-layer ensemble.

Table 9: Controlled ablations of the Log-Cosh scoring rule. Correlations are equal-weight test-set means over eight targets (target-macro averages); brackets give 95% reference-group bootstrap CIs. Sig. ↑ counts target–manipulation pairs whose pooled-SD effect interval lies entirely above zero. Bold marks the full method.

Ablation	Setting	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
		Role penalty		
No role penalty	$\lambda = 0$	.682 [.640, .718]	.670 [.616, .712]	30/40
Full method	$\lambda = 20$	**.629 [.575, .672]**	**.606 [.537, .657]**	**0/40**
Hard role deletion	$\lambda = \infty$	.619 [.569, .658]	.591 [.527, .643]	0/40
		Capacity selection		
Direct minimum	minimum validation squared error	.613 [.552, .659]	.594 [.521, .651]	1/40
Half-SE (full method)	within 0.5 bootstrap SE	**.629 [.575, .672]**	**.606 [.537, .657]**	**0/40**
One-SE	within 1 bootstrap SE	.613 [.553, .661]	.596 [.523, .651]	0/40
		Single-layer rules		
Layer 32	one independently fitted rule	.607 [.557, .649]	.586 [.525, .636]	1/40
Layer 40	one independently fitted rule	.551 [.480, .612]	.536 [.455, .604]	2/40
Layer 48	one independently fitted rule	.584 [.536, .629]	.569 [.508, .620]	1/40
		Backbone		
Qwen3.5-27B (full method)	layers 28/32/36/40/44/48	**.629 [.575, .672]**	**.606 [.537, .657]**	**0/40**
Llama-3.1-8B	layers 12/13/14/15	.606 [.531, .659]	.575 [.485, .644]	2/40
		Seeds per layer		
4 (full method)	fixed seeds 0–3	**.629 [.575, .672]**	**.606 [.537, .657]**	**0/40**
8	fixed seeds 0–7	.624 [.568, .668]	.601 [.530, .654]	0/40
16	fixed seeds 0–15	.624 [.571, .666]	.600 [.531, .652]	0/40

F Ablation Studies

Table 9 changes one declared component of the frozen Log-Cosh protocol at a time. Correlations are equal-weight means over the eight reported targets. Their 95% percentile intervals come from 50,000 reference-group bootstrap draws; targets within the same dataset share each resampled group draw before the eight correlations are averaged. The final column counts the manipulation intervals whose lower endpoint is strictly positive under the same 50,000-draw grouped procedure.

Role penalty. Without the role penalty, target-macro Pearson/Spearman rises from $.629/.606$ to $.682/.670$, but 30/40 manipulations significantly increase the score. Exact deletion of the clean mean-gap coordinate restores 0/40 but reduces correlation to $.619/.591$. The finite coefficient therefore recovers alignment relative to hard deletion while retaining the same count of 0/40 positive manipulation intervals. Across the 48 target–layer summaries, median role-subspace overlap falls from .0990 without the penalty to .000072 at coefficient 20, verifying that the intended shortcut is actually suppressed.

Capacity selection. All three rows use the same Qwen layers, four seeds, coefficient 20, and 512-component PCA fit budget. Direct validation-squared-error minimization selects scoring rules with more parameters and yields $.613/.594$ with 1/40 positive manipulation intervals. One-SE selection retains 0/40 but has lower alignment at $.613/.596$. The half-SE rule is the clean-validation intermediate: it prefers a scoring rule with fewer parameters only when its error lies within half a bootstrap standard error of the minimum. Each capacity rule uses only clean validation quantities.

Layers, backbone, and seeds. Among representative single-layer fits, layer 32 reaches $.607/.586$ with 1/40, while layers 40 and 48 reach $.551/.536$ and $.584/.569$ with 2/40 and 1/40 positive manipulation intervals, respectively. The main six-layer score average reaches $.629/.606$ and 0/40. Replacing Qwen with Llama-3.1-8B lowers the target-macro result to $.606/.575$ and gives 2/40; this transfer control deliberately reuses the Qwen-selected capacities rather than retuning on Llama, so it is not a comparison of separately optimized backbones. Finally, equally averaging 8 or 16 seeds per layer does not improve the target-macro correlation point estimates over the fixed four-seed mean and remains at 0/40. We do not report validation-selected seed subsets, because the estimator is defined by averaging a fixed prefix of independently initialized fits.

Table 10: Role-penalty ablation by target. The $\lambda = \infty$ setting exactly removes the clean role coordinate; bold marks the full method. CIs and Sig. ↑ follow Table 9.

Dataset	Target	Role penalty	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Peer Grading	Human	$\lambda = 0$	.694 [.592, .775]	.653 [.528, .742]	6/6
		$\lambda = 20$ (full)	**.618** [**.528**, **.704**]	**.548** [**.429**, **.651**]	**0/6**
		$\lambda = \infty$	.592 [.494, .680]	.510 [.376, .621]	0/6
Peer Grading	Haiku 4.5	$\lambda = 0$	.665 [.583, .730]	.708 [.595, .769]	6/6
		$\lambda = 20$ (full)	**.651** [**.576**, **.710**]	**.674** [**.576**, **.736**]	**0/6**
		$\lambda = \infty$	.611 [.519, .685]	.639 [.516, .721]	0/6
Peer Grading	GPT-5-mini	$\lambda = 0$	.751 [.682, .808]	.774 [.684, .835]	6/6
		$\lambda = 20$ (full)	**.649** [**.538**, **.729**]	**.665** [**.538**, **.753**]	**0/6**
		$\lambda = \infty$	.629 [.511, .715]	.647 [.509, .741]	0/6
Simulated Peer Grading	Outcome score	$\lambda = 0$	.937 [.922, .951]	.931 [.911, .945]	1/6
		$\lambda = 20$ (full)	**.933** [**.917**, **.947**]	**.929** [**.909**, **.943**]	**0/6**
		$\lambda = \infty$	.934 [.918, .949]	.929 [.908, .944]	0/6
MedAESQA	Haiku 4.5	$\lambda = 0$	.705 [.572, .796]	.688 [.520, .803]	2/4
		$\lambda = 20$ (full)	**.669** [**.500**, **.776**]	**.618** [**.399**, **.768**]	**0/4**
		$\lambda = \infty$	.695 [.529, .788]	.637 [.409, .779]	0/4
MedAESQA	GPT-5-mini	$\lambda = 0$	.692 [.606, .776]	.619 [.495, .740]	2/4
		$\lambda = 20$ (full)	**.623** [**.419**, **.775**]	**.546** [**.323**, **.725**]	**0/4**
		$\lambda = \infty$	.605 [.449, .715]	.499 [.314, .678]	0/4
LFQA-E	Haiku 4.5	$\lambda = 0$	.480 [.382, .571]	.478 [.365, .577]	3/4
		$\lambda = 20$ (full)	**.425** [**.320**, **.515**]	**.415** [**.289**, **.523**]	**0/4**
		$\lambda = \infty$	.419 [.313, .509]	.413 [.288, .521]	0/4
LFQA-E	GPT-5-mini	$\lambda = 0$	.533 [.404, .632]	.506 [.365, .621]	4/4
		$\lambda = 20$ (full)	**.467** [**.339**, **.569**]	**.449** [**.309**, **.568**]	**0/4**
		$\lambda = \infty$	.466 [.339, .570]	.452 [.312, .570]	0/4

Table 11: Capacity-selection ablation by target. Variants share the clean-validation statistic and differ only in the parsimony rule; bold marks the full method. Statistics follow Table 9.

Dataset	Target	Capacity selection	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Peer Grading	Human	Direct minimum	.580 [.449, .689]	.516 [.372, .627]	0/6
		Half-SE (full)	**.618** [**.528**, **.704**]	**.548** [**.429**, **.651**]	**0/6**
		One-SE	.585 [.481, .680]	.519 [.390, .633]	0/6
Peer Grading	Haiku 4.5	Direct minimum	.655 [.571, .722]	.693 [.586, .758]	0/6
		Half-SE (full)	**.651** [**.576**, **.710**]	**.674** [**.576**, **.736**]	**0/6**
		One-SE	.616 [.522, .697]	.646 [.523, .731]	0/6
Peer Grading	GPT-5-mini	Direct minimum	.642 [.534, .723]	.664 [.535, .755]	0/6
		Half-SE (full)	**.649** [**.538**, **.729**]	**.665** [**.538**, **.753**]	**0/6**
		One-SE	.649 [.538, .728]	.665 [.536, .753]	0/6
Simulated Peer Grading	Outcome score	Direct minimum	.933 [.917, .947]	.929 [.909, .943]	0/6
		Half-SE (full)	**.933** [**.917**, **.947**]	**.929** [**.909**, **.943**]	**0/6**
		One-SE	.933 [.917, .947]	.929 [.909, .944]	0/6
MedAESQA	Haiku 4.5	Direct minimum	.611 [.377, .743]	.553 [.280, .740]	1/4
		Half-SE (full)	**.669** [**.500**, **.776**]	**.618** [**.399**, **.768**]	**0/4**
		One-SE	.669 [.501, .776]	.618 [.400, .768]	0/4
MedAESQA	GPT-5-mini	Direct minimum	.560 [.391, .683]	.521 [.335, .669]	0/4
		Half-SE (full)	**.623** [**.419**, **.775**]	**.546** [**.323**, **.725**]	**0/4**
		One-SE	.541 [.303, .720]	.492 [.236, .695]	0/4
LFQA-E	Haiku 4.5	Direct minimum	.430 [.330, .518]	.417 [.297, .524]	0/4
		Half-SE (full)	**.425** [**.320**, **.515**]	**.415** [**.289**, **.523**]	**0/4**
		One-SE	.464 [.354, .555]	.439 [.317, .543]	0/4
LFQA-E	GPT-5-mini	Direct minimum	.490 [.371, .584]	.457 [.323, .569]	0/4
		Half-SE (full)	**.467** [**.339**, **.569**]	**.449** [**.309**, **.568**]	**0/4**
		One-SE	.448 [.303, .566]	.457 [.316, .575]	0/4

F.1 Per-Target Ablation Results

The macro summary above makes the overall tradeoff visible, but it can conceal heterogeneity across datasets and targets. We therefore report every target for every retained ablation in Tables 10 to 14. Each row includes test Pearson and Spearman with the same grouped-bootstrap 95% intervals as the main appendix table, plus the number of significantly positive manipulation effects for that target. This keeps the overview compact while exposing all target-level evidence.

Table 12: Representative single-layer ablations by target. The full method instead averages six independently fitted layer rules. Statistics follow Table 9.

Dataset	Target	Layer rule	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Peer Grading	Human	Layer 32	.586 [.498, .672]	.540 [.416, .643]	0/6
		Layer 40	.551 [.432, .650]	.470 [.318, .600]	0/6
		Layer 48	.569 [.471, .669]	.516 [.396, .629]	0/6
Peer Grading	Haiku 4.5	Layer 32	.647 [.570, .707]	.678 [.575, .741]	0/6
		Layer 40	.550 [.440, .635]	.587 [.455, .673]	0/6
		Layer 48	.611 [.522, .685]	.648 [.535, .726]	0/6
Peer Grading	GPT-5-mini	Layer 32	.626 [.513, .713]	.650 [.525, .742]	0/6
		Layer 40	.633 [.506, .722]	.646 [.504, .746]	0/6
		Layer 48	.566 [.452, .655]	.587 [.457, .688]	0/6
Simulated Peer Grading	Outcome score	Layer 32	.905 [.882, .925]	.903 [.875, .925]	0/6
		Layer 40	.909 [.885, .929]	.908 [.880, .929]	0/6
		Layer 48	.911 [.888, .930]	.909 [.881, .931]	1/6
MedAESQA	Haiku 4.5	Layer 32	.597 [.374, .735]	.567 [.332, .732]	0/4
		Layer 40	.618 [.379, .734]	.573 [.294, .736]	1/4
		Layer 48	.601 [.421, .708]	.571 [.334, .712]	0/4
MedAESQA	GPT-5-mini	Layer 32	.613 [.446, .766]	.522 [.261, .748]	0/4
		Layer 40	.389 [.117, .681]	.368 [.101, .652]	0/4
		Layer 48	.462 [.291, .647]	.419 [.222, .615]	0/4
LFQA-E	Haiku 4.5	Layer 32	.432 [.325, .520]	.408 [.284, .515]	0/4
		Layer 40	.349 [.226, .451]	.343 [.208, .458]	0/4
		Layer 48	.472 [.364, .558]	.443 [.324, .545]	0/4
LFQA-E	GPT-5-mini	Layer 32	.451 [.326, .549]	.417 [.275, .537]	1/4
		Layer 40	.409 [.264, .528]	.397 [.249, .522]	1/4
		Layer 48	.483 [.362, .578]	.461 [.329, .570]	0/4

Table 13: Backbone ablation by target. Qwen uses middle full-attention layers 28/32/36/40/44/48; Llama uses middle layers 12–15. Bold marks the full method; statistics follow Table 9.

Dataset	Target	Backbone	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Peer Grading	Human	Qwen3.5-27B (full)	**.618 [.528, .704]**	**.548 [.429, .651]**	**0/6**
		Llama-3.1-8B	.650 [.548, .729]	.562 [.404, .684]	0/6
Peer Grading	Haiku 4.5	Qwen3.5-27B (full)	**.651 [.576, .710]**	**.674 [.576, .736]**	**0/6**
		Llama-3.1-8B	.618 [.534, .687]	.651 [.548, .719]	0/6
Peer Grading	GPT-5-mini	Qwen3.5-27B (full)	**.649 [.538, .729]**	**.665 [.538, .753]**	**0/6**
		Llama-3.1-8B	.663 [.580, .727]	.690 [.579, .757]	0/6
Simulated Peer Grading	Outcome score	Qwen3.5-27B (full)	**.933 [.917, .947]**	**.929 [.909, .943]**	**0/6**
		Llama-3.1-8B	.926 [.898, .948]	.921 [.886, .943]	0/6
MedAESQA	Haiku 4.5	Qwen3.5-27B (full)	**.669 [.500, .776]**	**.618 [.399, .768]**	**0/4**
		Llama-3.1-8B	.547 [.203, .737]	.440 [.077, .721]	0/4
MedAESQA	GPT-5-mini	Qwen3.5-27B (full)	**.623 [.419, .775]**	**.546 [.323, .725]**	**0/4**
		Llama-3.1-8B	.596 [.405, .725]	.515 [.248, .696]	2/4
LFQA-E	Haiku 4.5	Qwen3.5-27B (full)	**.425 [.320, .515]**	**.415 [.289, .523]**	**0/4**
		Llama-3.1-8B	.405 [.278, .511]	.377 [.250, .492]	0/4
LFQA-E	GPT-5-mini	Qwen3.5-27B (full)	**.467 [.339, .569]**	**.449 [.309, .568]**	**0/4**
		Llama-3.1-8B	.441 [.291, .562]	.445 [.298, .567]	0/4

Table 14: Seed-count ablation by target. Each setting averages a fixed prefix of independently initialized fits. Bold marks the full method; statistics follow Table 9.

Dataset	Target	Seeds per layer	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Peer Grading	Human	4 (full)	**.618 [.528, .704]**	**.548 [.429, .651]**	**0/6**
		8	.612 [.517, .701]	.550 [.423, .654]	0/6
		16	.605 [.506, .695]	.538 [.408, .647]	0/6
Peer Grading	Haiku 4.5	4 (full)	**.651 [.576, .710]**	**.674 [.576, .736]**	**0/6**
		8	.644 [.562, .710]	.667 [.560, .734]	0/6
		16	.645 [.563, .711]	.671 [.562, .742]	0/6
Peer Grading	GPT-5-mini	4 (full)	**.649 [.538, .729]**	**.665 [.538, .753]**	**0/6**
		8	.648 [.541, .728]	.662 [.535, .752]	0/6
		16	.644 [.533, .725]	.660 [.530, .749]	0/6
Simulated Peer Grading	Outcome score	4 (full)	**.933 [.917, .947]**	**.929 [.909, .943]**	**0/6**
		8	.934 [.918, .948]	.930 [.910, .945]	0/6
		16	.935 [.918, .949]	.929 [.908, .944]	0/6
MedAESQA	Haiku 4.5	4 (full)	**.669 [.500, .776]**	**.618 [.399, .768]**	**0/4**
		8	.673 [.485, .784]	.617 [.365, .778]	0/4
		16	.666 [.481, .772]	.600 [.350, .757]	0/4
MedAESQA	GPT-5-mini	4 (full)	**.623 [.419, .775]**	**.546 [.323, .725]**	**0/4**
		8	.586 [.385, .726]	.512 [.314, .676]	0/4
		16	.592 [.434, .704]	.526 [.349, .676]	0/4
LFQA-E	Haiku 4.5	4 (full)	**.425 [.320, .515]**	**.415 [.289, .523]**	**0/4**
		8	.431 [.329, .521]	.416 [.293, .526]	0/4
		16	.437 [.337, .527]	.422 [.299, .532]	0/4
LFQA-E	GPT-5-mini	4 (full)	**.467 [.339, .569]**	**.449 [.309, .568]**	**0/4**
		8	.468 [.339, .569]	.451 [.309, .569]	0/4
		16	.473 [.347, .572]	.452 [.311, .568]	0/4

F.2 SCORING-FAMILY CONTROLS

To test whether the proper-scoring form adds value beyond the learned representation and role regularization, we compare Log-Cosh with ordinary linear regression and a matched quadratic tangent score. These controls use the Log-Cosh targets, clean-fitted role decomposition, unwhitened PCA prefixes, group-balanced target-standardized squared error, fixed Qwen layers 28/32/36/40/44/48, and clean half-SE minimum-parameter selection. Test labels and manipulated texts enter neither fitting nor capacity selection.

Role-regularized linear regression. For each layer, let $\boldsymbol{c}$ and $\boldsymbol{r}$ denote the processed reported-text and ground truth text representations, including the restored role coordinate. We fit

$$S_{\mathrm{lin}}(\boldsymbol{c}, \boldsymbol{r}) = b + \boldsymbol{w}_c^\top \boldsymbol{c} + \boldsymbol{w}_r^\top \boldsymbol{r}, \qquad \mathcal{L}_{\mathrm{lin}} = \mathcal{L}_{\mathrm{fit}} + 20(\boldsymbol{w}_c^\top \boldsymbol{e}_q)^2 + \|\boldsymbol{w}_r\|_2^2. \tag{38}$$

Here $\boldsymbol{e}_q$ selects the restored role coordinate. The reported-text coefficient receives the role penalty and the ground truth text coefficient receives unit ridge. This convex problem is solved deterministically; the four declared seed members are identical, not independent fits. We search the same seven PCA dimensions $p \in \{16, 32, 64, 128, 256, 384, 512\}$, select by the clean half-SE rule, and equally average six scalar layer predictions. There is no scoring-axis rank K for this baseline.

LinReg does not in general elicit the conditional mean: its expected score is affine in the reported-text representation, with slope $\boldsymbol{w}_c$ independent of the outcome distribution. It also lacks interaction terms between reported and ground truth texts. Thus it tests whether an additive predictor with matched preprocessing and regularization recovers the observed tradeoff; it does not isolate properness from every difference in expressiveness.

Matched quadratic tangent score. This control replaces only the Log-Cosh potential by $\phi(t) = t^2/2$, giving each learned axis the contribution $a_k(c_k r_k - c_k^2/2)$ with $a_k \geq 0$, up to terms depending only on the ground truth text. For $\mu_k = \mathbb{E}[r_k]$, the expected-score regret is $a_k(c_k - \mu_k)^2/2$, so the scoring rule remains proper. It retains role-overlap coefficient 20, ridge on the ground truth text coefficient, the clean half-SE capacity rule, four-seed averaging, and fixed middle layers. The Qwen and Llama quadratic capacities are selected independently. This is a separate control from the concatenated-layer Llama Brier pipeline in Table 2, whose preprocessing, regularization, selection, and QA targets differ.

Results and interpretation. Table 15 reports every target and macro interval. Qwen LinReg reaches target-macro Pearson/Spearman $.558/.532$ (95% CIs $[.510, .603]/[.476, .584]$), versus $.629/.606$ and 0/40 positive manipulation intervals for Log-Cosh. LinReg has 23/40 positive intervals: 13/18 on Peer, 2/8 on MedAESQA, and 8/8 on LFQA-E. Its 0/6 on Simulated accompanies near-zero alignment ($-.035/-.055$), illustrating that the count alone is insufficient. Conversely, LinReg has higher point-estimate alignment than Log-Cosh on several real-data targets, including Peer GPT ($.819/.826$), but all six manipulation intervals for that target are positive.

Matched Qwen Quadratic reaches $.611/.595$ with 6/40 positive intervals; independently selected Llama Quadratic reaches $.583/.562$ with 4/40. Refitting the fixed Qwen quadratic capacities with role coefficients 40 or 80 or exact role-coordinate deletion retains the same six positive intervals; larger finite penalties slightly reduce alignment, while deletion changes it only marginally. Role regularization alone therefore does not recover the Log-Cosh alignment–robustness balance, and numerical properness alone does not imply textual robustness. These are descriptive comparisons: marginal correlation intervals do not constitute a paired significance test, and the quadratic control also shows that the choice within proper score families matters.

Table 15: Scoring-family controls. Quadratic replaces the Log-Cosh potential by $\phi(t) = t^2/2$ under the matched protocol; it is distinct from the retained Brier pipeline in Table 2. Linear Regression uses $b + \boldsymbol{w}_c^\top \boldsymbol{c} + \boldsymbol{w}_r^\top \boldsymbol{r}$ with matched role-coordinate and ground truth text penalties. Capacities are independently clean-selected by the half-SE rule, except that the existing Llama Log-Cosh control reuses Qwen-selected capacities. Predictions are averaged across fixed layers; deterministic linear fits have identical seed members. CIs use 50,000 whole-group percentile-bootstrap draws. Sig. ↑ counts positive 95% manipulation intervals.

Backbone	Scoring rule	Scope	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
Qwen3.5-27B	Log-Cosh (full)	target macro	**.629 [.575, .672]**	**.606 [.537, .657]**	**0/40**
Qwen3.5-27B	Quadratic	target macro	.611 [.563, .648]	.595 [.522, .647]	6/40
Qwen3.5-27B	Linear Regression	target macro	.558 [.510, .603]	.532 [.476, .584]	23/40
Llama-3.1-8B	Log-Cosh	target macro	.606 [.531, .659]	.575 [.485, .644]	2/40
Llama-3.1-8B	Quadratic	target macro	.583 [.525, .634]	.562 [.486, .623]	4/40

Backbone	Dataset	Target	Test Pearson [95% CI]	Test Spearman [95% CI]	Sig. ↑
		Quadratic, Qwen3.5-27B			
Qwen	Peer grading	Human	.602 [.452, .725]	.551 [.406, .667]	0/6
Qwen	Peer grading	Haiku 4.5	.656 [.577, .712]	.687 [.579, .756]	0/6
Qwen	Peer grading	GPT-5-mini	.640 [.543, .707]	.662 [.543, .739]	0/6
Qwen	Simulated peer grading	Ground truth	.826 [.779, .865]	.829 [.773, .874]	0/6
Qwen	MedAESQA	Haiku 4.5	.614 [.473, .718]	.595 [.325, .770]	2/4
Qwen	MedAESQA	GPT-5-mini	.654 [.526, .745]	.558 [.347, .714]	2/4
Qwen	LFQA-E	Haiku 4.5	.424 [.323, .511]	.424 [.304, .527]	1/4
Qwen	LFQA-E	GPT-5-mini	.469 [.345, .571]	.456 [.319, .571]	1/4
		Quadratic, Llama-3.1-8B			
Llama	Peer grading	Human	.625 [.540, .703]	.565 [.425, .669]	0/6
Llama	Peer grading	Haiku 4.5	.642 [.565, .706]	.699 [.580, .769]	0/6
Llama	Peer grading	GPT-5-mini	.700 [.634, .751]	.740 [.643, .794]	0/6
Llama	Simulated peer grading	Ground truth	.827 [.784, .865]	.840 [.795, .876]	0/6
Llama	MedAESQA	Haiku 4.5	.568 [.361, .724]	.474 [.163, .702]	2/4
Llama	MedAESQA	GPT-5-mini	.526 [.369, .656]	.411 [.221, .584]	2/4
Llama	LFQA-E	Haiku 4.5	.400 [.261, .517]	.383 [.255, .497]	0/4
Llama	LFQA-E	GPT-5-mini	.375 [.207, .519]	.385 [.233, .518]	0/4
		Linear Regression, Qwen3.5-27B			
Qwen	Peer grading	Human	.640 [.496, .745]	.622 [.428, .754]	3/6
Qwen	Peer grading	Haiku 4.5	.650 [.566, .716]	.686 [.567, .764]	4/6
Qwen	Peer grading	GPT-5-mini	.819 [.743, .874]	.826 [.736, .882]	6/6
Qwen	Simulated peer grading	Ground truth	−.035 [−.124, .056]	−.055 [−.152, .043]	0/6
Qwen	MedAESQA	Haiku 4.5	.524 [.332, .711]	.411 [.193, .659]	1/4
Qwen	MedAESQA	GPT-5-mini	.755 [.688, .814]	.697 [.629, .752]	1/4
Qwen	LFQA-E	Haiku 4.5	.557 [.465, .637]	.530 [.435, .616]	4/4
Qwen	LFQA-E	GPT-5-mini	.550 [.428, .648]	.537 [.417, .637]	4/4