Conformal Prediction with Paraphrase-Aware Scoring for LLM Uncertainty Quantification

Jiayi Xin[*] University of Pennsylvania
Evan Qiang University of Pennsylvania
Zihan Zhu University of Pennsylvania
Xiang Li[*] University of Pennsylvania
Weijie J. Su[*] University of Pennsylvania
Qi Long[*] University of Pennsylvania

Abstract

Uncertainty quantification (UQ) for large language models (LLMs) aims to provide reliable measures of predictive confidence, yet current methods are often unstable under meaning-preserving perturbations. Semantically equivalent paraphrases can induce substantial variability in predictive confidence, even for methods with formal guarantees, such as conformal prediction. To address this issue, we propose a paraphrase-aware UQ framework robust to semantic rewordings. Our approach trains a lightweight proxy model on LLM hidden states and aggregates its predictions across paraphrases to construct label-wise nonconformity scores. Under score exchangeability, conformal calibration retains marginal coverage. This guarantee can also hold under test-only rewording, provided that the paraphrase pipeline satisfies an additional distributional alignment condition. We evaluate three settings (normal, fully reworded, and semi-reworded) which apply rewording to neither dataset, both calibration and test datasets, or only the test dataset, respectively. Across seven multiple-choice QA benchmarks and multiple model families, our method produces compact prediction sets with empirical coverage generally near the nominal target, even in the semi-reworded setting. Ablation studies show that the learned proxy accounts for most of the reduction in set size, while paraphrase-augmented training and inference-time aggregation improve stability under rewording. Code is available at `https://github.com/Raina-Xin/PA_Score`.

1 Introduction

Large language models (LLMs) are increasingly deployed in high-stakes domains such as education and medicine [4, 24]. In these settings, reliable uncertainty quantification (UQ) is indispensable, as decision-makers must assess when to trust model predictions. However, UQ in LLMs is often unstable: predictive confidence can be poorly calibrated and highly sensitive to superficial variations in input phrasing [35]. Such instability undermines the reliability of uncertainty estimates and limits their practical usefulness.

Another fundamental limitation is that many existing UQ methods lack formal guarantees. Most approaches rely on heuristic confidence scores [35, 27, 40, 3, 18], making it difficult to determine whether their outputs can be trusted, especially in high-stakes scenarios. This motivates the use of methods with theoretical guarantees. Conformal prediction (CP) provides such a framework, constructing prediction sets with finite-sample coverage under mild assumptions [44, 34]. In CP, a nonconformity score measures how atypical a candidate output is for a given input and determines whether it is included in the prediction set.

[*]Corresponding authors: Jiayi Xin (`jiayixin@engineering.upenn.edu`), Xiang Li (`lx10077@upenn.edu`), Weijie J. Su (`suw@wharton.upenn.edu`), and Qi Long (`qlong@upenn.edu`).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

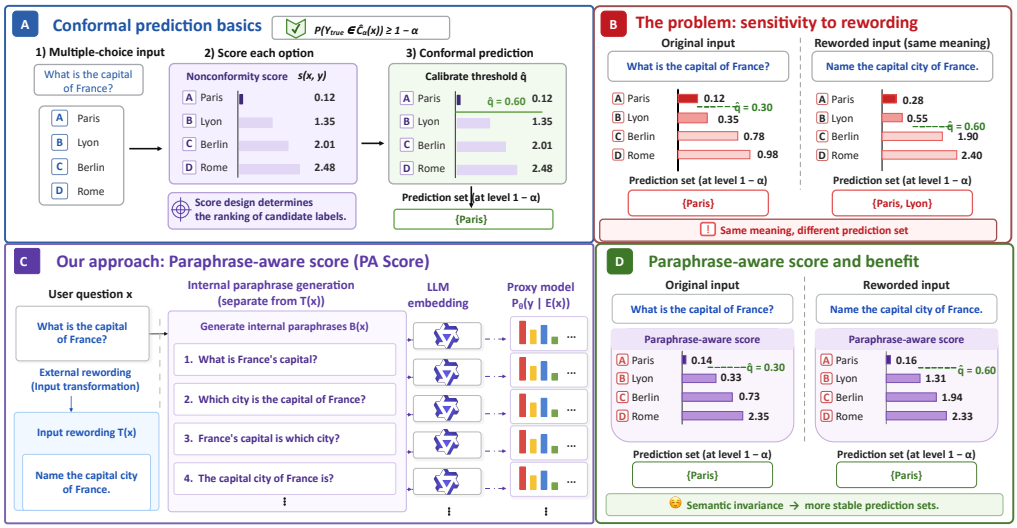


Figure 1: Prior scores are sensitive to rewording or paraphrasing of input question (denoted as $\mathcal{T}$), producing larger prediction sets (Panel A and B). Our PA framework aggregates multiple paraphrases with a proxy model to reduce score and prediction-set sensitivity to rewording (Panel C and D).

While CP ensures statistical validity, its practical reliability depends critically on the stability of the underlying uncertainty scores. Ideally, semantically equivalent inputs should yield comparable uncertainty estimates. For example, the questions "What is the capital of France?" and "Which city serves as France's capital?" should lead to similar prediction sets. In practice, however, even such paraphrases can induce substantial variability in uncertainty estimates, leading to unstable or inflated prediction sets. This simple example reveals a gap between theoretical guarantees and semantic robustness. This challenge is particularly pronounced in the LLM setting, where variability arises from semantic-preserving perturbations such as paraphrasing, rather than classical distributional shifts. Addressing such instability requires methods that can achieve both statistical validity and semantic robustness.

Contributions. To address this challenge, we introduce a paraphrase-aware scoring mechanism within the conformal prediction framework that reduces sensitivity to semantic rewordings. Rather than relying on a single input, our approach constructs uncertainty estimates by aggregating predictions across a set of paraphrases, thereby capturing variability induced by rewording. To enable consistent aggregation, we employ a lightweight proxy model to produce calibrated and comparable outputs across paraphrases. The resulting nonconformity scores can be integrated with standard CP, retaining its coverage guarantees when the required exchangeability conditions hold.

Focusing on the nonconformity score provides a convenient mechanism for incorporating problem-specific structure. Importantly, modifying the score leaves the underlying conformal guarantees unchanged, making it a simple and modular approach to improving robustness. We demonstrate that our method applies to both split CP and quasi-conditional CP (QCCP) [13]. The learned proxy accounts for most of the end-to-end set-size reduction, while paraphrase-augmented training and test-time aggregation contribute to wording stability.

Theoretically, we show that our scores preserve score-level exchangeability under paraphrased inputs, and more generally under transformations satisfying a distributional closedness condition (see Asm. 5.1). Empirically, we conduct large-scale experiments on five general QA benchmarks and two medical QA benchmarks, showing end-to-end prediction-set reductions of up to 2–4$\times$ against raw-probability baselines, with empirical coverage generally near the nominal target. We also conducted controlled ablations to separate the proxy, training-view, and inference-aggregation contributions.

2 Related Work

Conformal prediction provides finite-sample marginal coverage when calibration and test examples are exchangeable [44, 34]. Its applications to language models include conformal language modeling, multiple-choice question answering, and API-based inference [30, 22, 37]. Learned nonconformity

scores and quasi-conditional methods further adapt prediction sets to task structure [10, 13]. Our focus is the additional question of how label-wise scores and prediction sets change when an input is reworded without changing its intended meaning. Perturb-and-aggregate uncertainty estimation is established in SPUQ [12], while other work studies semantic uncertainty and prompt sensitivity [21, 29]. SPUQ's original instance-level uncertainty value is distinct from the per-option score required to form conformal prediction sets. Prior work establishes perturbation-based uncertainty estimation and conformal coverage, but leaves open how to design label-wise nonconformity scores that are stable under meaning-preserving rewording. This matters in practice because semantically equivalent queries can yield different prediction sets even when marginal coverage is maintained. Our work addresses this nonconformity score design problem through a unified treatment of methodology, evaluation, and validity analysis, extending the focus from instance-level uncertainty to the stability of conformal decisions. For detailed related work, please refer to Appendix A.

3 Preliminaries

Problem Setup. We study conformal prediction for classification when inputs may be reworded without changing their meaning or correct labels. Our goal is to construct small prediction sets that contain the correct label at a prescribed coverage level. We investigate how the choice of nonconformity score affects this coverage-efficiency tradeoff under paraphrasing, considering both matched calibration and test distributions and test-only rewording shifts. Coverage under the latter setting requires additional assumptions, which we formalize in Section 5.

Formally, we study supervised prediction with input $x \in \mathcal{X}$ (e.g., a natural language question) and a finite label space $\mathcal{Y}$ (e.g., multiple-choice answers). Given a calibration dataset $\{(X_i, Y_i)\}_{i=1}^n$ with $Y_i \in \mathcal{Y}$, the goal is to construct a prediction set $\widehat{C}(X_{n+1}) \subseteq \mathcal{Y}$ for a new data pair (X_{n+1}, Y_{n+1}) such that $\mathbb{P}\big(Y_{n+1} \in \widehat{C}(X_{n+1})\big) \geq 1 - \alpha$ for a significance level α (e.g., $\alpha = 0.01$), while remaining robust to paraphrase perturbations of the input X_{n+1}. To ensure rigorous coverage guarantees, we work within the conformal prediction framework. Within this framework, two widely used approaches are split conformal prediction (split CP) and quasi-conditional conformal prediction (QCCP).

Split CP. Given an i.i.d. calibration set $\{(X_i, Y_i)\}_{i=1}^n$, split CP [44, 34] constructs prediction sets based on a *nonconformity score* $S : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, which measures how atypical a candidate label y is for an input x. The prediction set for a new input X_{n+1} is defined as

$$\widehat{C}(X_{n+1}) = \{y : S(X_{n+1}, y) \leq \widehat{\gamma}_\alpha\}, \tag{1}$$

where $\widehat{\gamma}_\alpha$ is the empirical $(1-\alpha)$ quantile of $\{S(X_i, Y_i)\}_{i=1}^n \cup \{\infty\}$. Split CP is simple and distribution-free, but it relies on a single global threshold $\widehat{\gamma}_\alpha$ for all inputs, which can be overly conservative in heterogeneous input spaces.

Quasi-Conditional CP. Quasi-conditional conformal prediction (QCCP) [13] extends split CP by allowing thresholds to depend on the input. Given a precollected i.i.d. calibration set $\{(X_i, Y_i)\}_{i=1}^n$, it first computes nonconformity scores $S_i = S(X_i, Y_i)$ for $i = 1, \ldots, n$. These scores are then used to fit the following quantile regressor, which for any given value of S, is defined as,

$$\widehat{g}_S = \arg\min_{g \in \mathcal{F}} \left[\sum_{i=1}^n \ell_\alpha\big(g(X_i), S_i\big) + \ell_\alpha\big(g(X_{n+1}), S\big) \right], \tag{2}$$

where $\mathcal{F} = \{\phi(\cdot)^\top \beta : \beta \in \mathbb{R}^d\}$ denotes a linear function class over a feature mapping ϕ (to be specified later), and ℓ_α is the pinball loss defined by $\ell_\alpha(\theta, S) = (S - \theta)\,(\mathbf{1}_{\{S \geq \theta\}} - \alpha)$. At inference time, for a new input X_{n+1}, we construct the prediction set[2]

$$\widehat{C}(X_{n+1}) = \big\{y \,:\, S(X_{n+1}, y) \leq \widehat{g}_{S(X_{n+1},y)}(X_{n+1})\big\}. \tag{3}$$

Compared to split CP (which uses a global threshold $\widehat{\gamma}_\alpha$), QCCP adapts thresholds via $\widehat{g}_S$ based on the feature mapping ϕ; see Gibbs et al. [13] and Appendix B for more details.

4 Paraphrase-Aware Nonconformity Scores for Conformal Prediction

A key limitation of existing CP methods is that the nonconformity scores S are computed from a single prompt, and thus fail to capture variability induced by semantically equivalent rewordings.

[2]Here, we use the notation $\widehat{g}_{S(X_{n+1},y)}$ to emphasize that the learned threshold function depends on the score value $S(X_{n+1}, y)$ which is evaluated on the candidate input–label pair (X_{n+1}, y).

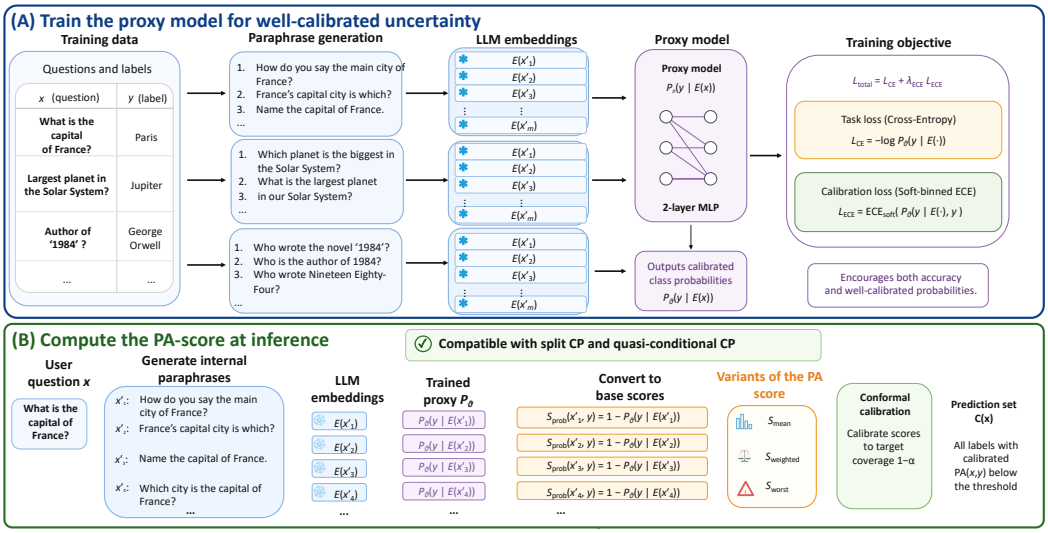


Figure 2: Schematic illustration of our proposed PA score integrated with CP.

As a result, even minor paraphrases can lead to unstable uncertainty estimates. To address this issue, we introduce *paraphrase-aware* (PA) nonconformity scores, which explicitly incorporate semantic variability by aggregating predictions across paraphrases. These scores can be integrated with either split CP or QCCP, yielding robust prediction sets to reworded inputs while retaining statistical coverage guarantees.

Overview of Score Construction. At a high level (Figure 2), our approach constructs uncertainty scores by aggregating predictions across paraphrases, thereby capturing variability induced by semantic rewordings. It proceeds as follows. For each input question x, we generate a set of paraphrases $\mathcal{B}(x)$. For each paraphrased input $x' \in \mathcal{B}(x)$, we extract its semantic embedding $E(x')$ (e.g., the corresponding LLM hidden state), resulting in a collection of embeddings $\{E(x') : x' \in \mathcal{B}(x)\}$. We then train a lightweight *proxy classifier* to map these embeddings to calibrated predictive probabilities. Based on these probabilities, we construct paraphrase-aware nonconformity scores, denoted by $S(x, y)$, which aggregate information across the paraphrase set $\mathcal{B}(x)$. The resulting scores are subsequently calibrated via split CP or QCCP to produce the final prediction set. Due to space constraints, the full procedure is summarized in Algorithm 1 in the appendix. Additional details on paraphrase quality and runtime are provided in Appendices E.5 and E.6.

Motivation for a Proxy Classifier. Raw LLM probabilities are often overconfident and poorly calibrated,[3] as they are trained for next-token prediction rather than uncertainty quantification [27, 40]. To address this limitation, we introduce a lightweight proxy classifier $P_\theta(y \mid E(x))$ that produces more reliable predictive probabilities for downstream decision making. The proxy takes the last-layer hidden state of the LLM as input and outputs a probability distribution over the label space $\mathcal{Y}$. In our implementation, the proxy is a shallow two-layer MLP trained on LLM embeddings $\{E(X_i)\}$ with labels $\{Y_i\}$, using a combination of task loss (cross-entropy) and a calibration-aware loss (soft-binned ECE; 20). This design improves calibration while remaining computationally efficient. The main variant requires hidden-state access and labeled proxy-training data. The blackbox API extension and full online generation cost are reported in Appendices G.1 and E.6.

Remark 4.1. *Since the PA score depends on embeddings $E(\cdot)$, one may be concerned about sensitivity to representation choices. In Appendix F.4, we show that the PA score remains robust across a range of embedding variants, including different ways of summarizing token-level representations (e.g., pooling) and selecting model layers.*

Paraphrase-Aware Nonconformity Scores. We now introduce paraphrase-aware nonconformity scores, the core component of our method. A standard choice in conformal prediction is the

[3]When well calibrated, the proxy outputs are also interpretable. For example, if $P_\theta(\texttt{Paris} \mid E(x)) = 0.8$, this indicates that approximately 80% of semantically similar inputs have "Paris" as the correct answer.

Table 1: Paraphrase-aware nonconformity scores. The weight is defined as $w(x, x') = 1/(1 + \|E(x) - E(x')\|_2)$, an inverse-distance weighting in the embedding space, so paraphrases closer to the original input receive larger weights. We use $\mathcal{T}(\cdot)$ to denote the paraphrase operator.

Score	Definition	Interpretation	Properties
Mean	$\frac{1}{m}\sum_{x' \in \mathcal{B}(x)} S_{\text{prob}}(x', y)$	Captures average semantic stability	Most efficient; may miss rare paraphrases
Weighted	$\frac{\sum_{x' \in \mathcal{B}(x)} w(\mathcal{T}(x), x')\, S_{\text{prob}}(x', y)}{\sum_{x' \in \mathcal{B}(x)} w(\mathcal{T}(x), x')}$	Emphasizes semantically similar paraphrases	Balances compactness and robustness
Worst-case	$\max_{x' \in \mathcal{B}(x)} S_{\text{prob}}(x', y)$	Sensitive to hardest paraphrase	Robust but conservative; sensitive to low-quality paraphrases

probability-based score [33], $S_{\text{prob}}(x, y) = 1 - P_\theta(y \mid E(x))$, which measures the model's lack of confidence for a single prompt. However, this score fails to capture instability under paraphrasing.

Given a paraphrase set $\mathcal{B}(x) = \{x'_1, \ldots, x'_m\}$ for each input x, we construct paraphrase-aware scores by aggregating $S_{\text{prob}}(x', y)$ across paraphrases. As summarized in Table 1, we consider three variants. The mean score S_{mean} averages base scores across paraphrases, capturing overall semantic stability. The weighted score S_{weighted} emphasizes semantically similar paraphrases by assigning higher weights to closer variants in the embedding space. The worst-case score S_{worst} takes the maximum across paraphrases, making it sensitive to the most difficult rewording.

Choice of the Class Function ϕ. We now describe how we construct the class function ϕ (introduced in (2)), which captures semantic similarity among input questions. At a high level, ϕ groups semantically similar inputs so that downstream procedures operate on coherent groups rather than a single global structure. Since most benchmarks lack explicit type annotations (e.g., commonsense vs. factual), we construct these groups automatically. Specifically, we embed each question–context pair using a pretrained SBERT encoder (`all-MiniLM-L6-v2`) [31], fit K-means on the calibration embeddings only, and assign each test input to its nearest centroid; the resulting cluster assignment defines $\phi(x)$. Formally, this partitions $\mathcal{X}$ into N clusters $\{\mathcal{G}_j\}_{j=1}^N$, and we set $\phi(x) = \sum_{j=1}^N \mathbf{1}\{x \in \mathcal{G}_j\} \cdot \phi_j$, where ϕ_j denotes the representative vector of cluster j. The number of clusters K can be fixed or selected via the silhouette score on calibration embeddings. This construction yields semantically coherent groupings and enables conditioning on latent question structure without requiring manual annotations. For more implementation details, see Appendix D.4.

5 Theoretical Analysis

When integrated into standard CP frameworks, our PA scores inherit the original theoretical guarantees under exchangeability, since these guarantees are agnostic to the specific choice of nonconformity score. However, such guarantees rely on the exchangeability of the resulting scores, which may be violated under distributional shifts. We show that, in certain structured settings, our PA construction can restore exchangeability at the score level, even when it does not hold for the original calibration–test pairs. We formalize this intuition below.

Assumption 5.1 (Distributional closedness)**.** *Let $\mathcal{T}$ be a Markov kernel on $\mathcal{X}$, where $\mathcal{T}(\cdot \mid x)$ denotes the distribution of a random paraphrase of x. We say $\mathcal{T}$ is distributionally closed if, for any $x \in \mathcal{X}$, $\mathcal{T}(\cdot \mid x') = \mathcal{T}(\cdot \mid x)$ for $\mathcal{T}(\cdot \mid x)$-almost every x'.*

Proposition 5.1 (Data exchangeability under distribution shift)**.** *Let Asm. 5.1 hold, and $\{(X_i, Y_i)\}_{i=1}^{n+1}$ be exchangeable (or i.i.d.). We denote by $\mathcal{T}(X_{n+1})$ a random paraphrase of X_{n+1} drawn from the kernel $\mathcal{T}(\cdot \mid X_{n+1})$. Then $(\mathcal{T}(X_{n+1}), Y_{n+1})$ may NOT be exchangeable with $\{(X_i, Y_i)\}_{i=1}^n$ due to the shift induced by $\mathcal{T}$. However, applying $\mathcal{T}$ to all inputs restores exchangeability, i.e., $\big(\mathcal{T} \circ \mathcal{T}(X_{n+1}), Y_{n+1}\big)$ is exchangeable with $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^n$.*

Prop. 5.1 shows that although the transformation $\mathcal{T}$ breaks exchangeability between calibration and test samples, this issue can be resolved by applying $\mathcal{T}$ across all data. The key mechanism is that, under Asm. 5.1, the paraphrase distribution is stable under repeated rewording $\mathcal{T}$. While $(\mathcal{T}(X_{n+1}), Y_{n+1})$ may not align with the original calibration distribution, $\mathcal{T}(X_{n+1})$ (and its multiple collections) becomes compatible with those from the calibration data.

Table 2: Summary of score exchangeability in Lemma 5.1 where $\{(X_i, Y_i)\}_{i=1}^{n+1}$ is i.i.d. Here, (*) requires Asm. 5.1.

Setting.	Calibration Data	Test Sample	Exchangeable
Normal	$\{(X_i, Y_i)\}_{i=1}^{n}$	(X_{n+1}, Y_{n+1})	✓
Full	$\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n}$	$(\mathcal{T}(X_{n+1}), Y_{n+1})$	✓
Semi	$\{(X_i, Y_i)\}_{i=1}^{n}$	$(\mathcal{T}(X_{n+1}), Y_{n+1})$	✓ (*)

Lemma 5.1 (Score exchangeability)**.** *Let $\{(X_i, Y_i)\}_{i=1}^{n+1}$ be i.i.d. For our proposed scores $S \in \{S_{\text{mean}}, S_{\text{weighted}}, S_{\text{worst}}\}$, the exchangeability properties summarized in Table 2 hold, i.e., the test score is exchangeable with the corresponding calibration scores.*

Implications for Paraphrasing. Meaning-preserving paraphrases express the same underlying content through different wordings. Distributional closedness formalizes the stronger requirement that this rewording leaves the distribution of generated paraphrases unchanged. Under this view, our PA scores leverage multiple independent samples drawn from $\mathcal{T}(\cdot \mid x)$ to construct a robust statistic for each input. Under Asm. 5.1, this invariance extends to the joint distribution of these samples. Lemma 5.1 then establishes exchangeability of the resulting scores under the stated score conditions. Consequently, our scores retain the corresponding marginal coverage guarantees within CP and quasi-conditional guarantees within QCCP under its joint feature–score assumptions, including when test inputs undergo rewording. A formal statement together with proofs is given in Appendix C.

Remark 5.1. *The analysis also quantifies departures from exact distributional closedness. Under the assumptions in Appendix C.3, a total-variation bound controls the change in coverage induced by rewording. Appendix E.4 complements this result with an empirical diagnostic of score-distribution shift and its effect on coverage.*

6 Empirical Performance

Datasets. We evaluate our framework on seven benchmark datasets. Five are general multiple-choice question answering (MCQA) datasets from the LLM-Uncertainty-Benchmark [53]: MMLU, CosmosQA, HellaSwag, HaluDial, and HaluSum, covering question answering (**QA**), reading comprehension (**RC**), commonsense inference (**CI**), dialogue response selection (**DRS**), and document summarization (**DS**). We further include two medical MCQA datasets, **MedMCQA** [28] and **MedQA** [19], to evaluate robustness in specialized domains. For each dataset, we sample 10,000 questions. Following the LLM-Uncertainty-Benchmark protocol [53], we augment the response space $|\mathcal{Y}|$ of all datasets to six options by adding `I don't know`" and `None of the above`." Together, these datasets span diverse reasoning tasks and domains, enabling a comprehensive evaluation of both generalization and robustness.

Baselines and Evaluation Metrics. We compare LAC [33] and APS [32] with label-wise adaptations inspired by TRON [46] and SPUQ [12], denoted TRON-adapt. and SPUQ-adapt. Let $p_y(x)$ denote the raw LLM probability of option y, and let $\bar{p}_y(x) = |\mathcal{B}(x)|^{-1} \sum_{\tilde{x} \in \mathcal{B}(x)} p_y(\tilde{x})$ denote its average across paraphrases. The implementations use the scores $-\log(p_y(x) + 10^{-12})$ and $-\log(\bar{p}_y(x) + 10^{-12})$, respectively. Figure labels TRON and SPUQ refer to the corresponding adaptations. We separately examine variants $1 - p_y(x)$ and $1 - \bar{p}_y(x)$ in Appendix D.3. Data are split into training, calibration, and test sets with a 40/30/30 ratio. We report Coverage Rate (CR), the fraction of test points whose true label lies in the prediction set, and Set Size (SS), the average size of the prediction set. The target coverage is $1 - \alpha = 0.90$.

Paraphrase-Based Distribution Shift. To simulate controlled distribution shifts, we generate reworded inputs by prompting a local LLM to paraphrase each question while preserving its semantics. Diversity is introduced via stochastic decoding (temperature sampling), followed by filtering to ensure validity and well-formedness. When generation fails, we retry with a stricter template or apply a rule-based rewrite. We denote the above overall (paraphrase) transformation by $\mathcal{T}$ for ease of reference. Further implementation details are provided in Appendix D.2.[4]

Evaluation Settings. We instantiate the three settings in Table 2 to distinguish the effect of rewording from a mismatch between calibration and deployment. In the **normal** setting, all training,

[4]We verify robustness to the choice of paraphrase generator and evaluate paraphrase quality in Appendix F.3 and Appendix E.5 respectively.

Table 3: Performance under split CP and QCCP with `Qwen3-8B` on normal and fully reworded (Full) test sets across five benchmarks. For coverage rate (CR, %), bold indicates values closest to the nominal 90% target. For set size (SS, ↓), bold indicates the smallest value.

	QA		RC		CI		DRS		DS	
Method	Normal	Full	Normal	Full	Normal	Full	Normal	Full	Normal	Full
Coverage Rate (CR, %) Better if closer to 90% — Split CP										
LAC	89.60	98.83	**90.53**	90.47	89.73	**90.03**	99.00	99.10	89.27	89.10
APS	99.73	98.90	99.07	99.20	98.90	99.00	100.00	99.97	100.00	98.50
SPUQ-adapt.	90.37	89.77	91.27	**90.20**	**89.87**	89.77	**90.90**	91.13	**89.80**	**89.73**
TRON-adapt.	**89.77**	90.40	**90.53**	90.67	89.80	90.20	90.93	91.53	89.53	89.70
PA	90.93	**89.90**	91.30	91.47	90.47	90.20	92.70	**90.40**	95.13	90.47
Coverage Rate (CR, %) Better if closer to 90% — QCCP										
LAC	94.47	95.13	92.23	93.67	91.87	91.87	97.80	96.40	94.47	94.23
APS	97.10	96.90	98.63	97.50	98.60	99.17	93.10	94.33	97.10	95.73
SPUQ-adapt.	91.23	**90.43**	91.67	**90.57**	90.67	**90.23**	91.67	91.23	**90.00**	**90.37**
TRON-adapt.	91.03	90.93	**90.70**	91.10	**90.07**	90.27	**91.07**	91.47	90.03	90.47
PA	**91.00**	90.80	91.60	91.83	92.00	91.83	92.90	**90.40**	95.63	91.17
Set Size (SS) ↓ — Split CP										
LAC	2.18	5.62	1.56	1.80	1.66	1.76	5.72	5.72	2.52	2.73
APS	5.72	5.64	5.57	5.61	5.60	5.61	6.00	5.97	5.97	5.65
SPUQ-adapt.	2.48	2.65	1.65	1.77	1.84	1.79	3.29	3.32	2.68	2.73
TRON-adapt.	2.17	2.58	1.57	1.83	1.67	1.77	3.38	3.37	2.57	2.82
PA	**2.05**	**2.19**	**1.25**	**1.25**	**1.44**	**1.37**	**1.55**	**1.36**	**1.00**	**1.08**
Set Size (SS) ↓ — QCCP										
LAC	3.58	4.28	2.47	3.11	2.61	2.68	5.03	4.74	4.27	4.30
APS	4.59	4.78	4.79	4.73	4.89	5.00	4.45	4.68	5.03	4.83
SPUQ-adapt.	2.66	2.83	1.88	1.96	2.18	2.15	3.37	3.32	2.85	2.92
TRON-adapt.	2.45	2.84	1.77	2.05	1.99	2.08	3.37	3.35	2.79	3.01
PA	**2.29**	**2.49**	**1.44**	**1.48**	**1.81**	**1.76**	**1.59**	**1.41**	**1.24**	**1.32**

calibration, and test inputs are original human-written questions. It provides a reference for coverage and prediction-set size for the two following scenarios. In the **fully reworded** setting, we apply the same paraphrasing operator $\mathcal{T}$ (introduced above) to all inputs across training, calibration, and test sets, and compute all scores on reworded questions. This tests how the LLM paraphrased input can influence the stability of different nonconformity scores. All CP scores are computed on these LLM-reworded questions. In the **semi-reworded** setting, training and calibration use original inputs, while test inputs are reworded by an LLM. This represents deployment in which users or an upstream system rephrase inputs after calibration, creating a structured distribution shift. Normal and Fully reworded retain matched calibration and test distributions, so standard conformal validity applies under the corresponding exchangeability assumptions. Semi-reworded tests whether coverage and efficiency persist across the rewording shift, with theoretical validity governed by the additional conditions stated in our analysis. Additional details are provided in Appendix E.3.

6.1 Stable Coverage and Compact Sets under Rewording

Normal and Fully Reworded Settings. We first evaluate whether the proposed PA score remains robust under prompt rewording, where each test question is rephrased using the procedure described above. Results on `Qwen-3-8B` across five MCQA benchmarks are summarized in Table 3. Across both normal and fully reworded settings, PA consistently achieves the smallest prediction sets among all methods, while maintaining coverage close to the nominal 90% target. In particular, PA's coverage is typically slightly conservative (ranging from 89.90% to 95.63%), but avoids the severe over-coverage exhibited by APS (93.10%–100.00%). Compared to methods such as TRON and SPUQ, which can be closer to the nominal target, PA achieves substantially smaller set sizes, often reducing them by a factor of $2\times$ or more. Overall, PA achieves a more favorable coverage–efficiency tradeoff, delivering compact prediction sets without sacrificing reliability too much.

Semi-reworded Setting. To further assess robustness under distribution shifts induced by paraphrasing, we evaluate all methods in the semi-reworded (Semi.) setting (Table 4). This setting is more challenging, as the calibration and test distributions no longer match. Under this shift, PA continues to produce the most compact prediction sets in most cases, while maintaining competitive coverage. Although TRON and SPUQ are occasionally closer to the nominal 90% target, they consistently require larger sets. Meanwhile, APS remains overly conservative, and LAC exhibits instability in several benchmarks. These fixed-α comparisons describe empirical robustness and compactness, so

Table 4: Performance under the **semi-reworded (Semi.)** setting. Bold numbers indicate the best.

	Method	QA	RC	CI	DRS	DS	QA	RC	CI	DRS	DS
	Semi.	Coverage Rate (CR, %) Better if closer to 90%					Set Size (SS) ↓				
Split CP	LAC	89.83	**88.60**	89.20	99.10	88.23	2.54	1.62	1.69	5.72	2.61
	APS	99.47	98.90	99.00	100.00	100.00	5.74	5.57	5.61	6.00	5.97
	SPUQ-adapt.	84.83	87.93	87.30	86.23	88.20	**2.21**	1.53	1.60	2.72	2.49
	TRON-adapt.	**90.10**	**88.60**	**89.23**	**91.40**	**88.70**	2.55	1.62	1.69	3.36	2.66
	PA	92.07	92.20	93.23	95.90	94.23	2.26	**1.31**	**1.56**	**1.70**	**1.00**
QCCP	LAC	92.13	**90.40**	89.43	96.97	91.60	3.43	2.43	2.28	5.07	3.70
	APS	95.00	96.97	97.60	93.57	94.53	4.19	4.15	4.61	4.67	4.47
	SPUQ-adapt.	87.10	88.27	87.90	88.10	88.07	2.40	1.81	1.81	2.99	2.57
	TRON-adapt.	88.77	89.17	88.53	91.57	**88.80**	2.55	1.90	1.90	3.49	2.75
	PA	**89.50**	90.67	**89.53**	**91.10**	94.20	**2.01**	**1.39**	**1.36**	**1.70**	**1.11**

Table 5: Cross-LLM generalization on MMLU. Coverage Rate (CR, %) closer to 90% is better; Set Size (SS) ↓ is better. Results are shown for normal, semi-reworded (Semi.), and fully reworded (Full).

		`Llama-3.1-8B`						`Phi-3-small`					
		CR (%)			SS			CR (%)			SS		
	Method	Normal	Semi.	Full	Normal	Semi.	Full	Normal	Semi.	Full	Normal	Semi.	Full
Split CP	LAC	88.97	86.50	88.87	2.63	2.75	2.95	91.03	**88.67**	89.03	5.65	5.55	5.57
	APS	95.87	94.90	94.93	3.81	3.97	3.98	99.53	99.67	99.67	5.98	5.98	5.98
	SPUQ-adapt.	88.50	85.40	89.63	2.68	**2.69**	3.04	90.33	84.47	90.13	5.61	5.38	5.61
	TRON-adapt.	89.00	86.53	88.93	2.63	2.75	2.95	**90.00**	87.17	89.37	5.60	5.49	5.58
	PA	**89.50**	**90.03**	**89.87**	**2.52**	2.70	**2.75**	90.67	91.50	**90.07**	**3.59**	**3.63**	**3.62**
QCCP	LAC	90.03	87.77	89.17	2.85	2.90	3.16	91.83	87.37	91.73	5.67	5.51	5.67
	APS	95.47	94.40	94.43	3.91	3.99	4.08	94.77	97.17	93.73	5.80	5.88	5.77
	SPUQ-adapt.	89.43	87.20	**89.90**	2.92	2.85	3.23	91.17	84.73	91.07	5.65	5.40	5.64
	TRON-adapt.	**89.97**	87.77	89.13	2.84	2.90	3.16	91.07	86.50	**90.60**	5.64	5.49	5.62
	PA	89.57	**89.63**	90.40	**2.49**	**2.60**	**3.02**	**90.33**	**90.67**	90.73	**3.57**	**3.60**	**3.65**

achieved coverage may differ across methods. Table 20 in the appendix gives a separate held-out, approximately coverage-matched comparison.

Generalization Across LLMs. Finally, we evaluate cross-model generalization using two models, `Llama-3.1-8B-Instruct` [9] and `Phi-3-small-8k-Instruct` [1], on MMLU under the normal, semi-reworded, and fully reworded settings. The results mirror the patterns previously: APS tends to inflate coverage, often overshooting the nominal 90% level, while LAC, SPUQ, and TRON can be closer to nominal coverage in some cases but generally require larger prediction sets. In contrast, our score (PA) consistently produces the most compact prediction sets and maintains competitive coverage close to the nominal target.

Extensions to Other Settings. Due to space constraints, many additional experimental results are deferred to Appendix E. We provide a brief summary here. First, we observe similar patterns on the remaining two medical MCQA datasets, where our method consistently produces smaller prediction sets while achieving coverage closer to the target level than competing methods. Second, our method is versatile and extends naturally to other settings. Extensions to API-only scenarios, other tasks (e.g., factuality verification), and image modalities are discussed in Appendix G in greater detail.

7 In-Depth Analysis and Ablation Studies

Proxy Model Improves Accuracy and Calibration. In our work, the proxy model P_θ is a lightweight two-layer MLP trained on top of frozen hidden states from the LLM. We find that P_θ produces more accurate and better-calibrated distributions than raw LLM logits. As shown in Figure 3, the proxy consistently outperforms logits across benchmarks on all calibration metrics. For example, accuracy improves significantly (**DS** $0.597 \rightarrow 0.944$, **DRS** $0.510 \rightarrow 0.742$), while both Brier score and NLL decrease (**RC** Brier $0.075 \rightarrow 0.039$, NLL $2.513 \rightarrow 0.487$). Here, Brier measures the mean squared error between predicted probabilities and true one-hot labels, while NLL penalizes assigning low probability to the correct answer; both capture calibration quality. This improvement arises because raw logits are optimized for next-token prediction rather than calibrated posteriors, often leading to miscalibration and poor class separation. In contrast, the proxy leverages hidden states, which encode richer task-relevant signals, and is trained with a calibration-aware loss. As a result, it achieves both higher accuracy and better calibration.

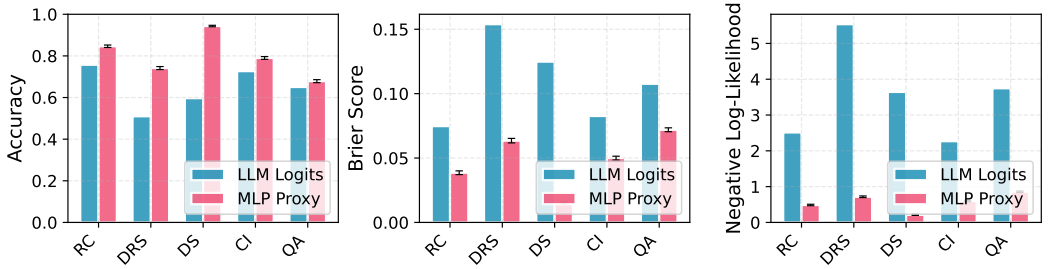


Figure 3: The proxy improves both task performance and calibration consistently. Evaluation of prediction accuracy and calibration metric using proxy model vs. raw LLM logits across datasets. Left: accuracy (higher is better). Middle: Brier (lower is better). Right: NLL (lower is better).

Table 6: Factorized attribution on `Qwen3-8B` (five datasets, three seeds; equal-weight mean over 15 dataset–seed results). Rewrite Acc. is label accuracy on rewritten inputs. Semi CR and SS are empirical coverage (%) and average set size under split CP in the semi-reworded setting at $\alpha = 0.1$. Set change is the mean number of labels whose membership differs between the prediction sets for an original input and its rewrite, $|\widehat{C}(x) \triangle \widehat{C}(\mathcal{T}(x))|$. The Full PA and augmented-proxy single-view rows use the same trained proxy checkpoint; only inference-time aggregation differs.

Score and proxy training	Inference	Rewrite Acc. ↑	Semi CR	Semi SS ↓	Set change ↓
Raw probabilities; no proxy	Single view	65.74	91.36	2.8929	0.3561
Raw-probability proxy; Orig-only	Single view	70.16	89.74	2.2032	0.2369
Hidden-state proxy; Orig-only	Single view	79.80	89.58	1.3725	0.2009
Hidden-state proxy; augmented	Single view	80.38	89.96	1.3587	0.2029
Full PA; same augmented proxy	Six-view S_{mean}	80.34	89.81	1.3331	0.1708
Raw probabilities; no proxy	Six-view mean	65.53	86.11	2.0394	0.5785

Where Do the Gains Come From? We ask how much of PA's improvement comes from learning a proxy and what paraphrases add during training and inference. Table 6 first compares raw probabilities with proxies trained on either probability vectors or hidden states. *Orig-only* uses one copy of each original training question; *augmented* adds up to six meaning-preserving paraphrases per question, retaining its label. We then compare single-view and six-view inference using the same augmented proxy checkpoint, isolating the effect of aggregation. The hidden-state proxy accounts for most of the compactness gain, reducing Semi SS from 2.8929 to 1.3725 without paraphrase training. Augmentation improves rewritten-input accuracy from 79.80% to 80.38% and reduces true-label score drift from 0.0596 to 0.0574. With this checkpoint fixed, six-view aggregation reduces set change by 15.8% and Semi SS from 1.3587 to 1.3331, with coverage changing from 89.96% to 89.81%. Raw-probability averaging instead increases set change from 0.3561 to 0.5785. These comparisons distinguish proxy-driven compactness from the additional effects of paraphrase training and aggregation. Appendix F further controls for augmentation's larger training budget by repeating original questions to match training rows and optimizer updates.

Better Class-conditional Coverage. We group questions into semantic classes (or clusters) and evaluate class-conditional coverage by visualizing empirical coverage rates on different classes. As shown in Figure 4, PA consistently tracks the 90% target across different classes, while LAC and APS overshoot and the TRON / SPUQ baselines exhibit larger per-cluster swings.

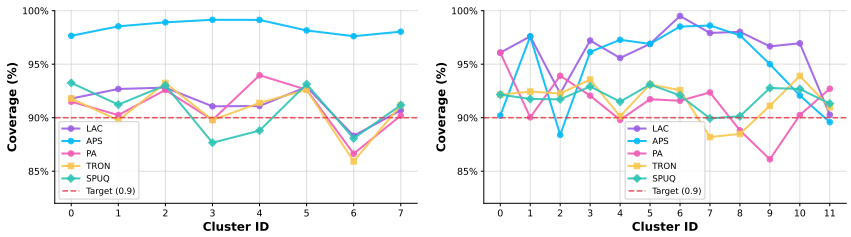


Figure 4: Evaluation of class-conditional coverage with QCCP on the CosmosQA dataset (left) and HaluDial dataset (right). Results on additional datasets are provided in Figure 9.

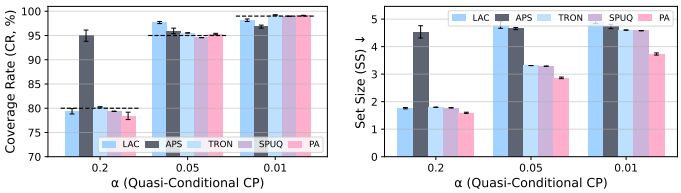


Figure 5: Coverage rate (left) and prediction set size (right) vs. different values of the level α on MMLU under QCCP.

Table 7: Effect of paraphrase budget m under split CP and QCCP.

	m=2	m=4	m=6
Coverage Rate (CR, %)			
PA (marginal)	89.77	89.67	**90.53**
PA (conditional)	90.40	90.40	**90.93**
Set Size (SS)			
PA (marginal)	1.92	**1.86**	1.88
PA (conditional)	2.19	2.16	**2.15**

Effect of Coverage Levels. To examine the effect of the level α, we vary $\alpha \in \{0.2, 0.05, 0.01\}$ and evaluate both coverage and prediction set size again. As shown in Figure 5, our PA score consistently tracks the nominal $(1-\alpha)$ target while maintaining the most compact sets. For example, at $\alpha = 0.05$, it achieves $\approx 95\%$ coverage with an average set size of 2.9, compared to LAC's inflated 4.8 and APS's 4.7. APS dramatically overshoots at $\alpha = 0.2$ (95% coverage with set size 4.5 vs. an 80% target), while LAC, TRON, and SPUQ track the target reasonably well at small α but produce strictly larger sets than PA at every level. Overall, PA consistently outperforms across different levels.

Larger Paraphrase Budgets Yield Smaller Prediction Sets. We now study the effect of the paraphrase budget m for `Qwen3-8B` [51] on the MMLU dataset. As shown in Table 7, increasing $m \in \{2, 4, 6\}$ leads our PA score to achieve higher coverage and smaller set sizes. This is intuitive: more paraphrases provide richer semantic views of each question, which reduces variance in the aggregated score and enables more precise calibration.

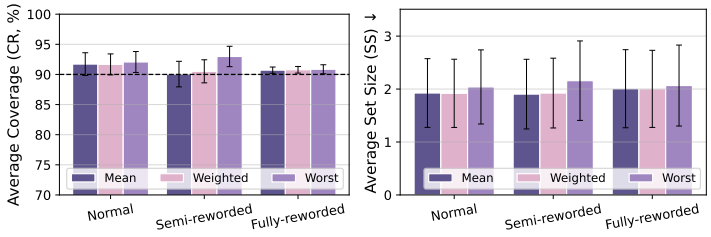


Figure 6: Comparison of different scores under QCCP.

Comparison of Different Scores. Finally, we compare the three PA scores, namely Mean, Weighted, and Worst, introduced in Section 4. We evaluate them under both split CP and QCCP, averaging across the seven datasets at each rewording setting (Normal, Semi-reworded, Fully-reworded). As shown in Figure 6, all three variants track the 90% coverage target across the three settings: under QCCP, Mean averages 91.7%/90.1%/90.7% and Weighted 91.7%/90.6%/90.8% on Normal / Semi-reworded / Fully-reworded prompts respectively, while Worst is mildly conservative at 92.1%/93.0%/90.9%. Mean and Weighted are nearly indistinguishable in practice (within ≈ 0.5 pp on coverage and ≈ 0.02 on set size) and yield the most compact prediction sets (avg SS $\approx$ 1.9–2.0); Worst trades a small coverage margin for $\approx$ 5–15% larger sets (avg SS $\approx$ 2.0–2.2). Split CP gives near-identical averages here, preserving the same relative ranking. Overall, the three variants span a smooth efficiency–conservatism tradeoff: Mean and Weighted are the tightest while still respecting the nominal target, and Worst supplies an additional coverage margin when extra robustness is desired.

8 Conclusion

In this work, we introduced a framework for paraphrase-robust conformal prediction through paraphrase-aware nonconformity scores applied to both split CP and quasi-conditional CP. Under the stated exchangeability conditions, our method inherits conformal coverage guarantees while producing compact prediction sets. The learned proxy drives most of the compactness gain, while paraphrase training and inference-time aggregation improve wording stability. Experiments on five general QA and two medical QA benchmarks demonstrate robustness under rewording and strong generalization across model families. These findings motivate further study of robust nonconformity score design across tasks and modalities, with preliminary claim-verification results and a vision–language extension discussed in Appendices G.3 and G.4.

Acknowledgments and Disclosure of Funding

This work was supported in part by grant R01-MH143267 and R01EB036016. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH. We thank the anonymous reviewers for their constructive feedback. We also thank Jiancong Xiao and various people at the student paper swap event (Weiqiu You, Helen Jin, Seewon Choi, and others) for their helpful comments and discussions.

References

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL `https://arxiv.org/abs/2404.14219`.

[2] Oleksandr Balabanov and Hampus Linander. Uncertainty quantification in fine-tuned LLMs using loRA ensembles. In *ICLR Workshop: Quantify Uncertainty and Hallucination in Foundation Models: The Next Frontier in Reliable AI*, 2025. URL `https://openreview.net/forum?id=L8KaEqT70q`.

[3] Neil Band, Xuechen Li, Tengyu Ma, and Tatsunori Hashimoto. Linguistic calibration of long-form generations. In *International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=rJVjQSQ8ye`.

[4] Dylan Bouchard and Mohit Singh Chauhan. Uncertainty quantification for language models: A suite of black-box, white-box, llm judge, and ensemble scorers. *arXiv preprint arXiv:2504.19254*, 2025.

[5] John Cherian, Isaac Gibbs, and Emmanuel Candes. Large language model validity via enhanced conformal prediction methods. In *Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=JD3NYpeQ3R`.

[6] Kyle Cox, Jiawei Xu, Yikun Han, Rong Xu, Tianhao Li, Chi-Yang Hsu, Tianlong Chen, Walter Gerych, and Ying Ding. Mapping from meaning: Addressing the miscalibration of prompt-sensitive language models. In *AAAI Conference on Artificial Intelligence*, 2025.

[7] Gianluca Detommaso, Martin Bertran, Riccardo Fogliato, and Aaron Roth. Multicalibration for confidence scoring in LLMs. In *Proceedings of the 41st International Conference on Machine Learning*, International Conference on Machine Learning. JMLR.org, 2024.

[8] Jinhao Duan, Hao Cheng, Shiqi Wang, et al. Shifting attention to relevance: Towards predictive uncertainty quantification of free-form large language models. In *ACL*, 2024. URL `https://aclanthology.org/2024.acl-long.276/`.

[9] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The Llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024.

[10] Bat-Sheva Einbinder, Yaniv Romano, Matteo Sesia, and Yanfei Zhou. Training uncertainty-aware classifiers with conformalized deep learning. *Advances in neural information processing systems*, 35:22380–22395, 2022.

[11] Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482, 2021.

[12] Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. SPUQ: Perturbation-based uncertainty quantification for large language models. In *Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2336–2346, 2024.

[13] Isaac Gibbs, John J Cherian, and Emmanuel J Candès. Conformal prediction with conditional guarantees. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, page qkaf008, 2025.

[14] Yu Gui, Ying Jin, and Zhimei Ren. Conformal alignment: Knowing when to trust foundation models with guarantees. In *Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=YzyCEJlV9Z`.

[15] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=d7KBjmI3GmQ`.

[16] Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1243. URL `https://aclanthology.org/D19-1243/`.

[17] Xinmeng Huang, Shuo Li, Mengxin Yu, Matteo Sesia, Hamed Hassani, Insup Lee, Osbert Bastani, and Edgar Dobriban. Uncertainty in language models: Assessment through rank-calibration. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 284–312, 2024.

[18] Yukun Huang, Yixin Liu, Raghuveer Thirukovalluru, Arman Cohan, and Bhuwan Dhingra. Calibrating long-form generations from large language models. In *Findings of the Association for Computational Linguistics: EMNLP*, 2024.

[19] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

[20] Archit Karandikar, Nicholas Cain, Dustin Tran, Balaji Lakshminarayanan, Jonathon Shlens, Michael C Mozer, and Becca Roelofs. Soft calibration objectives for neural networks. *Advances in Neural Information Processing Systems*, 34:29768–29779, 2021.

[21] Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *arXiv preprint arXiv:2406.15927*, 2024. URL `https://arxiv.org/abs/2406.15927`.

[22] Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. Conformal prediction with large language models for multi-choice question answering. *arXiv preprint arXiv:2305.18404*, 2023. URL `https://arxiv.org/abs/2305.18404`.

[23] Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.397. URL https://aclanthology.org/2023.emnlp-main.397/.

[24] L López, Shaza Elsharief, Dhiyaa Al Jorf, Firas Darwish, Congbo Ma, and Farah E Shamout. Uncertainty quantification for machine learning in healthcare: A survey. *arXiv preprint arXiv:2505.02874*, 2025.

[25] Nishanth Madhusudhan, Sathwik Tejaswi Madhusudhan, Vikas Yadav, and Masoud Hashemi. Do LLMs know when to NOT answer? investigating abstention abilities of large language models. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9329–9345, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.627/.

[26] Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. In *International Conference on Machine Learning*, volume 235, pages 36029–36047, 2024.

[27] Zachary Nado, Neil Band, Mark Collier, Alexander D'Amour, Josip Djolonga, Sebastian Farquhar, Andrew Foong, Alex Kendall, Andrey Malinin, Daniel Muñoz, et al. Uncertainty baselines: Benchmarks for uncertainty & robustness in deep learning. *arXiv preprint arXiv:2106.04015*, 2022. URL https://arxiv.org/abs/2106.04015.

[28] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR, 2022.

[29] Xin Qiu and Risto Miikkulainen. Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic space. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=LOH6qzI7T6.

[30] Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal language modeling. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=pzUhfQ74c5.

[31] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

[32] Yaniv Romano, Matteo Sesia, and Emmanuel Candes. Classification with valid and adaptive coverage. *Advances in neural information processing systems*, 33:3581–3591, 2020.

[33] Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525):223–234, 2019.

[34] Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9:371–421, 2008.

[35] Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Comput. Surv.*, 58(3), September 2025. ISSN 0360-0300. doi: 10.1145/3744238. URL https://doi.org/10.1145/3744238.

[36] Chenglei Si, Chen Zhao, Sewon Min, and Jordan Boyd-Graber. Re-examining calibration: The case of question answering. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2814–2829, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.204. URL https://aclanthology.org/2022.findings-emnlp.204/.

[37] Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. API is enough: Conformal prediction for large language models without logit-access. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 979–995, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.54. URL https://aclanthology.org/2024.findings-emnlp.54/.

[38] Sina Tayebati, Divake Kumar, Nastaran Darabi, et al. Learning conformal abstention policies for adaptive risk management in large language and vision-language models. *arXiv preprint arXiv:2502.06884*, 2025. URL https://arxiv.org/abs/2502.06884.

[39] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, 2018.

[40] Dennis Ulmer, Jes Frellsen, and Christian Hardmeier. Exploring predictive uncertainty and calibration in NLP: A study on the impact of method & data scarcity. In *Findings of EMNLP*, 2022. URL https://aclanthology.org/2022.findings-emnlp.198/.

[41] Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. Calibrating large language models using their generations only. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.824. URL https://aclanthology.org/2024.acl-long.824/.

[42] Harit Vishwakarma, Alan Mishler, Thomas Cook, Niccolo Dalmasso, Natraj Raman, and Sumitra Ganesh. Prune 'n predict: Optimizing LLM decision-making with conformal prediction. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=5g6LPR0Dlx.

[43] Vladimir Vovk. Conditional validity of inductive conformal predictors. In *Asian conference on machine learning*, pages 475–490. PMLR, 2012.

[44] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

[45] Qingni Wang, Tiantian Geng, Zhiyuan Wang, Teng Wang, Bo Fu, and Feng Zheng. Sample then identify: A general framework for risk control and assessment in multimodal large language models. *arXiv preprint arXiv:2410.08174*, 2024.

[46] Qingni Wang, Tiantian Geng, Zhiyuan Wang, Teng Wang, Bo Fu, and Feng Zheng. Sample then identify: A general framework for risk control and assessment in multimodal large language models. In *International Conference on Learning Representations*, 2025.

[47] Yibin Wang, Haizhou Shi, Ligong Han, Dimitris N. Metaxas, and Hao Wang. BLob: Bayesian low-rank adaptation by backpropagation for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=MaDykgj4Ru.

[48] Zhiyuan Wang, Jinhao Duan, Lu Cheng, Yue Zhang, Qingni Wang, Xiaoshuang Shi, Kaidi Xu, Heng Tao Shen, and Xiaofeng Zhu. ConU: Conformal uncertainty in large language models with correctness coverage guarantees. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6886–6898, 2024.

[49] Zhiyuan Wang, Jinhao Duan, Qingni Wang, et al. Coin: Uncertainty-guarding selective question answering for foundation models with provable risk guarantees. *arXiv preprint arXiv:2506.20178*, 2025. URL https://arxiv.org/abs/2506.20178.

[50] Zhiyuan Wang, Qingni Wang, Yue Zhang, Tianlong Chen, Xiaofeng Zhu, Xiaoshuang Shi, and Kaidi Xu. SConU: Selective conformal uncertainty in large language models, 2025. URL https://arxiv.org/abs/2504.14154.

[51] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[52] Guang Yang and Xinyang Liu. Conformal sets in multiple-choice question answering under black-box settings with provable coverage guarantees. *arXiv preprint arXiv:2508.05544*, 2025. URL https://arxiv.org/abs/2508.05544.

[53] Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F. Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. Benchmarking LLMs via uncertainty quantification. In *Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=LOoSfTroNE.

[54] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, 2019.

Appendix

Table of Contents

A Detailed Related Work **16**

B Brief Introduction to the Conformal Framework **17**

C Coverage Guarantees: Statements and Proofs **18**
C.1 Proofs for Score Exchangeability 18
C.2 Formal Coverage Results 20
C.3 Approximate Distributional Closedness 21

D Experimental Details **22**
D.1 Dataset Details 22
D.2 Paraphrase Generation for Full/Semi-Paraphrase Settings 22
D.3 Baseline Score Adaptations 23
D.4 Experimental Implementation Details 23
D.5 Hyperparameters for Reproducibility 24

E Additional Experimental Results **24**
E.1 Better Performance on Medical MCQA Datasets 24
E.2 Finite-Sample Behavior of Class-Conditional Coverage 25
E.3 Robustness under the Semi-Reworded Setting 27
E.4 Quantifying Coverage Deviations in the Semi-Reworded Setting 28
E.5 Semantic Consistency of Generated Paraphrases 29
E.6 End-to-End Runtime 30

F Analysis of the Proxy Model **30**
F.1 Different Class-Conditional Levels for QCCP 33
F.2 Different α-Level for Split CP 33
F.3 Different Paraphrase Generators 34
F.4 Different Embedding Choices 35
F.5 Different Proxy Training and Calibration Budgets 35
F.6 Different Aggregation Scoring Rules 36

G Scalability and Broader Applicability **37**
G.1 Extension to Black-Box / API-Only Settings 37
G.2 Extension to Larger Backbone LLMs 38
G.3 Extension to Factuality Verification Tasks 38
G.4 Extension to Image Modality 39

H Limitations and Broader Impacts **40**

A Detailed Related Work

Heuristic and Calibration-Based Uncertainty for LLMs. LLM uncertainty is often estimated from token-level signals such as entropy, logits, or ranks, followed by calibration. Early work studies calibration in deep models and NLP tasks [27, 40, 36], and extends to prompt- and generation-level schemes for long-form outputs [3, 18]. More recent approaches replace raw probabilities with representation-based surrogates, including semantic entropy probes [21], relevance-aware confidence [8], perturbation-based measures [12], multi-agent diversity signals [45], and semantic-density

metrics [29]. Complementary directions include abstention mechanisms [25], multicalibration of confidence scores [7], post-hoc calibration from generated text [41], and parameter-efficient Bayesian or ensemble-style methods for fine-tuned LLMs [2, 47]. While perturbation-based methods such as SPUQ explicitly aggregate across input variations, these methods address instance-level uncertainty. We study label-wise scores and prediction-set stability within CP, whose coverage guarantee additionally requires exchangeability.

Conformal Prediction for LLMs. Conformal prediction (CP) provides distribution-free prediction sets with finite-sample coverage under exchangeability [44, 34]. Recent work adapts CP to language modeling in several directions, including conformal language modeling [30, 30], API-only inference without logit access [37], and methods tailored to multiple-choice settings [22, 52]. CP has also been used for output alignment and certification [14], as well as benchmarking LLMs using uncertainty-aware metrics [53]. Extensions to generation tasks include `ConU`, which applies split CP to sets of sampled responses [48], and `SConU`, which studies exchangeability violations and approximates conditional guarantees [50]. Other approaches combine CP with re-asking strategies to improve accuracy and efficiency [42], while selective answering with risk control has been explored via conformal abstention [38, 49]. In contrast to these works, our approach focuses on *semantic invariance*: we develop paraphrase-aware scores that capture stability under meaning-preserving transformations, and integrate them with both split conformal prediction and quasi-conditional calibration on semantic embeddings, going beyond standard split CP.

Toward Conditional Guarantees and Paraphrase Robustness. Exact conditional coverage is unattainable in finite samples without distributional assumptions [43, 11], motivating relaxations such as *quasi-conditional* guarantees via augmented quantile regression [13]. Recent work extends CP to leverage feature-conditional structure in LLM settings [5]. In parallel, another line of work studies *semantic uncertainty*, introducing embedding-based metrics and perturbation procedures [12, 21, 17]. Related approaches examine prompt sensitivity and meaning-preserving perturbations [29, 6], but do not provide conformal guarantees. Our method bridges these directions by (i) quantifying predictive stability across paraphrases using semantic embeddings and a proxy model, and (ii) calibrating these scores within the conformal prediction framework, yielding prediction sets that are both semantically robust and statistically valid.

B Brief Introduction to the Conformal Framework

Theoretical guarantee of Split CP. We briefly review the standard coverage guarantee of split conformal prediction (CP), which serves as the foundation for our method. This result establishes finite-sample, distribution-free coverage under exchangeability.

Lemma B.1 ([44])**.** *Assume $(X_i, Y_i)_{i=1}^{n+1}$ are i.i.d. and that $S(X, Y)$ has a continuous distribution. The prediction set from split CP is*

$$\widehat{C}_{\mathrm{CP}}(X_{n+1}) := \{y : S(X_{n+1}, y) \leq Q_\alpha(\{S(X_i, Y_i)\}_{i=1}^n)\}$$

where $Q_\alpha(\{S(X_i, Y_i)\}_{i=1}^n)$ is the empirical $(1-\alpha)$ quantile of $\{S(X_i, Y_i)\}_{i=1}^n \cup \{\infty\}$. Then, it achieves marginal coverage:

$$1 - \alpha \;\leq\; \mathbb{P}(Y_{n+1} \in \widehat{C}_{\mathrm{CP}}(X_{n+1})) \;\leq\; 1 - \alpha + \frac{1}{n+1}. \tag{4}$$

Theoretical Guarantee of Quasi-Conditional CP. Lemma B.2 establishes the quasi-conditional coverage guarantee for QCCP. This guarantee is strictly stronger than the marginal coverage of split CP (which corresponds to the case where $\mathcal{F}$ contains only constant functions) and strictly weaker than full conditional coverage (which would require $\mathcal{F}$ to be the class of all measurable functions), thereby providing a principled middle ground.

In QCCP, the prediction set is constructed using an input-dependent threshold function $\widehat{g}_S$ learned from calibration scores via quantile regression (cf. (2)). We emphasize that $\widehat{g}_S$ depends on the choice of nonconformity score S, as it is trained on the induced calibration scores $\{S(X_i, Y_i)\}_{i=1}^n$. For each candidate score value s, the fitted function $\widehat{g}_s : \mathcal{X} \to \mathbb{R}$ is evaluated at X_{n+1}; its dependence on the candidate label y is through $s = S(X_{n+1}, y)$. In practice, computing $\widehat{g}_S$ reduces to a convex optimization; see Section 4 of Gibbs et al. [13] for details.

Algorithm 1 Paraphrase-Robust Quasi-Conditional CP

Input: Proper training set $\mathcal{D}_{\mathrm{tr}} = \{(X_i^{\mathrm{tr}}, Y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$; calibration set $\mathcal{D}_{\mathrm{cal}} = \{(X_i^{\mathrm{cal}}, Y_i^{\mathrm{cal}})\}_{i=1}^{n_{\mathrm{cal}}}$; paraphrase generator; embedding map E; proxy model P_θ; class function ϕ; confidence level α; PA score rule S.
Output: Prediction set $\widehat{C}(x)$ for a test question x.
Train proxy: Fit P_θ only on $\mathcal{D}_{\mathrm{tr}}$ using embeddings $\{(E(X_i^{\mathrm{tr}}), Y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$.
Compute calibration scores: For each $(X_i^{\mathrm{cal}}, Y_i^{\mathrm{cal}}) \in \mathcal{D}_{\mathrm{cal}}$, generate paraphrases, compute the PA score $S_i \leftarrow S(X_i^{\mathrm{cal}}, Y_i^{\mathrm{cal}})$ using the fixed proxy P_θ, and record class $\phi(X_i^{\mathrm{cal}})$.
Fit conformal threshold: Use $\{(S_i, \phi(X_i^{\mathrm{cal}}))\}_{i=1}^{n_{\mathrm{cal}}}$ to fit the QCCP quantile function $\widehat{g}_S$; for split CP, compute the global conformal quantile.
for each test question x **do**
  $\widehat{C}(x) \leftarrow \emptyset$
  Generate a paraphrase set $\mathcal{B}(x)$ and compute embeddings $\{E(x')\}_{x' \in \mathcal{B}(x)}$
  $z \leftarrow \phi(x)$
  for each $y \in \mathcal{Y}$ **do**
    Compute the PA score $S(x, y)$ using the fixed proxy P_θ
    Query $\widehat{g}_z$ for QCCP, or the global quantile for split CP
    if $S(x, y) \le \widehat{g}_z(x)$ **then**
      $\widehat{C}(x) \leftarrow \widehat{C}(x) \cup \{y\}$
    end if
  end for
  return $\widehat{C}(x)$
end for

Lemma B.2 (Theorem 2 in [13]). *Let $\mathcal{F} = \{\phi(\cdot)^\top \beta : \beta \in \mathbb{R}^d\}$ denote the class of linear functions over the feature map $\phi : \mathcal{X} \to \mathbb{R}^d$. Assume that $\{(X_i, Y_i)\}_{i=1}^{n+1}$ are i.i.d., and that $S(X, Y) \mid X$ has a continuous distribution. Define the prediction set as*

$$\widehat{C}_{\mathrm{QCCP}}(X_{n+1}) := \left\{y : S(X_{n+1}, y) \le \widehat{g}_{S(X_{n+1},y)}(X_{n+1})\right\}, \tag{3}$$

where $\widehat{g}_{S(X_{n+1},y)}$ is the fitted quantile regressor obtained by solving (2) *for a given value of $S(X_{n+1}, y)$ and $\ell_\alpha(\theta, S) = (S - \theta)\,(\mathbf{1}_{\{S \ge \theta\}} - \alpha)$ denoting the pinball loss. Then, for any $f \in \mathcal{F}$,*

$$\left|\mathbb{E}\Big[f(X_{n+1})\left(\mathbf{1}\{Y_{n+1} \in \widehat{C}_{\mathrm{QCCP}}(X_{n+1})\} - (1-\alpha)\right)\Big]\right| \le \frac{d}{n+1} \cdot \mathbb{E}\left[\max_{1 \le i \le n+1} |f(X_i)|\right].$$

C Coverage Guarantees: Statements and Proofs

C.1 Proofs for Score Exchangeability

Proposition C.1 (Restatement of Proposition 5.1). *Suppose Asm. 5.1 holds, and let $\{(X_i, Y_i)\}_{i=1}^{n+1}$ be an exchangeable (or i.i.d.) dataset. $(\mathcal{T}(X_{n+1}), Y_{n+1})$ might not be exchangeable (or i.i.d.) with $(X_i, Y_i)_{i=1}^n$ due to the semantic shift induced by $\mathcal{T}$. However, applying $\mathcal{T}$ to all inputs restores exchangeability (or i.i.d.), i.e., $\big(\mathcal{T} \circ \mathcal{T}(X_{n+1}), Y_{n+1}\big)$ is exchangeable (or i.i.d.) with $\big\{(\mathcal{T}(X_i), Y_i)\big\}_{i=1}^n$.*

Proof of Proposition C.1. Let $Z_i = (X_i, Y_i)$. Exchangeability means that $(Z_{\pi(1)}, \ldots, Z_{\pi(n+1)}) \overset{d}{=} (Z_1, \ldots, Z_{n+1})$ for any permutation π. All draws from Markov kernels in this proof are taken to be conditionally independent across coordinates given the original data. Applying the Markov kernel $\mathcal{T}$ independently to each input coordinate preserves exchangeability. Hence, $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n+1}$ is exchangeable.

It remains to relate the last transformed sample to the semi-reworded test sample. Let $U_{n+1} \sim \mathcal{T}(\cdot \mid X_{n+1})$ denote the one-step transformed last coordinate, and let $U'_{n+1} \sim \mathcal{T}(\cdot \mid X_{n+1})$, $V_{n+1} \sim \mathcal{T}(\cdot \mid U'_{n+1})$ denote the two-step transformed last coordinate, using fresh randomness. By Asm. 5.1, for any x,

$$\int \mathcal{T}(A \mid x')\mathcal{T}(dx' \mid x) = \mathcal{T}(A \mid x)$$

for every measurable set $A \subseteq \mathcal{X}$. Thus the two-step kernel $\mathcal{T} \circ \mathcal{T}$ equals $\mathcal{T}$ in distribution:

$$(\mathcal{T} \circ \mathcal{T})(\cdot \mid x) = \mathcal{T}(\cdot \mid x).$$

Consequently, conditional on X_{n+1}, $\mathcal{T} \circ \mathcal{T}(X_{n+1}) \overset{d}{=} \mathcal{T}(X_{n+1})$. Moreover, this replacement does not change the joint law because V_{n+1} has the same conditional distribution as U_{n+1} given X_{n+1} and is generated independently of the remaining transformed coordinates conditional on the original data. Therefore, $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n}, \big(\mathcal{T} \circ \mathcal{T}(X_{n+1}), Y_{n+1}\big)$ has the same joint law as $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n+1}$, which is exchangeable. Hence, $\big(\mathcal{T} \circ \mathcal{T}(X_{n+1}), Y_{n+1}\big)$ is exchangeable with $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n}$.

The i.i.d. case follows as a special case of exchangeability. □

Notations. Fix any data point (X, Y) from either the calibration set or the test set. Our PA score first generates m paraphrased versions of X using a language model. Let $\mathcal{T}(\cdot \mid X)$ denote the distribution of a single paraphrase of X, and let

$$Z^{(1)}, \ldots, Z^{(m)} \overset{\text{i.i.d.}}{\sim} \mathcal{T}(\cdot \mid X).$$

More precisely, for any measurable set $A \subseteq \mathcal{X}^m$, let $\mathcal{T}^{\otimes m}(A \mid x)$ denote the product law of m conditionally independent draws from $\mathcal{T}(\cdot \mid x)$. All paraphrase draws are conditionally independent across data points and across paraphrase indices, given the original data. After paraphrasing, the pair (X, Y) is augmented into $(Z^{(1)}, \ldots, Z^{(m)}, Y)$. Since the calibrated predictor P_θ is trained on an unrelated dataset, it is treated as fixed throughout. Define the augmented feature

$$\widetilde{X} := (Z^{(1)}, \ldots, Z^{(m)}), \qquad \widetilde{\mathcal{T}}(X) := \widetilde{X}.$$

Therefore, the PA score can be written as

$$S_{\mathrm{PA}}(X, Y) \;=\; S(Z^{(1)}, \ldots, Z^{(m)}, Y) \;=\; S(\widetilde{\mathcal{T}}(X), Y) = S(\widetilde{X}, Y) \tag{5}$$

for some deterministic computation rule S. This notation highlights that the PA score simply operates on the augmented data $(\widetilde{X}, Y)$. The exact semi-reworded guarantee below applies to PA scores whose final value is a measurable function only of the internally generated paraphrase collection and the label. This includes the mean and worst-case PA scores. It also includes a weighted PA score if its weights are functions only of the generated paraphrase collection; if the weights additionally depend on the original input as an anchor, then an additional joint-invariance assumption is required.

Lemma C.1 (Exchangeability for augmented data)**.** *Let $\{(X_i, Y_i)\}_{i=1}^{n+1}$ be an i.i.d. dataset and $\{(\widetilde{\mathcal{T}}(X_i), Y_i)\}_{i=1}^{n+1}$ be the augmented data defined above. It follows that*

- ***Normal setting:*** *$(\widetilde{\mathcal{T}}(X_{n+1}), Y_{n+1})$ is i.i.d. with $\{(\widetilde{\mathcal{T}}(X_i), Y_i)\}_{i=1}^{n}$.*
- ***Fully reworded setting:*** *$(\widetilde{\mathcal{T}}(\mathcal{T}(X_{n+1})), Y_{n+1})$ is i.i.d. with $\{(\widetilde{\mathcal{T}}(\mathcal{T}(X_i)), Y_i)\}_{i=1}^{n}$.*
- ***Semi-reworded setting:*** *Under Asm. 5.1, $(\widetilde{\mathcal{T}}(\mathcal{T}(X_{n+1})), Y_{n+1})$ is i.i.d. with $\{(\widetilde{\mathcal{T}}(X_i), Y_i)\}_{i=1}^{n}$.*

With Lemma C.1, we can prove Lemma 5.1.

Proof of Lemma 5.1. By Lemma C.1, the corresponding augmented calibration variables and augmented test variable are i.i.d. in each of the three settings. Applying the same fixed measurable score rule S in (5) to each augmented input–label pair preserves exchangeability. Therefore, the induced calibration and test scores are exchangeable. □

Now, we provide the proof for Lemma C.1 below.

Proof of Lemma C.1. By construction, each entry of the random vector $\widetilde{\mathcal{T}}(X)$ is generated by i.i.d. sampling from $\mathcal{T}(\cdot \mid X)$, and the sampling is independent across data points. Therefore, the transformed dataset $\{(\widetilde{\mathcal{T}}(X_i), Y_i)\}_{i=1}^{n+1}$ is still i.i.d., which proves the first point.

For the second point, by a similar argument, $\{(\widetilde{\mathcal{T}}(\mathcal{T}(X_i)), Y_i)\}_{i=1}^{n+1}$ is i.i.d. because $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{n+1}$ is i.i.d.

For the third point, fix $x \in \mathcal{X}$ and a measurable set $A \subseteq \mathcal{X}^m$. Conditional on $X = x$, the augmented calibration input has law

$$\mathbb{P}\left(\widetilde{\mathcal{T}}(X) \in A \mid X = x\right) = \mathcal{T}^{\otimes m}(A \mid x).$$

For a semi-reworded test input, first draw $X' \sim \mathcal{T}(\cdot \mid x)$ and then draw m internal paraphrases conditionally independently from $\mathcal{T}(\cdot \mid X')$. Therefore,

$$\begin{aligned}\mathbb{P}\left(\widetilde{\mathcal{T}}(\mathcal{T}(X)) \in A \mid X = x\right) &= \int \mathcal{T}^{\otimes m}(A \mid x')\mathcal{T}(dx' \mid x) \\ &= \int \mathcal{T}^{\otimes m}(A \mid x)\mathcal{T}(dx' \mid x) \\ &= \mathcal{T}^{\otimes m}(A \mid x),\end{aligned}$$

where the second equality uses Asm. 5.1. Hence,

$$\mathcal{L}\left(\widetilde{\mathcal{T}}(\mathcal{T}(X)) \mid X = x\right) = \mathcal{L}\left(\widetilde{\mathcal{T}}(X) \mid X = x\right).$$

Averaging over (X, Y) gives

$$\mathcal{L}\left(\widetilde{\mathcal{T}}(\mathcal{T}(X)), Y\right) = \mathcal{L}\left(\widetilde{\mathcal{T}}(X), Y\right).$$

Since all paraphrase draws are conditionally independent across data points, $(\widetilde{\mathcal{T}}(\mathcal{T}(X_{n+1})), Y_{n+1})$ is i.i.d. with $\{(\widetilde{\mathcal{T}}(X_i), Y_i)\}_{i=1}^n$. □

C.2 Formal Coverage Results

Now, we are ready to state and prove the formal coverage results.

Theorem C.1 (Formal coverage guarantee)**.** *Assume $(X_i, Y_i)_{i=1}^{n+1}$ are i.i.d. and that, in the corresponding data setting, both $S_{\mathrm{PA}}(U_{n+1}, Y_{n+1}) \mid U_{n+1}$ and $S_{\mathrm{PA}}(U_{n+1}, Y_{n+1})$ have continuous distributions. Let $\mathcal{T}$ be an operator satisfying Assumption 5.1. We consider the following three data settings:*

- ***Normal setting:*** *The calibration dataset is $\{(X_i, Y_i)\}_{i=1}^n$ and the test data is (X_{n+1}, Y_{n+1}).*
- ***Fully reworded setting:*** *The calibration dataset is $\{(\mathcal{T}(X_i), Y_i)\}_{i=1}^n$ and the test data is $(\mathcal{T}(X_{n+1}), Y_{n+1})$.*
- ***Semi-reworded setting:*** *The calibration dataset is $\{(X_i, Y_i)\}_{i=1}^n$ and the test data is $(\mathcal{T}(X_{n+1}), Y_{n+1})$.*

For notational clarity, let U_i denote the effective input on which the PA score is evaluated:

$$U_i = \begin{cases} X_i, & \textit{normal setting}, \\ \mathcal{T}(X_i), & \textit{fully reworded setting}, \\ X_i, & \textit{semi-reworded setting and } i \le n, \\ \mathcal{T}(X_{n+1}), & \textit{semi-reworded setting and } i = n+1. \end{cases}$$

Thus the calibration scores are computed as $S_{\mathrm{PA}}(U_i, Y_i)$ for $i = 1, \ldots, n$, and the test score is computed as $S_{\mathrm{PA}}(U_{n+1}, Y_{n+1})$. For the above three data settings, the following coverage guarantees hold.

- ***PA score for CP.*** *Let $\widehat{C}_{\mathrm{CP}}(\cdot)$ denote the prediction set produced by CP using the PA score and defined in* (1)*. It follows that*

$$1 - \alpha \;\le\; \mathbb{P}(Y_{n+1} \in \widehat{C}_{\mathrm{CP}}(U_{n+1})) \;\le\; 1 - \alpha + \frac{1}{n+1}.$$

- ***PA score for QCCP.*** *Let $\mathcal{F} = \{\phi(\cdot)^\top \beta : \beta \in \mathbb{R}^d\}$ denote the class of linear functions over the feature vector ϕ. Let $\widehat{C}_{\mathrm{QCCP}}(\cdot)$ be the prediction set produced by QCCP using the PA score and defined in* (2)*. Then, for any $f \in \mathcal{F}$,*

$$\left|\mathbb{E}\left[f(W_{n+1})\left(\mathbf{1}\{Y_{n+1} \in \widehat{C}_{\mathrm{QCCP}}(U_{n+1})\} - (1-\alpha)\right)\right]\right| \le \frac{d}{n+1} \cdot \mathbb{E}\left[\max_{1 \le i \le (n+1)} |f(W_i)|\right].$$

Proof of Theorem C.1. By Lemma C.1, the PA scores $\{S_{\mathrm{PA}}(U_i, Y_i)\}_{i=1}^{n+1}$ are exchangeable in each of the three settings. The split-CP statement therefore follows directly from Lemma B.1.

For QCCP, let W_i denote the feature input (i.e., the input of ϕ) used by QCCP in the corresponding setting, and assume that the pairs. In the normal and fully reworded settings, we have $W_i = U_i$, and the same feature construction is applied to all points. In the semi-reworded setting, it holds if the QCCP feature is essentially $\mathcal{T}(U_i)$. In both cases, we have $\{(W_i, S_{\mathrm{PA}}(U_i, Y_i))\}_{i=1}^{n+1}$ is exchangeable (from Proposition C.1), giving exactly the joint exchangeability condition required by Lemma B.2. Applying Lemma B.2 yields the stated bound. □

C.3 Approximate Distributional Closedness

Assumption 5.1 requires the paraphrase kernel $\mathcal{T}$ to be distributionally closed, i.e., $\mathcal{T}(\cdot \mid x') = \mathcal{T}(\cdot \mid x)$ for $\mathcal{T}(\cdot \mid x)$-almost every x'. In practice, however, our paraphrase pipeline is input-conditioned and involves stochastic generation with retries and fallbacks, which may violate this condition. To quantify this deviation, we introduce a relaxed version of distributional closedness.

Assumption C.1 (Approximate distributional closedness)**.** *Let $\mathcal{T}$ be a Markov kernel on $\mathcal{X}$. We say $\mathcal{T}$ is ε-approximately distributionally closed if, for any $x \in \mathcal{X}$,*

$$\int d_{\mathrm{TV}}(\mathcal{T}(\cdot \mid x'), \mathcal{T}(\cdot \mid x))\, \mathcal{T}(dx' \mid x) \le \varepsilon.$$

This assumption captures the extent to which paraphrase distributions vary within the support of $\mathcal{T}(\cdot \mid x)$.

Proposition C.2 (Coverage under approximate closedness)**.** *Let $\widehat{C}(x)$ be the prediction set constructed via conformal prediction (i.e., via split CP or QCCP) using the PA score, where the score internally uses m paraphrases generated from x. Let the ideal coverage level be*

$$p_{\mathrm{ideal}} := \mathbb{P}\Big(Y \in \widehat{C}(X)\Big).$$

where the test sample is (X, Y). Under Assumption C.1, for this test sample $(\mathcal{T}(X_{n+1}), Y_{n+1})$, we have

$$\Big|\mathbb{P}\Big(Y \in \widehat{C}(\mathcal{T}(X))\Big) - p_{\mathrm{ideal}}\Big| \le m\varepsilon.$$

In particular, if $p_{\mathrm{ideal}} = 1 - \alpha$, then

$$\Big|\mathbb{P}\Big(Y \in \widehat{C}(\mathcal{T}(X))\Big) - (1 - \alpha)\Big| \le m\varepsilon.$$

Proposition C.2 shows that violations of distributional closedness translate into coverage deviations that scale linearly with the number of paraphrases. In particular, if the paraphrase pipeline deviates from ideal closedness by at most ε in total variation, then the resulting coverage gap is at most $m\varepsilon$ (up to finite-sample effects).

Proof of Proposition C.2. Let $A(x, y) = \mathbf{1}\{y \in \widehat{C}(x)\}$. By total variation, for any distributions P, Q, $|\mathbb{P}_P(A) - \mathbb{P}_Q(A)| \le d_{\mathrm{TV}}(P, Q)$. It suffices to bound the distance between $\mathrm{Law}(\widetilde{\mathcal{T}}(\mathcal{T}(X)), Y)$ and $\mathrm{Law}(\widetilde{\mathcal{T}}(X), Y)$. Condition on $X = x$. Then $\widetilde{\mathcal{T}}(X)$ consists of m i.i.d. samples from $\mathcal{T}(\cdot \mid x)$, while $\widetilde{\mathcal{T}}(\mathcal{T}(X))$ consists of m i.i.d. samples from $\mathcal{T}(\cdot \mid x')$, where $x' \sim \mathcal{T}(\cdot \mid x)$.

By the tensorization property of total variation,

$$d_{\mathrm{TV}}\big(\mathcal{T}(\cdot \mid x')^{\otimes m}, \mathcal{T}(\cdot \mid x)^{\otimes m}\big) \le m\, d_{\mathrm{TV}}(\mathcal{T}(\cdot \mid x'), \mathcal{T}(\cdot \mid x)).$$

Taking expectation over $x' \sim \mathcal{T}(\cdot \mid x)$ and using Assumption C.1, we obtain

$$d_{\mathrm{TV}}\Big(\mathrm{Law}(\widetilde{\mathcal{T}}(\mathcal{T}(X)) \mid X = x), \mathrm{Law}(\widetilde{\mathcal{T}}(X) \mid X = x)\Big) \le m\varepsilon.$$

Averaging over X yields

$$d_{\mathrm{TV}}\Big(\mathrm{Law}(\widetilde{\mathcal{T}}(\mathcal{T}(X)), Y), \mathrm{Law}(\widetilde{\mathcal{T}}(X), Y)\Big) \le m\varepsilon.$$

Therefore,

$$\Big|\mathbb{P}\{Y \in \widehat{C}(\widetilde{\mathcal{T}}(\mathcal{T}(X)))\} - \mathbb{P}\{Y \in \widehat{C}(\widetilde{\mathcal{T}}(X))\}\Big| \le m\varepsilon.$$

Combining this with the ideal coverage guarantee under exact closedness yields the result. □

D Experimental Details

D.1 Dataset Details

We used five general and two medical MCQA datasets to show that our paraphrase-aware score can be broadly used and can adapt to high-stakes scenarios where calibration is important. The five general datasets test a wide range of LLM capabilities and have been used in previous CP benchmarking papers [53, 42]. We use the five general datasets processed by Ye et al. [53]. Each of the datasets contains 10,000 questions.

1. **QA, MMLU** [15]: MMLU is a dataset designed to test the general knowledge and *question-answering* abilities of LLMs. Question topics range from sociology and high school geography to electrical engineering and abstract algebra.
2. **RC, CosmosQA** [16]: CosmosQA focuses on gauging an LLM's *reading comprehension* abilities. The LLM is given a short paragraph and is then asked to answer a follow-up question based on commonsense reasoning.
3. **CI, HellaSwag** [54]: HellaSwag evaluates if LLMs can use *commonsense inference* to construct a realistic and meaningful continuation of a given scenario. HellaSwag is deliberately designed so that LLMs struggle with questions that humans could normally answer with high confidence.
4. **DRS, HaluEval** [23]: HaluEval consists of hallucinated LLM responses to user queries. A subset of HaluEval contains queries that relate to *dialogue response selection*: the LLM must be able to choose a logical response for a conversation. We refer to this part of HaluEval as HaluDial.
5. **DS, HaluEval** [23]: The HaluEval dataset also has hallucinated *document summaries*. In an MCQA setting, the LLM must determine which summary in the answer choices is most relevant to a provided document. We refer to this part of HaluEval as HaluSum.
6. **MedMCQA** [28]: MedMCQA is a large-scale multiple-choice medical QA dataset comprising over 194,000 entrance-exam–style questions spanning 2400 healthcare topics and 21 medical subjects. We select a subset of 10,000 single-answer questions from MedMCQA (i.e. where exactly one option is marked correct) for our experiments.
7. **MedQA** [19]: MedQA is a medical exam QA dataset derived from professional medical board exams (e.g. USMLE), providing each question paired with candidate answer options and corresponding references. In our work, we use only the US-part of MedQA, and further restrict to 10,000 multiple-choice items that have exactly one correct answer.

Following the LLM-Uncertainty-Benchmark protocol [53], every question across all seven datasets is presented with a uniform label space of *six* options $|\mathcal{Y}| = 6$: the original answer choices (typically four for MMLU, CosmosQA, HellaSwag, and MedMCQA, and four to five for MedQA) are augmented with two additional options, "`I don't know`" and "`None of the above`", and HaluDial/HaluSum are likewise cast into this six-way format. This is why all reported set sizes are bounded by 6 rather than by the number of native answer choices.

D.2 Paraphrase Generation for Full/Semi-Paraphrase Settings

We only use this procedure to generate paraphrases for the input question used in the full- and semi-paraphrase evaluation settings. This set of reworded questions is not directly used for calculating the PA score, but is used as another set of input questions.

Prompt Pool. We maintain 7 short templates (e.g., "Rephrase this question using varied vocabulary and phrasing: {question}"). One template is sampled per input to induce lexical or syntactic variety without changing semantics.

Batched Generation. Given a batch of n questions, we form n prompts and generate once with: `temperature`=0.7, `top_p`=0.9, `do_sample`=True, `eos/pad_token_id` aligned. We decode with `skip_special_tokens=True`.

Cleaning & Validation. We remove the prompt prefix and extract the first question sentence via regex matching the earliest "?". We then strip boilerplate ("Rephrase:", "Paraphrase:", "Here's a ..."), quotes/bullets/code blocks, and normalize whitespace. The candidate paraphrase must satisfy: (i) non-empty, (ii) case-insensitive $\neq q$, (iii) ends with "?". Duplicates within the batch are dropped.

Retry & Fallback. Failures are retried with a stricter template: single line, no preface, $\leq$20 words, must end with "?". Remaining failures are rewritten by a deterministic rule set that preserves meaning (e.g., "Which of the following"$\rightarrow$"Which option", add trailing "?", etc.).

Algorithm 2 Paraphrase Pipeline (per question q)

1: Sample a paraphrase template; compose prompt $p(q)$.
2: Generate with (T=0.7, top-p=0.9, sampling).
3: Decode; strip prompt prefix; take first sentence ending with "?".
4: Clean and validate; if valid, return $\tilde{q}$.
5: Else: retry with strict one-line template; clean and validate.
6: Else: apply rule-based fallback; return $\tilde{q}$.

D.3 Baseline Score Adaptations

SPUQ-adapt. and TRON-adapt. are per-option score controls rather than the original SPUQ and TRON algorithms. For raw option probabilities $p_y(x)$, let $\bar{p}_y(x) = |\mathcal{B}(x)|^{-1} \sum_{\tilde{x}\in\mathcal{B}(x)} p_y(\tilde{x})$. The probability-based controls are $1 - \bar{p}_y(x)$ and $1 - p_y(x)$, respectively, and their negative-log counterparts are $-\log(\bar{p}_y(x) + 10^{-12})$ and $-\log(p_y(x) + 10^{-12})$. The probability-based TRON control is exactly LAC. When computed using the same proxy probabilities and paraphrase views as Full PA, the mean-probability control is algebraically identical to S_{mean}; this identity does not extend to the weighted, worst-case, or negative-log scores.

D.4 Experimental Implementation Details

Paraphrased Generation for Calculating PA Score. Paraphrases were generated by prompting `Qwen3-8B` [51] with the following query: "Rephrase the following question in your own words (preserving its meaning): Original question: {question} \n Rephrased question:". Additional details, such as any context or the answer choices, were not included with the original question. A total of 6 paraphrases were generated per question. Paraphrases that were equivalent to the original question or were previously generated were not included in the final set. The temperature of `Qwen3-8B` [51] was set to 1.1 to encourage diverse responses.

LLM Embeddings. We next obtained the LLM embeddings for each question and paraphrase. These LLM embeddings are used to train the proxy model and serve as the LLM's representation of the query. We treated the paraphrases as new samples in the dataset and assigned them the same answer choices and correct label as their parent question. For each sample, the input prompt included the question and the list of answer choices without the correct label. The LLM embedding was extracted from the final hidden layer; this layer simultaneously encodes the LLM's understanding of the question and its predicted answer. The LLM logits were also retrieved by finding the raw score corresponding to each answer represented as a token (e.g. 'A', 'B', etc.). We then applied softmax to the logits to obtain a probability distribution over the answer choices. For each dataset except the DS dataset, the LLM was provided with 2 example questions and their correct labels. Due to the long context needed for the questions in the DS dataset, only 1 example was provided to the LLM in this case. We followed this procedure to get the embeddings and logits of the instruction-tuned versions of `Qwen3-8B` [51], `Llama-3.1-8B` [9], and `Phi-3-small-8k` [1].

Proxy Model. The proxy model calibrates the LLM's probability distribution of the answer choices across the paraphrases. We used a 2-layer MLP with ReLU activation as our proxy model. The input to the proxy model is an LLM's embedding for a question or paraphrase, and the output is a vector with dimension $|\mathcal{Y}|$. The loss function used to train the proxy model is $L_{total} = L_{CE} + \lambda_{ece} \times L_{ECE}$, where L_{CE} is the standard cross-entropy loss for multiclass classification and L_{ECE} is the soft-binned expected calibration error loss [20]. λ_{ECE} is a hyperparameter that can be tuned to increase or decrease the relative importance of calibration in different contexts. Out of the 4,000 questions in the training set, 600 questions were used as the validation set. Since each question has 6 paraphrases, the effective training and validation set sizes are 23,800 and 4,200, respectively. We conducted grid search on the following hyperparameters: learning rate, weight decay, λ_{ECE}, hidden dimension, and

the batch size. The hyperparameters used for each dataset and each model are provided in Appendix D.5.

Score Calculation. For our baselines, we only use logits from the *original* question. The LAC score [33] is defined as $s_{\text{LAC}}(x, y) = 1 - f(x)_y$, where $f(x)_y$ is the softmax probability of label y for a question x. The APS score [32] is $s_{\text{APS}}(x, y) = \sum_{y' \in \mathcal{Y}:\ f(x)_{y'} \geq f(x)_y} f(x)_{y'}$, i.e., the cumulative probability of labels ranked at least as high as y. To find the score of a question x using the PA method, the LLM embeddings of its paraphrases (represented by the set $\mathcal{B}(x)$) are inputted into the trained proxy model. Then, the score for answer choice y of question x is given by $S_{\text{mean}}(x, y)$, $S_{\text{weighted}}(x, y)$, or $S_{\text{worst}}(x, y)$ (see Section 4). Note that only the scores of the 6 paraphrases are used in the PA formulas. The LLM's embedding of the original question does not contribute to that question's final PA score. This is in contrast to LAC and APS, which only rely on the logits of the original question and do not factor in the paraphrases.

Split Conformal Prediction. Split CP [44] uses a separate calibration dataset to calculate a global score threshold that determines the prediction sets for the test set. For each calibration example, we use the nonconformity score from the LLM's softmax probabilities or proxy model's probability distribution. These nonconformity scores are collected and the $(1 - \alpha)$ quantile is estimated with a finite-sample correction $q_\alpha = \text{Quantile}(\{s_i\}, \lceil (n + 1)(1 - \alpha) \rceil / n)$. At test time, each example's prediction set is formed by including all labels whose score is below q_α, with a fallback to the most probable label if the set is empty. This procedure ensures that the empirical coverage approaches the nominal $1 - \alpha$ guarantee.

Quasi-Conditional Conformal Prediction. In contrast to split CP, QCCP [13] calculates class-specific thresholds for pre-defined classes. The feature function ϕ is defined to assign a class to a question based on its features. In the case that ϕ is a constant function, then QCCP is equivalent to split CP; we use this ϕ function (ϕ = intercept) to obtain all split CP results. In our QCCP analysis, we focus on a ϕ function that takes in the embedding of each question generated by `all-MiniLM-L6-V2` [31]. K-means clustering is used to separate the questions into clusters, with each cluster acting as a class, and a different score threshold is selected for each cluster. For a given test question, the ϕ function is either a one-hot encoding of its assigned cluster or a vector of its embedding's distances to the three closest cluster centroids (dist3) (which can be used to calculate the threshold). The coverage rates reported using QCCP are calculated marginally with the exception of Figure 4 and Figure 9, which displays class-wise coverages. We used the `conditionalconformal` package [13] to implement QCCP.

D.5 Hyperparameters for Reproducibility

Tables 8 and 9 list the framework hyperparameters for the five general MCQA datasets and the two medical MCQA datasets, respectively. Tables 10 and 11 report the corresponding hyperparameters for the proxy model on the general and medical datasets.

E Additional Experimental Results

E.1 Better Performance on Medical MCQA Datasets

Due to space constraints, we report results on medical MCQA datasets here. We evaluate whether paraphrase-aware conformal prediction generalizes to domain-specific settings using two medical benchmarks, MedMCQA [28] and MedQA [19], with `Qwen3-8B` under normal, semi-reworded, and fully reworded conditions. As shown in Table 12, the results are consistent with those in the general domain (shown in Tables 3 and 4). APS tends to overshoot the 90% target, with coverage ranging from 92.93% to 100.00%. LAC is also often conservative and typically produces larger prediction sets. While SPUQ and TRON can achieve coverage closer to the nominal level in some cases, they generally require larger sets than PA. In contrast, our PA score maintains coverage close to the target, ranging from 89.57% to 91.77%, while yielding compact prediction sets of size 2.02 to 3.61. Notably, PA achieves the smallest set size across all 12 medical-domain settings. These results demonstrate that PA preserves a strong coverage–efficiency trade-off in the medical domain, highlighting its ability to generalize beyond general-purpose benchmarks.

Table 8: Hyperparameters for QA, RC, CI, DRS, DS. We train an MLP proxy with ECE regularization and compute paraphrase-aware scores using 6 paraphrases per question. Evaluation includes QCP with SBERT-clustered Φ and plain CP with intercept Φ, using the same manifest-based splits and α=0.1.

Setting / Hyperparameter	**QA, RC, CI, DRS, DS**
General	
Base LLM (for reps/logits)	Qwen3-8B
Random seed	42
Samples per question (n)	7 (1 base + 6 paraphrases)
Options per item ($\lvert\mathcal{Y}\rvert$)	6 (A–F)
Proxy model (training)	
Input dim	4096
Architecture	MLP: 4096 $\to h$ (ReLU) $\to$ 6
Optimizer	Adam
Max epochs / patience	200 / 20
Loss	CE + soft-binned ECE (15 bins, temp 0.1))
Paraphrase-aware score computation	
Paraphrases per question	6
Metric used	`S_mean`
Conformal prediction / QCP evaluation	
Prompting / ICL	`base` / `icl1`
Error level (α)	0.1
Split config	manifest: `tr0.4 / cf0.3 / tf0.3`
QCP Φ mode	`cluster_sbert`
SBERT model	MiniLM-L6-v2
Cluster selection	auto
Φ representation	dist3
Embedding norm / mini-batch k-means	on / on
Also reported (plain CP)	`intercept` Φ (same splits, α)

E.2 Finite-Sample Behavior of Class-Conditional Coverage

When applying our PA scores within QCCP, we obtain population-level guarantees of the form

$$1 - \alpha \leq \Pr(Y \in \widehat{C}(X) \mid X \in G) \leq 1 - \alpha + \text{slack},$$

for each cluster G, where the slack term decreases with the effective sample size within the cluster (see Lemma B.2).

It is important to note that this guarantee holds at the population level, rather than for finite-sample empirical estimates. In practice, when we estimate coverage within each cluster using a finite number of samples, the observed coverage rate will generally fluctuate around the target level $1 - \alpha = 0.90$. These deviations arise from two sources: (i) sampling variability due to finite data, and (ii) the presence of a nonzero slack term, which depends on the size and heterogeneity of each cluster. As a result, empirical cluster-wise coverage is not expected to match 0.90 exactly, even when the theoretical guarantee holds. Therefore, moderate deviations from 0.90 in our experiments should not be interpreted as a violation of coverage guarantees, but rather as a natural consequence of finite-sample estimation under the QCCP framework.

To illustrate this phenomenon, we conduct three additional analyses in the semi-reworded setting, varying the number of clusters $K \in \{5, 7, 10, 20\}$.

Wilson Interval Analysis. To assess whether deviations from the nominal coverage are due to finite-sample variability, we construct Wilson 95% confidence intervals for the empirical coverage of each cluster. Pooling clusters across datasets and choices of $K \in \{5, 7, 10, 20\}$, we observe a clear pattern. Among clusters with empirical coverage in $[0.85, 0.90)$—which appear mildly under-covered in Figure 4—a large majority (85.9%, $67/78$) have confidence intervals that still contain

Table 9: Hyperparameters for MedMCQA-10k and MedQA-10k. We train an MLP proxy with ECE regularization and compute paraphrase-aware scores using 6 paraphrases per question. Evaluation includes QCP with SBERT-clustered Φ and plain CP with intercept Φ, using the same manifest-based splits and α=0.1.

Setting / Hyperparameter	**MedMCQA-10k**	**MedQA-10k**
General		
Base LLM (for reps/logits)	Qwen3-8B	
Random seed	42	
Samples per question (n)	7 (1 base + 6 paraphrases)	
Options per item (K)	6 (A–F)	
Proxy model (training)		
Input dim	4096	
Architecture	MLP: 4096 $\rightarrow$ h (ReLU) $\rightarrow$ 6	
Optimizer	Adam	
Max epochs / patience	50 / 5	
Loss	CE + soft-binned ECE (15 bins, temp 0.1)	
Paraphrase-aware score computation		
Paraphrases per question	6	
Metric used	`S_mean`	
Conformal prediction / QCP evaluation		
Prompting / ICL	`task` / `icl1`	
Error level (α)	0.1	
Split config	manifest: `tr0.4 / cf0.3 / tf0.3`	
QCP Φ mode	`cluster_sbert`	
SBERT model	MiniLM-L6-v2	
Cluster selection / K	fixed / 20	
Φ representation	one-hot (cluster ID)	
Embedding norm / mini-batch k-means	off / off	
Also reported (plain CP)	`intercept` Φ (same splits, α)	

Table 10: Proxy model specific hyperparameters for the 5 general datasets

Dataset	**Hidden dimension h**	**Batch size**	**Learning rate**	**Weight decay**	λ_{ECE}
MMLU	256	64	1e-3	0.0	0.5
CosmosQA	512	128	1e-3	0.0001	0.5
HellaSwag	256	128	1e-3	0.0001	0.5
HaluDial	256	64	1e-3	0.0	0.5
HaluSum	256	128	1e-4	0.0	0.5

Table 11: Proxy model specific hyperparameters for MedMCQA and MedQA.

Dataset	**Hidden dimension h**	**Batch size**	**Learning rate**	**Weight decay**	λ_{ECE}
MedMCQA	256	64	1e-3	0.0	0.5
MedQA	256	64	1e-3	0.0	0.5

the nominal 0.90 target. This suggests that these moderate deviations are largely attributable to finite-sample fluctuations. In contrast, for clusters with empirical coverage below 0.85, none ($0/13$) of the corresponding confidence intervals contain 0.90, indicating that these larger deviations are unlikely to be explained by sampling variability alone.

Table 12: Results on two medical QA benchmarks with `Qwen3-8B`. Coverage rate (CR, %) closer to the nominal 90% is better, while smaller set size (SS, $\downarrow$) denotes better. Results are shown for normal, semi-reworded (Semi.), and fully reworded (Full) inputs.

		MedMCQA						**MedQA**					
		CR (%)			SS			CR (%)			SS		
	Method	Normal	Semi.	Full	Normal	Semi.	Full	Normal	Semi.	Full	Normal	Semi.	Full
Split CP	LAC	99.03	98.90	90.20	5.64	5.63	2.74	98.97	98.73	99.40	5.81	5.82	5.90
	APS	98.97	98.93	98.90	5.64	5.65	5.65	99.97	100.00	100.00	6.00	6.00	6.00
	SPUQ-adapt.	90.10	86.37	91.27	2.59	2.34	2.86	**89.97**	81.83	88.80	4.05	3.72	4.43
	TRON-adapt.	90.17	87.73	91.03	2.46	2.50	2.81	90.47	84.97	89.00	3.97	3.99	4.35
	PA	**90.00**	**89.63**	**89.87**	**2.23**	**2.02**	**2.17**	91.73	**90.80**	**90.50**	**3.34**	**3.52**	**3.52**
QCCP	LAC	92.87	93.53	94.83	3.64	3.98	4.13	95.60	94.47	93.67	4.97	5.20	5.06
	APS	96.93	97.27	96.87	4.43	4.85	4.45	94.27	92.93	93.73	5.02	5.10	5.16
	SPUQ-adapt.	90.27	87.43	91.70	2.74	2.61	3.02	**90.53**	82.37	89.40	4.14	3.80	4.50
	TRON-adapt.	**90.10**	89.10	91.40	2.61	2.77	2.99	90.60	85.37	88.87	4.05	4.02	4.37
	PA	89.57	**90.03**	**89.67**	**2.47**	**2.48**	**2.40**	91.77	**90.23**	**90.30**	**3.43**	**3.55**	**3.61**

Table 13: Cluster-size-weighted coverage deviation (WCD, $\downarrow$) under QCCP in the semi-reworded setting on `Qwen3-8B`. Lower values indicate more uniform alignment with the nominal $1-\alpha=0.9$ target. Bold marks the best method per dataset.

Dataset	**LAC**	**APS**	**PA**
QA (MMLU)	0.0555	0.0687	**0.0461**
CI (HellaSwag)	0.0402	0.0860	**0.0293**
DRS (HaluDial)	0.0787	0.0476	**0.0262**

Effect of Cluster Sample Size. Cluster size has a clear impact on empirical coverage. Larger clusters tend to track the nominal target more closely, as expected from the theoretical guarantees. We partition PA-QCCP clusters into terciles based on calibration sample size (thresholds $q_1 = 166.67$ and $q_2 = 318.67$, with 168 clusters in total). The average coverage increases monotonically across these groups: 0.8963 (smallest), 0.9025 (middle), and 0.9059 (largest). A similar trend appears in the tail behavior. The fraction of clusters with empirical coverage ≤ 0.85 drops sharply from approximately 10.7% in the smaller two terciles to 1.8% in the largest tercile. These results confirm that effective local sample size plays a key role in stabilizing coverage. At the same time, it is not the sole factor: some large clusters, particularly in semantically challenging regions, still exhibit noticeable deviations.

Cluster-Size-Weighted Coverage Deviation. To summarize cluster-level alignment with a single scalar, we define the *cluster-size-weighted coverage deviation* (WCD):

$$\mathrm{WCD} \;=\; \sum_{k=1}^{K} \frac{n_k}{N} \left|\mathrm{CR}_k - (1-\alpha)\right|, \tag{6}$$

where n_k is the size of cluster k, $N = \sum_k n_k$, and CR_k is the empirical coverage in cluster k. Lower values indicate better alignment with the nominal target. Since all methods share the same cluster assignment, WCD provides a fair and comparable summary across baselines.

As shown in Table 13, PA achieves the lowest WCD on QA, CI, and DRS at $K = 20$ on `Qwen3-8B`, indicating more uniform alignment with the 0.9 target across semantic clusters than LAC or APS.

E.3 Robustness under the Semi-Reworded Setting

Recall that the two versions of each dataset in our evaluation are defined as $\mathcal{D}_{\text{orig}} = \{(X_i, Y_i)\}_{i=1}^{N}$ and $\mathcal{D}_{\text{reworded}} = \{(\mathcal{T}(X_i), Y_i)\}_{i=1}^{N}$, where $\mathcal{T}$ represents the rewording operation. In the semi-reworded setting, we test a realistic semantic shift scenario where the calibration set is drawn from $\mathcal{D}_{\text{orig}}$ (human-written inputs), but the test set is drawn from $\mathcal{D}_{\text{reworded}}$ (LLM-reworded inputs). Under this shift, the exchangeability assumption required for standard conformal prediction is violated for baselines like LAC and APS, as the distribution of their nonconformity scores (which depend on raw logits) changes between calibration and testing.

However, we hypothesize that our PA score remains robust to this shift. Our PA score is calculated by aggregating predictions over the set of paraphrases. We denote this paraphrasing generation as $\mathcal{T}^*$ to distinguish between the rewording used to create a distinct dataset ($\mathcal{T}$) and the paraphrasing used to calculate the PA score ($\mathcal{T}^*$). Consequently, for a calibration point X, the score $S(X, Y)$ depends on $\mathcal{T}^*(X)$. For a test point $\mathcal{T}(X)$, the score depends on the composition $\mathcal{T}^*(\mathcal{T}(X))$. Invoking Assumption 5.1 (distributional closedness), we posit that $\mathcal{T}^*(\mathcal{T}(X)) \overset{d}{\approx} \mathcal{T}^*(X)$. Under the exact alignment and score conditions stated in our analysis, calibration and test scores retain the required exchangeability. Under approximate alignment, only the corresponding coverage-deviation bound applies. We do not establish these conditions for an arbitrary external rewriter and internal generator. Please refer to Table 15 which reports the observed clean-to-rewrite change at a fixed cutoff.

Cluster Φ training in the semi-reworded setting. For QCCP, the conditioning feature $\Phi(X)$ must remain exchangeable between calibration and test paired with the score $s(X, Y)$. In the semi-reworded setting, calibration uses X but test inputs are $\mathcal{T}(X)$. To preserve (Φ, s) exchangeability for QCCP under this shift, we therefore train and apply the SBERT cluster Φ on a fixed paraphrase $\mathcal{T}^\dagger(X)$ (a single LM rewording, identical for calibration and test pieces), rather than on the raw original or raw rewritten text. The PA / LAC / APS / TRON / SPUQ scores themselves are computed exactly as in the main text; only the input to the cluster encoder is changed.

We further evaluate our hypothesis in the semi-reworded setting using `Llama-3.1-8B` as both the dataset-level rewriter ($\mathcal{T}$) and the internal paraphrase generator ($\mathcal{T}^*$). As shown in Table 14, LAC performs poorly under this shift, achieving only 87.17% coverage on CosmosQA and 86.97% on HellaSwag, both well below the nominal 90% target. APS overshoots on coverage and does so with substantially inflated prediction sets (SS up to 5.85 on CosmosQA and 4.79 on HellaSwag), making its predictions considerably less actionable.

In contrast, PA delivers the best tradeoff between reliability and efficiency. Under split CP, PA achieves 89.97% coverage on CosmosQA and 90.27% on HellaSwag—both tightly aligned with the nominal 90%—while producing the smallest prediction sets (SS $\approx$ 1.41–2.03). Under QCCP, PA again reaches coverage closest to 90% and maintains compact sets, substantially outperforming both LAC and APS. Compared with Table 4, these results highlight that the quality of the paraphrase generator $\mathcal{T}^*$ is crucial: with a stronger model such as `Llama-3.1-8B`, empirical coverage improves for the evaluated generator.

Table 14: Performance of nonconformity scores under the **semi-reworded (Semi.)** setting on `Llama-3.1-8B` as paraphrase generator. Coverage Rate (CR, %) closer to 90% is better; Set Size (SS) ↓ is better. Results are reported on RC (Cosmos) and CI (HellaSwag). Bold numbers indicate the best performance.

	Dataset	**RC**	**CI**	**RC**	**CI**
	Semi.	Coverage Rate (CR)		Set Size (SS)	
Split CP	LAC	87.17	86.97	3.51	3.91
	APS	99.33	90.90	5.85	4.79
	PA	**89.97**	**90.27**	**1.41**	**2.03**
QCCP	LAC	87.00	87.77	3.51	3.98
	APS	97.40	95.10	5.41	5.33
	PA	**90.00**	**90.43**	**1.43**	**2.01**

E.4 Quantifying Coverage Deviations in the Semi-Reworded Setting

Our implementation uses an external rewriter $\mathcal{T}$ and a separate internal paraphrase generator $\mathcal{T}^*$. Exact distributional closedness is a sufficient idealization; neither the exact theorem nor its approximate bound establishes that this implemented pipeline satisfies the assumption. We therefore examine the realized score mismatch at the conformal decision boundary.

For each dataset and seed, we freeze the clean-trained proxy and the clean split-CP cutoff $\hat{q}$, then score the same test questions in original and externally rewritten form. Let F_{clean} and F_{rewrite} be the

Table 15: Paired fixed-clean-cutoff diagnostics on `Qwen3-8B` with S_{mean} and split CP at $\alpha = 0.1$. Each row is the mean over seeds 7, 42, and 123. The proxy and clean calibration cutoff are held fixed while the same test questions are evaluated before and after rewriting. Coverage error is relative to the nominal 90% target; $\Delta_{\hat{q}}$ is relative to clean empirical coverage.

Dataset	Semi CR (%)	Error from 90% (pp)	$\Delta_{\hat{q}}$ (pp)	KS
MMLU	85.5	-4.5	-4.5	0.059
CosmosQA	89.1	-0.9	-1.1	0.052
HellaSwag	89.5	-0.5	-0.5	0.084
HaluDialogue	87.6	-2.4	-2.7	0.135
HaluSummarization	93.1	$+3.1$	$+2.9$	0.085

empirical CDFs of their true-label nonconformity scores. We report the signed cutoff-local change

$$\Delta_{\hat{q}} = F_{\text{rewrite}}(\hat{q}) - F_{\text{clean}}(\hat{q}). \tag{7}$$

A negative value means that rewriting moves true-label score mass above the cutoff, creating undercoverage pressure. The global KS distance is $\sup_t |F_{\text{rewrite}}(t) - F_{\text{clean}}(t)|$. KS is the maximum absolute gap between the two CDFs over all thresholds and therefore measures global shift.

The two middle columns need not be numerically identical: the coverage error is measured relative to the nominal 90% target, whereas $\Delta_{\hat{q}}$ is measured relative to clean empirical coverage. For example, MMLU loses approximately 4.5 percentage points of true-label mass below the operative cutoff and undercovers by 4.5 points. In contrast, HaluDialogue has the largest global KS distance (0.135), but only a 2.7-point local shift and a 2.4-point coverage error, showing why global score shift alone does not determine coverage.

Across the 15 underlying dataset–seed cells, rather than the five displayed means, cutoff-local shift and Semi coverage deviation have Pearson $r = 0.983$ (paired-question bootstrap 95% CI $[0.935, 0.985]$). At a fixed cutoff, this local shift directly accounts for the observed clean-to-rewrite empirical coverage change.

E.5 Semantic Consistency of Generated Paraphrases

To verify that the generated paraphrases approximately preserve the semantics of the original questions, we conduct a quantitative paraphrase quality evaluation across CosmosQA (RC), HellaSwag (CI), and MedMCQA (medical MCQA). For each dataset, we use the same paraphrase generation pipelines as in the main experiments and evaluate all paraphrases that are actually used for conformal prediction.

Paraphrase sets and generators. Recall that for each original question x we use three paraphrase-related sets: (i) $\mathcal{T}^*(x)$, the set of paraphrases used for PA score computation in the normal setting; (ii) $\mathcal{T}(x)$, the fully reworded questions used in the fully reworded setting; and (iii) $\mathcal{T}^*(\mathcal{T}(x))$, the paraphrases of fully reworded questions used for PA score computation in the fully reworded setting. We instantiate these sets using two independent LLM paraphrase generators: `Llama-3.1-8B` and `Qwen3-8B`. All quality metrics are reported separately for each generator and each scenario.

Metrics. Given an original question x and a paraphrase p, we compute three standard similarity metrics:

1. **SBERT cosine similarity.** We use the `all-MiniLM-L6-v2` SentenceTransformer to embed both x and p, normalize the embeddings, and compute

   $$\cos(x, p) \;=\; \langle \text{SBERT}(x), \text{SBERT}(p) \rangle.$$

   This serves as a semantic similarity measure that is robust to minor lexical changes.
2. **BLEU.** We compute sentence-level BLEU scores using `sacrebleu` with effective order enabled, treating the original question as the reference and the paraphrase as the hypothesis.
3. **ROUGE-L.** We compute ROUGE-L F-measure using the `rouge_score` library with stemming enabled, again using the original question as the reference and the paraphrase as the candidate.

Aggregation and reporting. For each question x and each paraphrase set (e.g., $\mathcal{T}^*(x)$), we first compute the metric values for all paraphrases $p \in \mathcal{T}^*(x)$ and then average them to obtain a single per-question score. We repeat this for SBERT cosine, BLEU, and ROUGE-L. At the dataset level, we report the mean and standard deviation across all questions that have at least one paraphrase in the corresponding set. This yields, for each dataset, scenario, and model, a tuple of the form

$$\text{metric} \;=\; \mu \pm \sigma,$$

which we summarize in Table 16.

Findings. Across CosmosQA and MedMCQA, we observe moderate to high SBERT cosine similarity (typically in the range 0.61–0.87, with Qwen3-8B paraphrases consistently above 0.81), indicating that the paraphrases remain close in semantic space to the original questions while allowing stylistic variation. HellaSwag exhibits lower cosine scores (0.37–0.59), which is expected due to its longer, multi-sentence contexts and richer surface-level variability. BLEU and ROUGE-L follow the same pattern across datasets and paraphrase scenarios. Overall, these results suggest that the paraphrase sets $\mathcal{T}^*(x)$, $\mathcal{T}(x)$, and $\mathcal{T}^*(\mathcal{T}(x))$ largely preserve the semantics of the original questions and are suitable for use in our paraphrase-aware conformal prediction framework.

E.6 End-to-End Runtime

We measure the complete online path on one NVIDIA B200 with `Qwen3-8B`. Online Full PA includes paraphrase generation, backbone inference, proxy scoring, aggregation, and prediction-set construction. The cached column assumes paraphrases and hidden representations have already been stored. All entries are mean milliseconds per question.

In these measurements, online Full PA with $m = 6$ is 22.7–83.4 times slower than raw single-input inference; paraphrase generation accounts for 67–97% of the online time. Once the views are cached, the remaining scoring, aggregation, and set construction take at most 0.20 ms per question. The online generation cost is therefore a practical limitation when latency matters.

F Analysis of the Proxy Model

A natural question is whether the gain of our method comes from the proxy classifier alone—which improves the calibration of each individual prediction—or from the paraphrase-aware (PA) score that aggregates variability across rewordings. To disentangle the two, we compare four variants under the same `Qwen3-8B` backbone and the same CP/QCCP wrapper:

1. **Full PA** (proxy + paraphrase aggregation): the method proposed in the main paper.
2. **Paraphrase-only**: PA score computed directly from raw LLM logits, with no proxy classifier.
3. **Aug-Proxy Single**: the paraphrase-augmented proxy evaluated on a single input view, without inference-time aggregation.
4. **Neither** ($\equiv$ LAC): raw logits with no proxy and no paraphrase aggregation.

Removing the Proxy Alone. We first report proxy-removal ablation results under split CP in Figure 7; the same trend holds under QCCP (Figure 8). Coverage remains close to 90%, but set sizes inflate substantially without the proxy.

Single-Run Attribution. Table 18 reports CR/SS for the three PA variants in both the semi-reworded and fully reworded settings, under both CP and QCCP. The pattern is consistent across datasets:

- In the **semi-reworded** setting, *paraphrase-only* becomes unstable without proxy calibration: under split CP it drops to 84.83% on QA and 87.93% on CosmosQA, while under QCCP it swings to $94.20\%/4.53$ on HaluDial. *Aug-Proxy Single* is more stable but tends to be conservative, especially on DS where QCCP reaches 95.30% coverage. The *full method* is usually the most efficient variant, keeping the smallest or tied-smallest sets while staying close to nominal on RC, CI, and DRS.

Table 16: **Paraphrase quality evaluation** using Llama-3.1-8B and Qwen3-8B as paraphrase generators. We report SBERT cosine similarity, BLEU, and ROUGE-L (mean ± std) for three paraphrase scenarios: $\mathcal{T}^*(x)$, $\mathcal{T}(x)$, and $\mathcal{T}^*(\mathcal{T}(x))$.

RC (CosmosQA)			
Scenario / Model	**Cosine**	**BLEU**	**ROUGE-L**
$\mathcal{T}^*(x)$ — *paraphrases for human-written inputs*			
Llama	0.82 ± 0.10	16.68 ± 8.57	45.18 ± 12.02
Qwen3	0.81 ± 0.10	16.06 ± 8.60	42.19 ± 11.64
$\mathcal{T}(x)$ — *LLM-reworded inputs*			
Llama	0.70 ± 0.21	11.94 ± 13.24	35.84 ± 19.98
Qwen3	0.84 ± 0.11	19.28 ± 14.52	49.17 ± 17.48
$\mathcal{T}^*(\mathcal{T}(x))$ — *paraphrases of LLM-reworded inputs*			
Llama	0.70 ± 0.20	11.73 ± 9.83	35.62 ± 16.60
Qwen3	0.82 ± 0.11	19.07 ± 11.22	48.20 ± 13.76

CI (HellaSwag)			
Scenario / Model	**Cosine**	**BLEU**	**ROUGE-L**
$\mathcal{T}^*(x)$ — *paraphrases for human-written inputs*			
Llama	0.49 ± 0.03	9.80 ± 2.08	34.39 ± 4.53
Qwen3	0.54 ± 0.03	8.96 ± 2.85	31.08 ± 4.97
$\mathcal{T}(x)$ — *LLM-reworded inputs*			
Llama	0.37 ± 0.15	6.71 ± 4.64	24.65 ± 11.55
Qwen3	0.57 ± 0.02	9.11 ± 3.53	39.16 ± 6.78
$\mathcal{T}^*(\mathcal{T}(x))$ — *paraphrases of LLM-reworded inputs*			
Llama	0.37 ± 0.13	6.66 ± 2.75	24.61 ± 8.13
Qwen3	0.59 ± 0.06	20.91 ± 7.20	48.79 ± 7.13

MedMCQA			
Scenario / Model	**Cosine**	**BLEU**	**ROUGE-L**
$\mathcal{T}^*(x)$ — *paraphrases for human-written inputs*			
Llama	0.81 ± 0.10	10.41 ± 6.50	38.25 ± 11.39
Qwen3	0.84 ± 0.08	12.32 ± 7.74	40.16 ± 11.59
$\mathcal{T}(x)$ — *LLM-reworded inputs*			
Llama	0.61 ± 0.27	5.81 ± 5.69	24.51 ± 14.60
Qwen3	0.87 ± 0.09	14.81 ± 11.51	46.09 ± 15.90
$\mathcal{T}^*(\mathcal{T}(x))$ — *paraphrases of LLM-reworded inputs*			
Llama	0.61 ± 0.27	5.81 ± 5.69	24.51 ± 14.60
Qwen3	0.84 ± 0.10	12.33 ± 8.88	40.71 ± 13.43

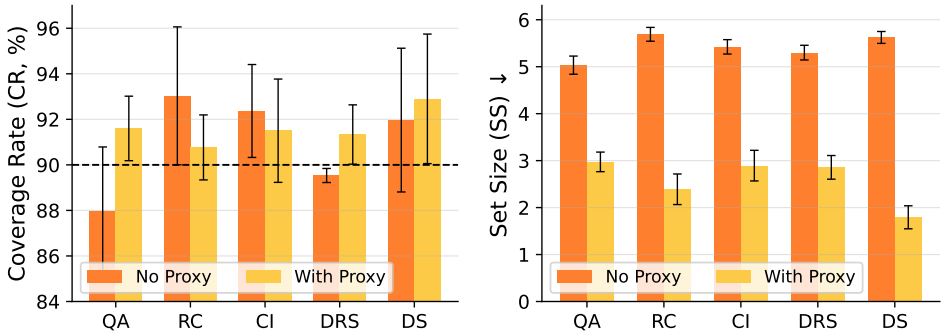


Figure 7: Effect of removing the proxy model under the split CP setting on `Qwen3-8B`. Coverage remains close to 90% (dashed line), but set size increases notably without the proxy model.

Table 17: End-to-end latency (ms/question) on one NVIDIA B200. Batch denotes the inference batch size; m is the number of paraphrases. Cached Full PA excludes paraphrase generation and backbone representation extraction.

Dataset	Batch	Raw LAC	Aug-Proxy Single	Full PA m=2	Full PA m=6	Cached m=6
MMLU	1	19.51	18.73	1599.76	1626.58	0.200
CosmosQA	1	24.14	23.52	1565.11	1611.55	0.195
MMLU	16	9.63	9.57	187.54	357.37	0.056
CosmosQA	16	10.78	10.74	150.32	245.09	0.056

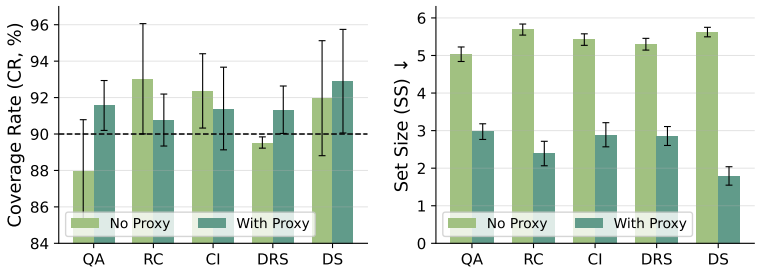


Figure 8: Effect of removing the proxy model under the quasi-conditional CP setting. Coverage remains close to 90% (dashed line), but set size increases notably without the proxy model.

- In the **fully reworded** setting, all three variants are close to nominal coverage, but the full method remains the most efficient in most columns: on CosmosQA under split CP its set size is 1.17 versus 1.77 for *paraphrase-only*, and on HaluDial under QCCP it is 1.39 versus 4.86. Even when *augmented-proxy single-view* is competitive, the full method still matches or slightly improves its set size on nearly every dataset.

These results motivate the controlled decomposition in Table 6: the proxy supplies most of the compactness, and same-checkpoint aggregation supplies a smaller stability gain.

Table 18: Single-run attribution of the PA score on `Qwen3-8B`. Each cell reports **CR / SS** (coverage rate, %, closer to 90 is better; set size ↓ is better). **Para+Proxy** denotes the full method, **Para only** removes the proxy, and **Aug-Proxy Single** removes inference-time aggregation while retaining paraphrase-augmented training. Bold entries highlight severe under- or over-coverage exhibited by single-component variants. These single-run rows retain the paraphrase-trained proxy in the single-view arm and are kept separate from the three-seed controls in Table 6 and Table 22; they are not pooled with those later three-seed controls.

Variant	QA (MMLU)	RC (Cosmos)	CI (HellaSwag)	DRS (HaluDial)	DS (HaluSum)
Semi-reworded, Split CP					
Para + Proxy (Full)	87.10 / **1.85**	**90.10 / 1.12**	89.13 / **1.28**	**89.67 / 1.36**	94.13 / **1.00**
Para only	84.83 / 2.21	87.93 / 1.53	87.30 / 1.60	86.23 / 2.72	**88.20** / 2.49
Aug-Proxy Single	**89.53** / 2.02	90.47 / 1.14	**89.37** / 1.29	90.87 / 1.43	95.20 / **1.00**
Semi-reworded, QCCP					
Para + Proxy (Full)	89.50 / **2.01**	**90.67 / 1.39**	89.53 / **1.36**	**91.10 / 1.70**	94.20 / **1.11**
Para only	87.73 / 2.54	88.70 / 1.94	88.63 / 2.06	94.20 / 4.53	**89.90** / 3.21
Aug-Proxy Single	**89.77** / 2.16	90.73 / 1.40	**89.90** / 1.51	91.87 / 1.76	95.30 / **1.11**
Fully reworded, Split CP					
Para + Proxy (Full)	90.37 / **2.13**	90.97 / **1.17**	**90.03** / 1.34	**90.57 / 1.35**	90.50 / **1.07**
Para only	**89.77** / 2.65	**90.20** / 1.77	89.77 / 1.79	91.13 / 3.32	**89.73** / 2.73
Aug-Proxy Single	90.57 / 2.15	91.03 / 1.19	89.73 / **1.32**	91.37 / 1.41	90.47 / **1.07**
Fully reworded, QCCP					
Para + Proxy (Full)	**90.43 / 2.40**	**90.97 / 1.37**	91.80 / **1.73**	**90.33 / 1.39**	**90.73 / 1.31**
Para only	91.47 / 3.12	**90.97** / 2.08	**91.50** / 2.69	97.00 / 4.86	94.97 / 4.46
Aug-Proxy Single	90.50 / 2.42	91.80 / 1.41	91.80 / 1.74	90.77 / 1.44	91.27 / **1.31**

Training-View Control. The main factorized comparison is in Table 6. Here we distinguish standard Orig-only training, which uses each original question once, from Repeat-Orig training, which repeats the same originals to match the augmented run's number of training rows and optimizer

Table 19: Training-view control for the hidden-state proxy on `Qwen3-8B`, averaged equally over five datasets and three seeds. All rows use single-view inference. Score drift is the mean absolute clean/rewrite change in the true-label nonconformity score. Repeat-Orig controls for repeated training updates; standard Orig-only is the primary no-paraphrase training baseline.

Proxy training	Rewrite Acc. ↑	Semi CR	Semi SS ↓	Score drift ↓	Set change ↓
Standard Orig-only	79.80	89.58	1.3725	0.0596	0.2009
Repeat-Orig	79.64	89.27	1.3717	0.0635	0.2188
Paraphrase-augmented	80.38	89.96	1.3587	0.0574	0.2029

Table 20: Held-out, approximately coverage-matched comparison in the Semi setting. The coverage columns report achieved coverage on the held-out evaluation half. Negative Full PA minus Single set-size differences favor Full PA. This is a separate operating-point analysis, not the fixed-α experiment.

Dataset	Single CR (%)	Full PA CR (%)	Full PA − Single SS
MMLU	89.81	89.43	−0.0022
CosmosQA	91.27	91.03	−0.0233
HellaSwag	89.90	89.43	−0.0404
HaluDialogue	90.12	90.18	−0.0178
HaluSummarization	89.27	89.36	+0.0002

updates. Augmented training instead uses the originals and up to six meaning-preserving paraphrases with their original labels.

Relative to standard Orig-only, paraphrase-augmented training improves rewritten-input accuracy by 0.58 percentage points and reduces true-label score drift by 3.7%. Repeat-Orig performs slightly worse than standard Orig-only on both measures, so the improvement cannot be attributed simply to seeing more copies of the original questions. The set-change measure for augmented single-view inference is slightly higher than for Orig-only (0.2029 versus 0.2009); the reduction to 0.1708 occurs when the same augmented checkpoint is used with six-view inference (Table 6).

Set size at closely matched empirical coverage. The fixed-α comparison in Table 6 uses the same nominal coverage target for every method, but achieved coverage can differ. As a separate descriptive analysis in the semi-reworded setting, we use one half of the shifted examples to choose an operating point near 90% coverage for each method and evaluate on the other half. Both methods use the same paraphrase-augmented proxy checkpoint. The results are shown in Table 20. At approximately matched achieved coverage, Full PA has smaller sets on four of the five datasets and is effectively tied on HaluSummarization.

Same-Checkpoint Inference Control. A separate five-dataset, three-seed study compares one versus six inference views using the same augmented proxy checkpoint, calibration/test IDs, and evaluator. Table 22 uses different frozen splits and validation-selected checkpoints from Table 6. In that study, aggregation reduces mean clean/rewrite set change by 12.9% with an original-input-fitted cutoff and 5.2% with a reworded-input-fitted cutoff (Appendix Table 21).

F.1 Different Class-Conditional Levels for QCCP

We report class-conditional coverage for QCCP on the remaining datasets in Figure 9, complementing Figure 4 in the main text. The results are consistent with those in the main text: PA tracks the 90% target across clusters, while LAC and APS overshoot and the TRON / SPUQ baselines exhibit larger per-cluster swings.

F.2 Different α-Level for Split CP

We report results at different user-specified risk levels under split CP in Figure 10, which complements Figure 5 for QCCP in the main text. The trends are consistent: PA tracks the nominal $(1-\alpha)$ target most tightly across all three levels (e.g., 79.7% at α=0.2, 95.1% at α=0.05, 99.2% at α=0.01) while keeping the most compact sets (1.63, 2.67, and 3.48 respectively); APS dramatically overshoots at α=0.2 ($\approx 95\%$ coverage with set size ≈ 4.5), and LAC produces strictly larger sets than PA at low α

Table 21: Paired prediction-set stability in the same separate study. Set change is the mean symmetric-difference size $|\widehat{C}(x) \triangle \widehat{C}(\mathcal{T}(x))|$, the number of labels entering or leaving a set under rewriting. Lower is better. Values are five-dataset macro means $\pm$ sample standard deviation across seeds 7, 42, and 123.

Proxy and cutoff fitted on	Aug-Proxy Single	Full PA	Relative change
Original questions	0.1994 ± 0.0082	0.1737 ± 0.0089	-12.9%
Reworded questions	0.2213 ± 0.0004	0.2098 ± 0.0043	-5.2%

Table 22: Separate same-checkpoint study of inference-time aggregation under split CP at $\alpha = 0.1$. Both variants use the same paraphrase-augmented proxy. CR is empirical coverage in percent and SS is mean prediction-set size. Values are five-dataset macro means $\pm$ sample standard deviation across seeds 7, 42, and 123.

Setting	Aug-Proxy Single: CR / SS	Full PA: CR / SS
Normal	90.58 ± 0.16 / 1.293 ± 0.007	90.50 ± 0.06 / 1.299 ± 0.005
Semi	90.40 ± 0.18 / 1.329 ± 0.008	89.56 ± 0.31 / 1.286 ± 0.006
Fully reworded	90.59 ± 0.37 / 1.352 ± 0.012	90.17 ± 0.31 / 1.338 ± 0.010

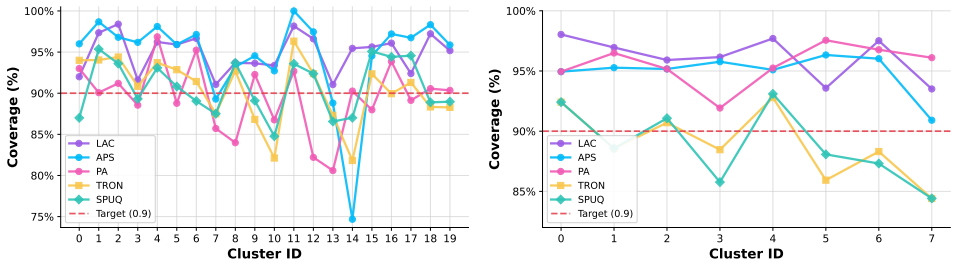


Figure 9: Evaluation of class-conditional coverage with QCCP on the MMLU (left) and HaluSum (right) datasets under `Qwen3-8B`.

(set size ≈ 4.8 at α=0.05 and ≈ 4.9 at α=0.01). The TRON and SPUQ baselines track the target reasonably well at small α but produce larger sets than PA at every level.

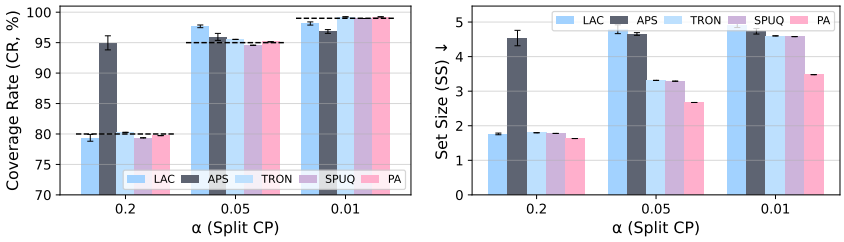


Figure 10: Coverage rate (left) and prediction set size (right) vs. α on MMLU under Split CP, with `Qwen3-8B` as the backbone.

F.3 Different Paraphrase Generators

To assess sensitivity to the choice of paraphrase generator, we use another model, `Llama-3.1-8B`, as the paraphrase generator while keeping the rest of the pipeline fixed. In particular, we continue to use `Qwen3-8B` for embedding extraction. We evaluate this setup on three datasets (CosmosQA, HellaSwag, and MedMCQA) under both normal and fully reworded settings. As shown in Table 23, PA remains robust to this change. Under split CP, it maintains coverage close to the 90% target while achieving smaller set sizes across all datasets. Similar trends are observed under QCCP.

Table 23: Performance of nonconformity scores under split CP and QCCP with `Llama-3.1-8B` as the paraphrase generator and `Qwen3-8B` as the embedding model. Results are reported on normal and fully reworded test sets across three benchmarks. Bold numbers indicate the best performance.

	RC (Cosmos)		**CI (HellaSwag)**		**MedMCQA**	
Method	Normal	Full	Normal	Full	Normal	Full
Coverage Rate (CR, %) Better if closer to 90% *— Split CP*						
LAC	90.40	99.00	89.83	**89.87**	**89.42**	99.20
APS	99.27	99.10	99.63	98.90	90.92	99.97
PA	**90.13**	**90.60**	**90.13**	90.67	91.03	**89.83**
Coverage Rate (CR, %) Better if closer to 90% *— QCCP*						
LAC	91.90	93.70	91.93	91.50	92.31	95.00
APS	98.60	97.70	99.00	98.70	96.69	96.80
PA	**90.63**	**90.57**	**90.63**	**90.80**	**91.88**	**90.13**
Set Size (SS) $\downarrow$ *— Split CP*						
LAC	1.55	5.56	1.67	1.75	2.45	5.67
APS	5.65	5.59	5.71	5.59	2.68	5.87
PA	**1.12**	**1.27**	**1.35**	**1.65**	**2.27**	**2.55**
Set Size (SS) $\downarrow$ *— QCCP*						
LAC	2.26	3.16	2.60	2.65	3.66	4.19
APS	4.86	4.61	5.06	5.00	4.54	4.46
PA	**1.34**	**1.47**	**1.70**	**1.97**	**2.51**	**2.71**

F.4 Different Embedding Choices

To evaluate whether the PA nonconformity score depends on a particular LLM representation, we conduct a controlled sensitivity analysis on the `Qwen3-8B` model using the **CosmosQA** benchmark. In the main paper, PA uses the last-layer hidden state at the final token position. Here, we consider three alternative representations that are commonly used in transformer analysis:

- **Mean pooling over tokens.** We average hidden states across all sequence positions (masking padding tokens). This yields a sequence-level representation that does not rely on a specific token location.
- **Intermediate-layer hidden states.** Instead of the last layer, we extract representations from a configurable layer index. In the experiment below, we uses the second-to-last layer). We keep the extraction location fixed (final token), so only the layer depth varies.
- **Attention-weighted pooling.** We use attention weights from the last layer to produce a weighted average over token embeddings. Specifically, we take the last token's attention distribution, average over attention heads, and use this vector as weights over the hidden states of all tokens.

Results. Tables 24 and 25 report coverage rate (CR) and set size (SS) under split CP. Across all pooling and layer choices, PA-based nonconformity scores remain extremely stable. This demonstrates that PA does *not* depend on a specific pooling strategy or layer depth, and that the semantic structure encoded throughout the transformer stack is sufficient for proxy training.

F.5 Different Proxy Training and Calibration Budgets

A practical concern is whether PA requires large labeled datasets to fit the proxy classifier. To investigate this, we study how performance varies with two label budgets: (i) the *proxy-train fraction*, denoted by tf, the fraction of training instances used to fit the proxy classifier; and (ii) the *calibration fraction*, denoted by cf, the fraction of calibration instances used by CP/QCCP. We vary $\mathrm{tf} \in \{0.01, 0.05, 0.10, 1.00\}$ and $\mathrm{cf} \in \{0.05, 1.00\}$ using `Qwen3-8B` in the normal setting, with three random seeds per configuration. Table 26 reports CR/SS on CosmosQA and HellaSwag under PA-QCCP with the S_{weighted} score.

On the stronger `Qwen3-8B` backbone, the failure mode at very small tf is much milder:

Table 24: Coverage Rate (CR, %) under different pooling and layer choices for `Qwen3-8B` on CosmosQA. CR is better if closer to 90%. Results are reported for both the normal and fully reworded (Full) settings. Best results are in bold.

		Last Hidden (Paper)		**Attention Weighted**		**Intermediate Layer**		**Mean Pooling**	
CP Type	**Method**	Normal	Full	Normal	Full	Normal	Full	Normal	Full
Split CP	LAC	**90.53**	**90.47**	90.60	98.87	**90.47**	98.87	**90.47**	98.87
	APS	99.07	99.20	99.40	99.50	99.60	100.00	99.60	100.00
	PA	91.00	90.97	**90.00**	**90.10**	89.27	**89.43**	88.83	**90.30**
QCCP	LAC	92.23	93.67	91.63	93.37	91.73	93.13	91.73	93.13
	APS	98.63	97.50	98.63	97.77	98.40	97.73	98.40	97.73
	PA	**91.37**	**90.97**	**90.43**	**90.10**	**90.17**	**90.37**	**89.33**	**90.77**

Table 25: Prediction set size (SS, ↓) under different pooling and layer choices for `Qwen3-8B` on CosmosQA. Lower SS indicates sharper uncertainty in Normal and fully reworded (Full) settings. Best results are in bold.

		Last Hidden (Paper)		**Attention Weighted**		**Intermediate Layer**		**Mean Pooling**	
CP Type	**Method**	Normal	Full	Normal	Full	Normal	Full	Normal	Full
Split CP	LAC	1.56	1.80	1.61	5.56	1.57	5.56	**1.57**	5.56
	APS	5.57	5.61	5.64	5.66	5.77	6.00	5.77	6.00
	PA	**1.13**	**1.17**	**1.15**	**1.12**	**1.16**	**1.12**	2.58	**2.52**
QCCP	LAC	2.47	3.11	2.35	2.91	2.39	2.87	**2.39**	2.87
	APS	4.79	4.73	4.80	4.68	4.61	4.59	4.61	4.59
	PA	**1.34**	**1.37**	**1.36**	**1.34**	**1.40**	**1.35**	2.73	**2.69**

Table 26: Low-label sensitivity of PA on `Qwen3-8B` in the normal setting under PA-QCCP. **tf** is the proxy-training fraction; **cf** is the calibration fraction. Each cell is CR / SS, mean ± std over 3 random seeds. Bold rows mark the full-budget ($\mathrm{tf} = 1$) reference.

		RC (CosmosQA)		**CI (HellaSwag)**	
tf		$\mathrm{cf} = 0.05$	$\mathrm{cf} = 1.00$	$\mathrm{cf} = 0.05$	$\mathrm{cf} = 1.00$
0.01	CR	91.16 ± 0.73	91.17 ± 0.46	92.51 ± 1.57	90.31 ± 0.10
	SS	1.859 ± 0.147	1.602 ± 0.116	2.309 ± 0.030	1.848 ± 0.128
0.05	CR	91.00 ± 2.37	91.31 ± 0.25	92.41 ± 0.94	91.14 ± 0.77
	SS	1.640 ± 0.182	1.408 ± 0.050	2.205 ± 0.037	1.802 ± 0.097
0.10	CR	90.79 ± 1.81	91.01 ± 0.16	91.06 ± 2.04	90.18 ± 0.95
	SS	1.641 ± 0.349	1.395 ± 0.054	2.126 ± 0.131	1.774 ± 0.047
1.00	CR	$\mathbf{91.20 \pm 1.78}$	$\mathbf{91.27 \pm 0.35}$	$\mathbf{92.19 \pm 2.35}$	$\mathbf{90.92 \pm 0.35}$
	SS	$\mathbf{1.470 \pm 0.129}$	$\mathbf{1.338 \pm 0.044}$	$\mathbf{1.992 \pm 0.082}$	$\mathbf{1.674 \pm 0.077}$

- At $\mathrm{tf} = 0.01$, PA already reaches near-nominal coverage (CR $\approx 91\%$ on RC, 90–93% on CI) and pays only a modest set-size penalty relative to the full-budget reference. Larger tf steadily tightens prediction sets: from $\mathrm{tf} = 0.01$ to $\mathrm{tf} = 1.00$ at $\mathrm{cf} = 1.00$, set size shrinks by 16.5% on RC ($1.602 \to 1.338$) and 9.4% on CI ($1.848 \to 1.674$).
- Increasing tf to 0.05–0.10 already recovers most of the efficiency gain, with near-target coverage in the full-calibration columns.
- Reducing the calibration fraction from 1 to 0.05 retains broadly similar empirical coverage, but increases set size by 9.9–24.9% across these cells.

This confirms that PA does not require the full labeled training budget to remain effective: a modest 5–10% proxy-training fraction is enough to nearly match the full-budget reference, while reducing the calibration budget incurs a measurable set-size cost.

F.6 Different Aggregation Scoring Rules

Finally, we analyze the behavior of the Mean, Weighted, and Worst nonconformity scores across the five general QA benchmarks, computing statistics over the calibration and test splits.

As shown in Table 27, the Worst score consistently exhibits substantially higher mean and standard deviation than the Mean and Weighted variants. Since higher nonconformity scores correspond to lower predicted probabilities (i.e., lower confidence), this pattern highlights the brittleness of LLM predictions: models often struggle with specific "hard" paraphrases, assigning high nonconformity scores to the correct answer even when other paraphrases are handled correctly. In contrast, the Mean and Weighted scores remain lower and more stable (with smaller standard deviation). This suggests that our aggregation strategy effectively smooths out the noise induced by brittle paraphrases, yielding uncertainty estimates that better reflect the model's underlying semantic knowledge rather than its sensitivity to surface-level variations.

Table 27: Score statistics for Qwen3-8B displayed as mean ± std for the correct answer choice on each dataset. These statistics are calculated using scores from both the calibration and test sets. A lower score indicates higher confidence (higher predicted probability) from the proxy model.

Dataset	Mean Score	Weighted Score	Worst Score
MMLU	0.385 ± 0.326	0.375 ± 0.332	0.534 ± 0.368
CosmosQA	0.190 ± 0.284	0.182 ± 0.287	0.286 ± 0.356
HellaSwag	0.258 ± 0.313	0.249 ± 0.317	0.371 ± 0.373
HaluDial	0.299 ± 0.310	0.288 ± 0.316	0.445 ± 0.363
HaluSum	0.068 ± 0.174	0.060 ± 0.182	0.151 ± 0.297

G Scalability and Broader Applicability

G.1 Extension to Black-Box / API-Only Settings

The white-box PA variant in the main paper relies on LLM hidden states to train a proxy classifier. To evaluate whether the paraphrase-aware idea extends to API-only deployments—where only token log-probabilities are available—we instantiate an SPUQ-inspired per-option CP adaptation that uses neither hidden states nor a proxy model.

Given a question x with paraphrases $\tilde{x}_1, \ldots, \tilde{x}_P$ and option y, suppose the API returns option probabilities $p_j(y)$ for each paraphrase $\tilde{x}_j$. We define

$$S^{\text{API}}_{\text{mean}}(x, y) \;=\; \frac{1}{P}\sum_{j=1}^{P}\big(1 - p_j(y)\big), \qquad S^{\text{API}}_{\text{worst}}(x, y) \;=\; \max_{j\in[P]}\big(1 - p_j(y)\big). \tag{8}$$

These scores rely only on option logprobs or probabilities; they do not access hidden states and do not train a proxy. We then plug them into the same PA-CP / PA-QCCP pipeline as the main method.

Table 28 reports the API-only PA variant on `Qwen3-8B` in the semi-reworded QCCP setting (the most demanding setting in our paper). The goal of this experiment is to demonstrate *applicability under black-box access*, not parity with the white-box pipeline.

Table 28: Black-box / API-only PA variant on `Qwen3-8B` in the **semi-reworded QCCP** setting. Each cell is CR / SS. The API-only variant uses option logprobs only—no hidden states and no proxy classifier—and is provided as evidence that the paraphrase-aware idea is not restricted to white-box access.

Method	QA	RC	CI	DRS	DS
LAC	93.17 / 3.72	90.47 / 2.50	91.07 / 2.63	97.87 / 5.02	94.23 / 4.30
APS	95.83 / 4.68	98.43 / 4.83	98.60 / 4.92	93.37 / 4.45	96.57 / 5.03
PA (API-only)	87.83 / **2.66**	88.43 / **1.76**	89.37 / **2.33**	93.97 / **4.38**	91.97 / **3.77**

API-only PA reduces set size substantially relative to LAC and APS on all five datasets. Coverage falls below 90% on QA, RC, and CI (87.83%, 88.43%, and 89.37%), while DRS and DS are conservative. We therefore view the API-only variant as a meaningful but *partial* extension: it shows that paraphrase-aware aggregation does not fundamentally require hidden-state access, while the absence of a proxy still leaves some robustness gap in the harder shifted cases.

G.2 Extension to Larger Backbone LLMs

We evaluate whether the advantage of PA persists as the backbone LLM scales. Specifically, we consider `Qwen3-14B` and `Qwen3-32B`, and report results under standard split CP on `CosmosQA` and `HellaSwag` across the normal, semi-reworded, and fully reworded settings.

Table 29 shows that the qualitative pattern remains consistent. Across both datasets and model sizes, PA yields substantially smaller prediction sets than LAC and APS, reducing set size from roughly 1.3–6.0 to about 1.0–1.3.

- On `CosmosQA`, PA maintains coverage close to the 90% target in the normal and fully reworded settings while preserving this efficiency gain. In the semi-reworded setting, the LAC baseline shows the largest deviations (e.g., 86.13% with the 32B model), confirming that this remains the most challenging transfer regime for that baseline.
- A similar pattern holds on `HellaSwag`. PA again achieves the smallest prediction sets, while maintaining near-nominal coverage across all three settings: in the semi-reworded setting it reaches 91.27% coverage with the 14B model and 91.33% with the 32B model.

Overall, these results show that the core benefit of PA—substantially tighter prediction sets at competitive coverage—persists at larger scales.

Table 29: Performance of nonconformity scores with larger backbone LLMs under **standard split CP**. Results are reported on RC (`CosmosQA`) and CI (`HellaSwag`) using `Qwen3-14B` and `Qwen3-32B`. Each entry is **CR / SS** (coverage rate / set size), where coverage closer to 90% is better and set size $\downarrow$ is better. Normal / Semi / Full denote **normal / semi-reworded / fully reworded**. Boldface highlights the smallest set size in each column.

	Method	**Qwen3-14B**			**Qwen3-32B**		
Dataset		Normal	Semi	Full	Normal	Semi	Full
RC	LAC	89.87 / 1.71	88.40 / 1.78	90.90 / 1.99	88.90 / 1.49	86.13 / 1.57	90.20 / 1.86
	APS	100.00 / 6.00	100.00 / 6.00	100.00 / 6.00	99.77 / 5.74	99.77 / 5.76	97.93 / 3.49
	PA	89.60 / **1.01**	88.90 / **1.01**	88.80 / **1.07**	90.23 / **1.00**	89.60 / **1.00**	90.40 / **1.01**
CI	LAC	89.40 / **1.28**	87.97 / 1.32	89.93 / 1.46	91.30 / 1.30	90.63 / 1.34	91.30 / 1.37
	APS	99.33 / 5.56	99.40 / 5.57	99.83 / 5.72	99.87 / 5.69	99.90 / 5.72	99.87 / 5.69
	PA	91.77 / 1.29	91.27 / **1.26**	89.87 / **1.19**	91.83 / **1.10**	91.33 / **1.09**	90.83 / **1.11**

G.3 Extension to Factuality Verification Tasks

While our primary framework addresses tasks with finite label spaces (i.e., $|\mathcal{Y}| < \infty$), it naturally extends to critical sub-problems within open-ended generation, specifically *factuality verification*. Although the space of generated text is theoretically infinite, modern evaluation protocols typically decompose long-form generations into discrete atomic claims, each evaluated independently against a knowledge source [26, 5].

This decomposition effectively reduces the verification problem to a binary classification task. Formally, let an open-ended generation be parsed into a set of atomic claims $\{c_1, c_2, \ldots, c_k\}$. We treat each claim c_i as an input $x \in \mathcal{X}$, where the label space is restricted to $\mathcal{Y} = \{\text{True}, \text{False}\}$.

Our paraphrase-robust framework is particularly well-suited for this setting due to the sensitivity of LLM-based evaluators to surface-level phrasing. To apply our method:

1. **Paraphrase Generation:** For a given claim x, we generate a set of semantic paraphrases $B(x)$.
2. **Proxy Scoring:** We utilize our lightweight proxy model $P_\theta(y \mid E(x))$ to predict the truthfulness of the claim based on the embeddings of these paraphrases.
3. **Robust Uncertainty:** We calculate PA scores across $B(x)$.

By integrating these scores into the conformal prediction framework, we ensure that the resulting uncertainty estimates, as well as the subsequent decision to flag a claim as hallucinated, are robust to lexical diversity. This guarantees that a claim is not penalized merely due to the specific syntax chosen by the LLM, but rather evaluated based on its semantic validity.

Preliminary experiment on claim-verification. To provide concrete evidence beyond MCQA, we conducted a preliminary claim-verification experiment on FEVER [39], a standard benchmark for factuality verification. We evaluated the same three settings used in the main paper—*clean*, *semi-reworded*, and *fully-reworded*—and report coverage and average set size at $\alpha = 0.1$. As in the main text, coverage closer to 0.9 is better, while smaller set size is preferred.

Table 30 and Table 31 show that the proposed PA score transfers meaningfully to this open-ended verification setting under `Qwen3-8B`. On FEVER, the backbone is sharp enough that all non-APS methods return essentially singleton prediction sets, so the main distinction is whether coverage stays closest to the nominal target under shift. PA remains the most stable method across settings, especially in the fully reworded case, where it reaches 90.5% under Split CP and 90.3% under QCCP, while LAC/TRON/SPUQ dip slightly below 90%. APS remains overly conservative, with prediction sets close to the full label set. These results suggest that the benefit of paraphrase-aware scoring is not specific to MCQA, but can also improve robustness in claim-level factuality verification, which serves as a practically relevant building block for open-ended generation evaluation.

Table 30: Preliminary extension to open-ended generation on FEVER under *Split CP* at $\alpha = 0.1$, with **Qwen3-8B** as the backbone. Coverage Rate (CR, %) closer to 90% is better; Set Size (SS) $\downarrow$ is better. Results are reported under the **clean**, **semi-reworded (Semi.)**, and **Fully-reworded (Full)** settings. TRON-adapt. and SPUQ-adapt. denote the per-option probability-score controls here; the probability-score adaptation equals LAC for any label count, so these two rows match by construction.

Method	Clean		Semi.		Full	
	CR	SS	CR	SS	CR	SS
LAC	93.7	1.000	90.5	1.000	89.7	1.000
APS	100.0	1.910	100.0	1.965	100.0	1.993
TRON-adapt.	93.7	1.000	90.5	1.000	89.7	1.000
SPUQ-adapt.	93.7	1.000	90.7	1.000	89.4	1.000
PA	94.0	1.000	91.2	1.000	90.5	1.000

Table 31: Preliminary extension to open-ended generation on FEVER under *QCCP* at $\alpha = 0.1$, with **Qwen3-8B** as the backbone. Coverage Rate (CR, %) closer to 90% is better; Set Size (SS) $\downarrow$ is better. Results are reported under the **clean**, **semi-reworded (Semi.)**, and **Fully-reworded (Full)** settings. TRON-adapt. and SPUQ-adapt. denote the per-option probability-score controls here; the probability-score adaptation equals LAC for any label count, so these two rows match by construction.

Method	Clean		Semi.		Full	
	CR	SS	CR	SS	CR	SS
LAC	93.7	1.000	90.5	1.000	89.9	1.002
APS	100.0	1.947	99.8	1.982	100.0	1.987
TRON-adapt.	93.7	1.000	90.5	1.000	89.9	1.002
SPUQ-adapt.	93.5	1.000	90.5	1.000	89.7	1.005
PA	94.5	1.000	90.7	1.000	90.3	1.000

Overall, we view this experiment as a preliminary but encouraging step toward broader open-ended applications. A full treatment of long-form generation, including claim decomposition and aggregation back to response-level guarantees, is an important direction for future work.

G.4 Extension to Image Modality

Although our experiments focus on text-based tasks, the proposed framework is not limited to textual inputs and can, in principle, be extended to vision–language settings such as visual question answering (VQA). The key requirement of our method is access to a stable hidden representation for each input variant, which is naturally provided by modern vision–language models (VLMs).

Concretely, given an image–question pair, we treat the visual input as part of the conditioning context and apply the same paraphrase-aware uncertainty pipeline at the representation level. In lieu of

textual paraphrases, visual perturbations can be used to induce semantic-preserving variants of the input image. For example, this can be realized by adding Gaussian noise to the image, producing multiple visually indistinguishable variants that preserve the underlying semantic content while altering surface-level pixel statistics.

Each perturbed image is processed by a frozen VLM encoder, and we extract the final-layer hidden representation corresponding to the answer token, mirroring the text-only pipeline. Paraphrase-aware (variant-aware) nonconformity scores are then computed across these image variants and integrated into the same conformal prediction framework without modification.

H Limitations and Broader Impacts

Limitations. The strongest variant requires hidden states and labeled proxy-training data. Online paraphrase generation is a material cost: the B200 six-view measurements are 22.7–83.4 times slower than raw single-input inference (Appendix E.6). The blackbox extension shows applicability, but still not in parity with the white-box model (Appendix G.1). Arbitrary test-only rewriting is still outside the coverage guarantee. Our framework relies on the quality of the paraphrase generator: if generated paraphrases exhibit semantic drift or limited diversity, the effectiveness of the paraphrase-aware score may degrade. Although we observe stable behavior across several paraphrase generators, domain-specific settings may require stronger or more specialized paraphrasing models. More broadly, our work focuses on semantic-preserving perturbations induced by paraphrasing and does not explicitly address other forms of distribution shift, such as topic drift, temporal changes in factual knowledge, or adversarial perturbations. In addition, our experiments are primarily conducted on English multiple-choice QA benchmarks. Extending paraphrase-aware conformal prediction to multilingual settings and free-form generation tasks remains an important direction for future work.

Broader impacts. Our method improves the stability and reliability of LLM uncertainty estimates under semantic rewording, which may be particularly helpful for users whose phrasing differs from typical training distributions, such as non-native speakers or users of regional dialects. At the same time, prediction sets produced by conformal methods should not be interpreted as guarantees of correctness in high-stakes applications, but rather as uncertainty-aware decision support.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state that we identify paraphrase-induced instability in LLM uncertainty quantification, propose paraphrase-aware nonconformity scores using a lightweight proxy model, and demonstrate improved coverage–efficiency tradeoffs across multiple MCQA benchmarks and LLM families. These claims are supported by the method in Sections 4–5, the empirical results in Tables 3, 4, and 5, and the additional analyses in Appendix E.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated discussion is provided in Appendix H. It covers the approximate nature of distributional closedness under realistic paraphrase generators, dependence on paraphrase quality, the MCQA-centered evaluation scope, white-box hidden-state access, black-box adaptation costs, additional inference-time compute, and remaining exchangeability assumptions under broader distribution shifts.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The main theoretical assumptions and statements are introduced in Section 5, including Assumption 5.1, Proposition 5.1, and Lemma 5.1. Full formal coverage statements and proofs are provided in Appendix B, Appendix C, and Appendix C.3.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper?

   Answer: [Yes]

   Justification: The experimental protocol, datasets, models, and conformal prediction setup are described in the empirical section and Appendix D. Dataset preparation is detailed in Appendix D.1, paraphrase generation and implementation details are given in Appendix D.2 and Appendix D.4, and hyperparameters are reported in Appendix D.5.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: An GitHub code repository is provided at `https://github.com/Raina-Xin/PA_Score`, including the paraphrase generation pipeline, proxy-model training scripts, PA score computation, and split CP/QCCP evaluation harness. The datasets used in the experiments are publicly available and documented in Appendix D.1.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details necessary to understand the results?

   Answer: [Yes]

Justification: The paper specifies the train/calibration/test split, nominal level $\alpha = 0.1$, model choices, paraphrase settings, proxy-model architecture, calibration loss, and QCCP clustering procedure. Full implementation details and hyperparameter grids are provided in Appendix D.4 and Appendix D.5.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports finite-sample coverage diagnostics and Wilson 95% confidence intervals for cluster-conditional coverage in Appendix E.4 and the class-conditional coverage analysis around Figure 4. Additional sensitivity experiments report mean $\pm$ standard deviation over random seeds, for example in Table 26 and Table 27.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Runtime and compute-resource details are reported in Appendix E.6 and Table 17. The complete online timing table uses one NVIDIA B200 and reports batch sizes, paraphrase budgets, raw single-input inference and cached scoring separately. Online six-view PA is 22.7–83.4 times slower than raw single-input inference; cached scoring excludes paraphrase generation and representation extraction.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics and confirm that the research complies with it. The experiments use publicly available benchmarks and pretrained models under their respective licenses, do not involve human subjects or personally identifiable data, and do not release new high-risk generative models or scraped datasets.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impacts are discussed in Appendix H. Positive impacts include more reliable uncertainty estimates for users whose phrasing differs from typical training distributions and for high-stakes domains, while negative considerations include false confidence in coverage guarantees, lack of adversarial robustness, and increased inference-time compute from paraphrase generation.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?

    Answer: [N/A]

    Justification: The paper releases evaluation code and lightweight proxy-model training/evaluation pipelines, but does not release new pretrained generative models, scraped datasets, or other artifacts that create elevated misuse risk beyond the upstream LLMs and public benchmarks already used under their licenses.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets used in the paper properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: Existing assets are cited throughout the paper. The base models include Qwen models [51], Llama-3.1 [9], and Phi-3 [1]; sentence embeddings use SBERT [31]; datasets include MMLU [15], CosmosQA [16], HellaSwag [54], HaluEval-derived tasks [23], MedMCQA [28], and MedQA [19]; and QCCP uses `conditionalconformal` [13]. These assets are used for research evaluation in accordance with their published terms.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new assets are the paraphrase-aware nonconformity score, proxy-model training pipeline, and evaluation harness for PA, LAC, APS, TRON, and SPUQ under split CP and QCCP. These are documented in the GitHub code release at `https://github.com/Raina-Xin/PA_Score`; the paper does not release new datasets or pretrained model weights.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing or research with human subjects. All paraphrases are generated programmatically by LLMs, and all evaluations use public benchmark datasets collected by their original authors.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed, and whether IRB approvals or equivalent review were obtained?

    Answer: [N/A]

    Justification: This work does not involve research with human subjects, so IRB review was not required. The medical MCQA experiments use publicly released exam-style benchmarks, MedMCQA and MedQA, solely for evaluation and do not involve clinical or patient data.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [Yes]

    Justification: LLMs are central to the methodology: Qwen3-8B is used for paraphrase generation in Appendix D.4, and Qwen, Llama, and Phi models provide logits and hidden states for the proxy classifier. Their roles, prompting setup, temperature, paraphrase budget, hidden-state extraction, and filtering pipeline are described in Appendix D.2 and Appendix D.4.