BEYOND SAMPLED ANSWERS: CONFORMAL PREDICTION FOR LANGUAGE MODELS VIA LATENT-SPACE EXPANSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Conformal prediction (CP) provides prediction sets with finite-sample marginal coverage at level $1-\alpha$ and has recently been applied to language models (LMs). For open-ended generation, sampling-based CP scores calibration examples using the empirical frequency of their ground-truth answers across repeated generations. However, if repeated generation misses the ground-truth answer for more than an α fraction of calibration inputs, these methods may fail to attain the target coverage or return the uninformative entire answer space. To recover ground-truth answers missed during generation, we introduce a hybrid nonconformity score that combines empirical generation frequencies with discrepancies measured in an encoder-induced latent space. The hybrid prediction set therefore extends beyond sampled answers through a latent region that permits checking whether an answer is included but does not enumerate the answers it covers. We enumerate the additional answers covered by this region within a finite vocabulary to obtain an explicit semantic prediction set. To reduce set size, we partition the discrepancy space into cells and prioritize those that recover more missed ground-truth answers while admitting fewer incorrect vocabulary answers. Under exchangeability, the hybrid prediction set guarantees marginal coverage of at least $1-\alpha$. At a target coverage of 0.9, its empirical coverage averages 90.25% across five datasets versus 39.59% for sampling-based CP. The semantic prediction set retains this guarantee when the vocabulary contains all missed ground-truth answers.

1 INTRODUCTION

Building on advances in the Transformer architecture (Vaswani et al., 2017), language models (LMs) have developed remarkable linguistic and reasoning capabilities, emerging as general-purpose engines for a wide range of tasks, including agentic systems (Luo et al., 2025) and robotics (Zeng et al., 2025). Their strong empirical performance in natural language processing, planning, and tool use, together with their scalability, has led to widespread adoption in practice (Chatterji et al., 2025). Despite these advances, concerns regarding the reliability of LMs remain significant, spanning issues such as hallucinated factual content (Huang et al., 2025), inconsistent reasoning (Uceda-Sosa et al., 2024), and sensitivity to prompt variations (Sclar et al., 2024). In this work, we focus on one particularly important aspect of reliability, factual hallucination, where models generate confidently stated but incorrect information when faced with unfamiliar queries (Kalai et al., 2025). This issue is especially critical in high-stakes applications such as legal services (Li et al., 2025) and healthcare (Yang et al., 2024). Consequently, ensuring the reliability and trustworthiness of LMs has become a central challenge in their development.

A broad class of uncertainty quantification (UQ) methods has been developed to identify unreliable LM responses (Shorinwa et al., 2025; Liu et al., 2025). Among them, conformal prediction (CP) has emerged as a prominent model-agnostic framework (Vovk et al., 2005; Angelopoulos & Bates, 2021; Papadopoulos et al., 2002). CP computes a nonconformity score for each labeled calibration example and uses the empirical $(1-\alpha)$-quantile of these scores as a threshold to construct prediction sets that contain the ground-truth answer with marginal probability at least $1-\alpha$. In open-ended LM generation, sampling-based CP instantiates this principle through repeated generation. It is particularly useful for *short-form question answering*, where outputs can be readily aggregated into

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

distinct answers with empirical frequencies (Wang et al., 2024a;b; Su et al., 2024; Kaur et al., 2024). During calibration, each example's nonconformity score incorporates the empirical frequency of its ground-truth answer, with higher frequencies indicating lower nonconformity. However, restricting prediction sets to generated answers limits coverage to sampling recall: if repeated generation recovers the ground-truth answer with probability below $1 - \alpha$, the target coverage cannot be attained without extending the set beyond generated answers. The trivial fallback of returning the entire answer space ensures coverage but yields uninformative predictions (Wang et al., 2025). This limitation is particularly relevant for lower-performing models or difficult tasks.

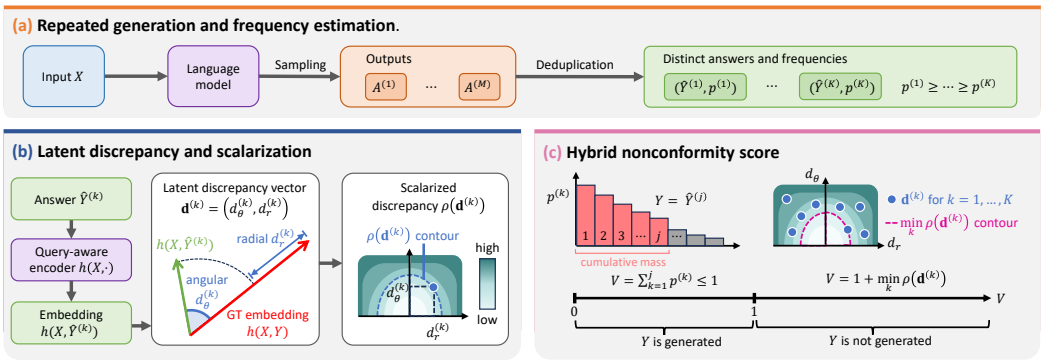

Figure 1: **Overview of the hybrid nonconformity score.** (a) Repeated LM outputs are deduplicated into an answer set ordered by empirical generation frequency. (b) Each generated answer $\widehat{Y}^{(k)}$ is paired with the input X and processed by the query-aware encoder $h(X, \cdot)$, producing the embedding $h(X, \widehat{Y}^{(k)})$. Comparing it with the ground-truth (GT) embedding $h(X, Y)$ yields the angular and radial discrepancies $\mathbf{d}^{(k)} = (d_\theta^{(k)}, d_r^{(k)})$, which are scalarized by ρ. (c) Empirical frequencies and the minimum scalarized discrepancy define the $[0, 1]$ and > 1 portions of a unified scale for nonconformity score V.

We supplement frequency-based scoring with discrepancies in an encoder-induced latent space to extend prediction sets beyond LM-generated answers. For each calibration input, we repeatedly sample the LM, aggregate duplicate outputs into distinct generated answers, and estimate the generation frequency of each. We then compare the ground-truth answer with every distinct generated answer using angular and radial discrepancies between their embeddings, and scalarize each two-dimensional discrepancy vector. If the ground-truth answer is generated, its nonconformity score is the corresponding cumulative-frequency value in $[0, 1]$; otherwise, it is one plus the minimum scalarized discrepancy across these comparisons. The one-unit offset places the discrepancy contribution beyond the frequency range, allowing calibration to account for both generated and missed ground-truth answers on a common nonconformity scale. Figure 1 summarizes this construction[1]. Accordingly, the prediction set has a hybrid representation, comprising generated answers selected by their empirical frequencies and a latent region representing additional possible answers. To obtain an explicit semantic prediction set, we retain the selected generated answers and convert the latent region into textual answers drawn from a finite vocabulary. This conversion may nevertheless produce large semantic sets. To improve set efficiency, we partition the two-dimensional discrepancy space into Voronoi cells and learn an ordering that controls the expansion of the latent region. We rank the cells according to their ability to recover missed ground-truth answers while admitting few incorrect answers. Cells are then added from most to least efficient, and independent conformal recalibration determines how many cells are required to attain the target coverage.

Under exchangeability, the hybrid prediction set guarantees finite-sample marginal coverage of at least $1 - \alpha$. At a target coverage of $1 - \alpha = 0.9$, it achieves an average empirical coverage of 90.25% across five datasets and two inference LMs, compared with 39.59% averaged across the four sampling-based CP baselines. On recoverable versions of the datasets, where the vocabulary contains every ground-truth answer missed during generation, the semantic prediction set retains this guarantee and achieves an average empirical coverage of 90.07%. Scalarization through cell ordering reduces the mean semantic prediction set size by 20.02% compared with scalarizing each discrepancy vector using its ℓ_2 norm.

[1] In Figure 1, we write $\mathbf{d}^{(k)}$ for $\mathbf{d}^{(k)}(X, Y)$ to simplify the illustration.

2 Related work

Uncertainty quantification in LM generation has been estimated using token-level probabilities and predictive entropy (Margatina et al., 2023; Manakul et al., 2023; Fadeeva et al., 2024; Duan et al., 2023), verbalized confidence (Kadavath et al., 2022), semantic consistency across repeated generations (Kuhn et al., 2023), and internal attention patterns or hidden representations (Zhang et al., 2023; Chen et al., 2024; Sriramanan et al., 2024; Ren et al., 2023b). Although these approaches yield useful uncertainty signals, the scale and interpretation of these signals can vary across models and tasks, making them difficult to translate into actionable reliability statements. Conformal prediction (CP) addresses this limitation by using held-out calibration data to construct prediction sets with a direct probabilistic guarantee of containing the ground truth (Kaur et al., 2024).

Early applications of CP to LMs primarily considered structured output spaces, including multiple-choice question answering (Kumar et al., 2023; Ren et al., 2023a; Kostumov et al., 2024) and numerical grading (Sheng et al., 2025). In open-ended language generation, sampling-based CP constructs prediction sets from repeated LM generations, often incorporating empirical generation frequencies into nonconformity scores. When prediction sets contain only generated answers, their coverage cannot exceed sampling recall. Existing methods differ in how they treat this limitation. TRON (Wang et al., 2024a) and ConU (Wang et al., 2024b) require at least one correct generated answer for every calibration input. Kaur et al. (2024) instead restrict calibration to clusters containing only correct answers. LofreeCP assigns infinite scores to calibration examples with missed ground-truth answers; an infinite threshold admits all generated answers but cannot increase coverage beyond sampling recall (Su et al., 2024). SConU analyzes the coverage limitation imposed by sampling and discusses the uninformative fallback of returning the entire output space when generated answers cannot support the target coverage (Wang et al., 2025). Our method addresses this limitation by extending prediction sets beyond generated answers through an encoder-induced latent region, whose extent is determined by conformal calibration.

A separate line of work decomposes each generated answer into atomic claims and removes claims that cannot be certified as supported or entailed (Cherian et al., 2024; Mohri & Hashimoto, 2024); related methods combine claim decomposition with repeated generation (Quach et al., 2023). Claim filtering is particularly suitable for *long-form question answering* containing multiple independently verifiable claims, although it may remove every claim when none can be certified. Our method instead treats each complete answer as the prediction unit and expands the set of answers to recover ground-truth answers missed during generation. Claim-filtering methods therefore address a complementary objective and are not directly comparable baselines.

3 Hybrid conformal prediction

3.1 Hybrid nonconformity score

Let (X, Y) be a labeled example taking values in $\mathcal{X} \times \mathcal{Y}$, where X is an input sequence and Y is its ground-truth answer. Given X, the LM induces a conditional distribution $\Pr_{\mathrm{LM}}(\cdot \mid X)$ over possible output sequences. We independently draw M outputs via repeated generation such that $A^{(1)}, \ldots, A^{(M)} \overset{\text{i.i.d.}}{\sim} \Pr_{\mathrm{LM}}(\cdot \mid X)$. After deduplicating these outputs, let $\widehat{\mathcal{Y}}(X) = \{\widehat{Y}^{(1)}, \ldots, \widehat{Y}^{(K)}\}$ denote the set of K distinct answers. Let $\mathbf{1}\{\cdot\}$ denote the indicator function. The empirical generation frequency of $\widehat{Y}^{(k)}$ is given by

$$p^{(k)} = \frac{1}{M} \sum\nolimits_{m=1}^{M} \mathbf{1}\{A^{(m)} = \hat{Y}^{(k)}\}. \tag{1}$$

Frequency-based scoring Su et al. (2024) use the empirical generation frequency of the ground-truth answer Y as a proxy for its model probability when computing the nonconformity score. Following this principle, we order the K distinct generated answers such that $p^{(1)} \geq p^{(2)} \geq \cdots \geq p^{(K)}$.

We then adapt the cumulative-mass score of adaptive prediction sets (Romano et al., 2020), replacing class probabilities with empirical generation frequencies. Specifically, we define the cumulative-frequency score of the k-th ranked answer as

$$P^{(k)} = \sum\nolimits_{\ell=1}^{k} p^{(\ell)}, \qquad k = 1, \ldots, K. \tag{2}$$

Because the distinct answers exhaust the M sampled outputs, $P^{(K)} = 1$. Consequently, answers ranked earlier by empirical frequency receive smaller cumulative-frequency scores. When the ground-truth answer appears in $\widehat{\mathcal{Y}}(X)$, its rank j in the frequency ordering determines the nonconformity score according to

$$Y = \widehat{Y}^{(j)} \quad \Longrightarrow \quad V = P^{(j)}. \tag{3}$$

Discrepancy-based scoring When the ground-truth answer is absent from $\widehat{\mathcal{Y}}(X)$, the frequency-based scoring cannot assign it a finite value because no corresponding frequency rank exists. Rather than assigning $+\infty$, we use the geometry of encoder embeddings to quantify how far the missed answer lies from the distinct answers generated by the LM.

Because an answer may be ambiguous without its input, a statement-generation model g converts (X, Y) into a self-contained statement, which an encoder f maps to the ground-truth embedding:

$$h(X, Y) = f(g(X, Y)). \tag{4}$$

For the k-th distinct generated answer, we similarly define $h^{(k)} = h(X, \widehat{Y}^{(k)}) = f(g(X, \widehat{Y}^{(k)}))$.

A single geometric quantity can obscure the different ways in which two representations depart from one another. We therefore retain their directional and magnitude differences as separate quantities. In particular, assuming that the encoder embeddings have nonzero norms, define the angular and radial discrepancies between Y and $\widehat{Y}^{(k)}$ by

$$d_\theta^{(k)}(X, Y) = \arccos\left(\frac{h(X, Y)^\top h^{(k)}}{\|h(X, Y)\|_2 \|h^{(k)}\|_2}\right), \quad d_r^{(k)}(X, Y) = \frac{\|h(X, Y)\|_2 - \|h^{(k)}\|_2}{\|h^{(k)}\|_2}. \tag{5}$$

The two quantities form a discrepancy vector relative to the generated answer $\widehat{Y}^{(k)}$:

$$\mathbf{d}^{(k)}(X, Y) = \left[d_\theta^{(k)}(X, Y),\, d_r^{(k)}(X, Y)\right]^\top \in \mathbb{R}^2. \tag{6}$$

To summarize the discrepancy vector by a nonnegative value, we introduce a scalarization function $\rho : \mathbb{R}^2 \to \mathbb{R}_+$. We require the scalarization to satisfy $\rho(\mathbf{0}) = 0$ and $\rho(\mathbf{z}) > 0$ for $\mathbf{z} \neq \mathbf{0}$. By construction, if $Y = \widehat{Y}^{(j)}$, then $\mathbf{d}^{(j)}(X, Y) = \mathbf{0}$, and hence $\rho\big(\mathbf{d}^{(j)}(X, Y)\big) = 0$. Conversely, we assume that a ground-truth answer absent from $\widehat{\mathcal{Y}}(X)$ has nonzero discrepancy vectors. Thus, if $Y \notin \widehat{\mathcal{Y}}(X)$, $\rho\big(\mathbf{d}^{(k)}(X, Y)\big) > 0$, for $k = 1, \ldots, K$.

One simple scalarization satisfying the required positivity condition is the ℓ_2 norm, which combines angular and radial discrepancies into a scalar measure of geometric proximity (Klein et al., 2025). In particular, for a discrepancy vector $\mathbf{d}^{(k)}(X, Y) \in \mathbb{R}^2$, we define

$$\rho_{\text{norm}}(\mathbf{d}^{(k)}(X, Y)) = \|\mathbf{d}^{(k)}(X, Y)\|_2. \tag{7}$$

We retain the generic notation ρ in the following construction, which applies to any admissible scalarization. If the ground-truth answer is not generated, we apply ρ to its discrepancy vector relative to each distinct generated answer and retain the smallest value. Thus, its nonconformity score is given by

$$Y \notin \widehat{\mathcal{Y}}(X) \quad \Longrightarrow \quad V = 1 + \min_{1 \le k \le K} \rho\Big(\mathbf{d}^{(k)}(X, Y)\Big). \tag{8}$$

The minimum selects the most favorable comparison available among the generated answers. The offset of one places this value beyond the complete frequency range, since $P^{(K)} = 1$.

Hybrid nonconformity score The frequency-based scoring in Eq. (3) and discrepancy-based scoring in Eq. (8) can now be combined as consecutive stages of a single set expansion. To define this expansion over the entire answer space $\mathcal{Y}$, let y denote an arbitrary possible answer, and define $h(X, y)$ and $\mathbf{d}^{(k)}(X, y)$ analogously to their ground-truth counterparts in Eq. (4) and Eq. (6), respectively. For an expansion level $q \ge 0$, define the following nested family:

$$\begin{aligned} \mathcal{C}_\rho^{\mathcal{Y}}(X; q) = &\left\{\widehat{Y}^{(k)} \in \widehat{\mathcal{Y}}(X) : P^{(k)} \le q\right\} \\ &\cup \left\{y \in \mathcal{Y} \setminus \widehat{\mathcal{Y}}(X) : \min_{1 \le k \le K} \rho\Big(\mathbf{d}^{(k)}(X, y)\Big) \le q - 1\right\}. \end{aligned} \tag{9}$$

For $0 \le q < 1$, the second set in Eq. (9) is empty, while the first progressively includes distinct generated answers according to their cumulative frequencies. At $q = 1$, all answers in $\widehat{\mathcal{Y}}(X)$ have entered the family because $P^{(K)} = 1$. For $q > 1$, the first set remains unchanged, whereas the second expands beyond the generated answers according to their scalarized discrepancies. Moreover, the family is nested because increasing q preserves all previously included answers.

The hybrid nonconformity score of the ground-truth answer is the smallest expansion level at which it enters the nested family:

$$s_\rho^{\mathcal{Y}}(X, Y) = \inf \left\{ q \ge 0 : Y \in \mathcal{C}_\rho^{\mathcal{Y}}(X; q) \right\}. \tag{10}$$

3.2 Conformal calibration and the hybrid prediction set

We next calibrate the proposed score to determine how far the prediction set must expand. We first consider a scalarization ρ that is fixed independently of the calibration examples.

Let $\mathcal{D}_{\text{cal}} = \{(X_i, Y_i)\}_{i=1}^n$ be a calibration set. For each input X_i, we independently repeat the sampling and deduplication procedure, obtaining $\widehat{\mathcal{Y}}(X_i) = \{\widehat{Y}_i^{(1)}, \ldots, \widehat{Y}_i^{(K_i)}\}$, where K_i is the number of distinct generated answers and may vary across inputs. We then compute the nonconformity score

$$V_i = s_\rho^{\mathcal{Y}}(X_i, Y_i), \qquad i = 1, \ldots, n. \tag{11}$$

Every calibration example receives a finite hybrid nonconformity score, even when its ground-truth answer is not generated. For a target miscoverage level α[2], let τ denote the finite-sample corrected $(1-\alpha)$-quantile of $\{V_i\}_{i=1}^n$, namely their $\lceil (1-\alpha)(n+1) \rceil$-th order statistic.

For a test input X_{n+1}, we again draw M outputs and construct its distinct answer set $\widehat{\mathcal{Y}}(X_{n+1})$ with K_{n+1} answers. Evaluating the nested family at the threshold τ gives the hybrid prediction set:

$$\mathcal{C}_\rho^{\mathcal{Y}}(X_{n+1}; \tau) = \{ y \in \widehat{\mathcal{Y}}(X_{n+1}) \cup \mathcal{Y} : s_\rho^{\mathcal{Y}}(X_{n+1}, y) \le \tau \} = \{ y \in \mathcal{Y} : s_\rho^{\mathcal{Y}}(X_{n+1}, y) \le \tau \}, \tag{12}$$

where the second equality follows from $\widehat{\mathcal{Y}}(X_{n+1}) \subseteq \mathcal{Y}$. Algorithm 1 presents the pseudocode for hybrid conformal prediction.

For a fixed scalarization ρ, independently applying the same repeated-generation procedure to every calibration and test input preserves exchangeability of the induced nonconformity scores. Consequently, under exchangeability of the labeled examples, the standard split-conformal argument gives

$$\Pr\big(Y_{n+1} \in \mathcal{C}_\rho^{\mathcal{Y}}(X_{n+1}; \tau)\big) \ge 1 - \alpha. \tag{13}$$

4 Semantic prediction set

4.1 Semantic prediction set construction and coverage

The hybrid prediction set in Eq. (12) combines selected generated answers with a region in the encoder-induced latent space. Although an answer's inclusion in this region can be checked through its embedding, the region does not explicitly enumerate the answers it covers as text.

To obtain an explicit semantic prediction set, let $\mathcal{W} \subseteq \mathcal{Y}$ be a finite vocabulary of admissible answers. We restrict the additional answers represented by the latent region to $\mathcal{W}$, while retaining the nonconformity score $s_\rho^{\mathcal{Y}}$ and its calibrated threshold τ. The semantic prediction set is

$$\mathcal{C}_\rho^{\mathcal{Y} \to \mathcal{W}}(X_{n+1}; \tau) = \left\{ y \in \widehat{\mathcal{Y}}(X_{n+1}) \cup \mathcal{W} : s_\rho^{\mathcal{Y}}(X_{n+1}, y) \le \tau \right\}. \tag{14}$$

Here, $\mathcal{Y} \to \mathcal{W}$ indicates that calibration uses the entire answer space $\mathcal{Y}$, while the latent region is converted to textual answers drawn from $\mathcal{W}$. Generated answers remain eligible regardless of their membership in $\mathcal{W}$ and are selected by the same inclusion rule.

By Eq. (14), a coverage failure occurs only if $s_\rho^{\mathcal{Y}}(X_{n+1}, Y_{n+1})$ exceeds τ, or if the ground-truth answer is neither generated nor contained in $\mathcal{W}$. Define the probability of the latter event as

$$\delta_{\mathcal{W}} = \Pr\Big(Y_{n+1} \notin \widehat{\mathcal{Y}}(X_{n+1}),\ Y_{n+1} \notin \mathcal{W}\Big). \tag{15}$$

[2]We assume $\alpha \ge 1/(n+1)$ so that the quantile index does not exceed the number of calibration examples.

Combining the coverage guarantee of the hybrid prediction set with a union bound gives

$$\Pr\big(Y_{n+1} \in \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1};\tau)\big) \geq 1-\alpha-\delta_{\mathcal{W}}. \tag{16}$$

If every ground-truth answer missed during generation belongs to $\mathcal{W}$, then $\delta_{\mathcal{W}} = 0$, and the semantic prediction set retains marginal coverage of at least $1-\alpha$. The lower bound in Eq. (16) can be conservative; we discuss its tightness and provide a detailed coverage analysis in Appendix A.

In addition to coverage, we seek smaller semantic prediction sets. For a fixed vocabulary $\mathcal{W}$, the scalarization ρ determines which additional answers are admitted at each expansion level q and thus influences set size after calibration. With $\rho = \rho_{\text{norm}}$ in Eq. (14), an ungenerated vocabulary answer is included when the minimum norm of its discrepancy vectors does not exceed $\tau - 1$. Any threshold that recovers a missed ground-truth answer therefore also admits incorrect vocabulary answers with smaller minimum norms, potentially producing a large semantic prediction set.

4.2 Cell scalarization

We propose ρ_{cell} by partitioning the two-dimensional discrepancy space into cells and learning an ordering that prioritizes cells recovering more missed ground-truth answers while admitting fewer incorrect vocabulary answers. Through the mapping from latent embeddings to discrepancy vectors, this ordering controls the expansion of the latent region, with the aim of reducing semantic prediction set size. Learning the cell partition and its ordering requires labeled examples, while conformal calibration must evaluate the learned scalarization on separate data. We therefore divide the available labeled examples into a cell-construction set $\mathcal{D}_{\text{cell}} = \{(X_i, Y_i)\}_{i=1}^{c}$ and a held-out recalibration set $\mathcal{D}_{\text{recal}} = \{(X_i, Y_i)\}_{i=c+1}^{n}$. The former determines the partition and ordering; the latter is used to calibrate the threshold τ_{cell} after ρ_{cell} has been fixed.

Cell partition. We construct the cells around discrepancy vectors associated with ground-truth answers that can be newly recovered from the vocabulary. For each $(X_i, Y_i) \in \mathcal{D}_{\text{cell}}$, we apply the same repeated-generation procedure and compare Y_i with each of the K_i distinct generated answers, obtaining $\mathbf{d}^{(k)}(X_i, Y_i)$ for $k = 1, \ldots, K_i$. We use all K_i discrepancy vectors from this example as candidate cell centers only if its ground-truth answer Y_i belongs to the vocabulary $\mathcal{W}$ but is absent from the generated answer set $\widehat{\mathcal{Y}}(X_i)$. This focuses cell construction on missed ground-truth answers that can be recovered through the vocabulary.

Candidate centers from different examples may coincide. Since coincident centers would induce the same cell, we retain one copy of each distinct vector and denote the centers by

$$\{\boldsymbol{\mu}_\ell\}_{\ell=1}^{L} = \left\{\mathbf{d}^{(k)}(X_i, Y_i) : 1 \leq i \leq c,\ Y_i \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i),\ 1 \leq k \leq K_i\right\}. \tag{17}$$

These centers induce a Voronoi partition of the two-dimensional discrepancy space by assigning each vector to its nearest center:

$$\mathcal{B}_\ell = \left\{\mathbf{z} \in \mathbb{R}^2 : \ell = \operatorname*{arg\,min}_{1\leq j\leq L} \|\mathbf{z} - \boldsymbol{\mu}_j\|_2\right\}, \qquad \ell = 1, \ldots, L. \tag{18}$$

Cell ordering. Although the cell centers are constructed from missed ground-truth answers, the cells may also contain discrepancy vectors of incorrect vocabulary answers. Learning an ordering therefore requires examining how both types of answers are distributed across the cells. For each input X_i, we compare every ungenerated vocabulary answer $y \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)$ with the K_i distinct generated answers. These comparisons produce discrepancy vectors that may lie in different cells. We denote the set of indices of these cells by

$$\Gamma_i(y) = \left\{\ell \in \{1, \ldots, L\} : \exists k \in \{1, \ldots, K_i\} \text{ such that } \mathbf{d}^{(k)}(X_i, y) \in \mathcal{B}_\ell\right\}. \tag{19}$$

Starting from an empty collection, we construct the ordering by adding one cell at each step, favoring collections with fewer incorrect vocabulary answers per recovered ground-truth answer. Let $\mathcal{I}_t \subseteq \{1, \ldots, L\}$ denote the indices of the cells selected after t steps, with $\mathcal{I}_0 = \varnothing$.

The number of missed ground-truth answers recovered from the vocabulary $\mathcal{W}$ at step t is

$$T(\mathcal{I}_t) = \sum_{i=1}^{c} \mathbf{1}\left\{Y_i \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)\right\} \mathbf{1}\{\Gamma_i(Y_i) \cap \mathcal{I}_t \neq \varnothing\}. \tag{20}$$

The first indicator requires that $Y_i \in \mathcal{W}$ but was not generated for X_i. The second checks whether $\Gamma_i(Y_i) \cap \mathcal{I}_t \neq \varnothing$. A nonempty intersection means that Y_i has already been admitted by selected cells indexed by $\mathcal{I}_t$, so selecting further cells that admit Y_i does not count as an additional recovery.

Selecting cells may also add incorrect vocabulary answers to the semantic prediction set, increasing its size without improving coverage. Their accumulated count after t steps is

$$F(\mathcal{I}_t) = \sum_{i=1}^{c} \sum_{y \in \mathcal{W} \setminus \left(\widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}\right)} \mathbf{1}\{\Gamma_i(y) \cap \mathcal{I}_t \neq \varnothing\}. \tag{21}$$

The inner sum excludes the ground-truth answer and previously generated answers, so only additional incorrect answers are counted. The outer sum covers all cell-construction inputs, since incorrect answers can be added even when the ground-truth answer was generated.

We now use these accumulated counts to choose cells sequentially. At step t, adding an unselected cell $\mathcal{B}_\ell$ gives the collection indexed by $\mathcal{I}_{t-1} \cup \{\ell\}$. We evaluate its inefficiency as

$$Q_\ell^{(t)} = \frac{F(\mathcal{I}_{t-1} \cup \{\ell\})}{T(\mathcal{I}_{t-1} \cup \{\ell\})}. \tag{22}$$

A smaller value indicates fewer incorrect vocabulary answers per recovered ground-truth answer across the entire collection after the addition. Let $\pi(t)$ denote the index of the cell chosen at step t. We select the cell with the smallest inefficiency and update the collection:

$$\pi(t) = \operatorname*{arg\,min}_{\ell \in \{1,\ldots,L\} \setminus \mathcal{I}_{t-1}} Q_\ell^{(t)}, \qquad \mathcal{I}_t = \mathcal{I}_{t-1} \cup \{\pi(t)\}. \tag{23}$$

Repeating this procedure for $t = 1, \ldots, L$ produces the complete ordering $\pi(1), \ldots, \pi(L)$. Figure 2 illustrates the cell-ordering procedure. Cell partitioning is also used in multi-dimensional CP (Tawachi & Laufer-Goldshtein, 2025). We discuss the differences in Appendix E.

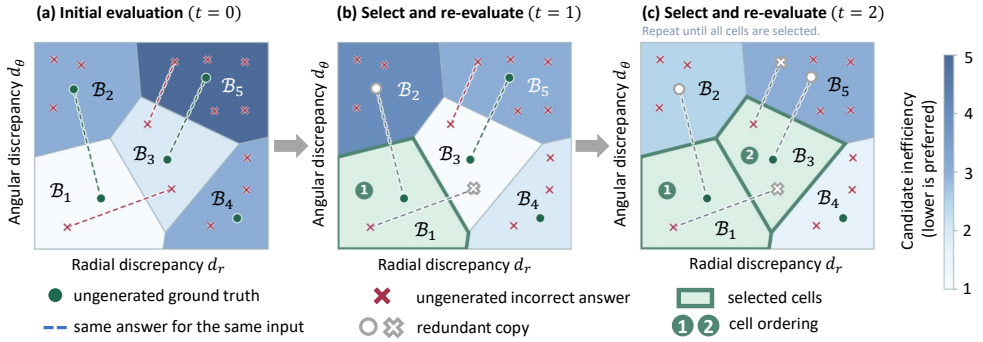

Figure 2: **Sequential cell ordering.** For a given input, an answer is compared with multiple distinct generated answers, so its discrepancy vectors may occupy several cells. Once any of these cells is selected, the answer is admitted; its remaining vectors become redundant because admitting the same answer again provides neither additional ground-truth recovery nor additional set-size cost. We therefore re-evaluate every remaining cell after each selection, using the accumulated true and false counts that would follow its addition and counting each input–answer pair only once. Panels (a–c) illustrate the first two selections, each choosing the cell with the lowest candidate inefficiency, defined as the accumulated false-to-true count ratio after adding that cell.

Cell scalarization. We use the position of each cell in this ordering to scalarize the discrepancy vectors it contains. A nonzero vector $\mathbf{z}$ in $\mathcal{B}_{\pi(t)}$ is reached after the first t cells have been included. Assigning zero to the origin, we define

$$\rho_{\text{cell}}(\mathbf{z}) = \begin{cases} 0, & \mathbf{z} = \mathbf{0}, \\ t, & \mathbf{z} \neq \mathbf{0},\ \mathbf{z} \in \mathcal{B}_{\pi(t)}, \quad t = 1, \ldots, L. \end{cases} \tag{24}$$

With ρ_{cell} fixed, we apply the same repeated-generation procedure to the held-out recalibration examples and compute $V_i = s^{\mathcal{Y}}_{\rho_{\text{cell}}}(X_i, Y_i)$ for $i = c+1, \ldots, n$. We obtain τ_{cell} using the same finite-sample corrected $(1-\alpha)$-quantile rule, with recalibration sample size $n - c$. The semantic prediction set for a test input is then $\mathcal{C}^{\mathcal{Y}\to\mathcal{W}}_{\rho_{\text{cell}}}(X_{n+1}; \tau_{\text{cell}})$, as defined in Eq. (14). Algorithm 2 presents the pseudocode for constructing the semantic prediction set using cell scalarization.

5 EXPERIMENT

Datasets. We evaluate on short-form question-answering: TriviaQA (Joshi et al., 2017), SciQ (Welbl et al., 2017), Natural Questions-Open (NQ-Open) (Lee et al., 2019), SimpleQuestions-Wiki (SQ-Wiki) (Diefenbach et al., 2017), and WebQuestions (WQ) (Berant et al., 2013).

Models. Gemma-3-4B-Instruct (Gemma Team, 2025) and LLaMA-3.2-3B-Instruct (Grattafiori et al., 2024) are used as inference LMs. For each input, we sample $M = 10$ outputs from inference LMs at a decoding temperature of 1.0. Qwen2.5-7B (Qwen et al., 2025) serves as the statement-generation model g, and all-mpnet-base-v2 (Song et al., 2020) as the encoder f.

Answer vocabulary. For each dataset, we reserve an independent development set to prune WordNet (Ciaramita & Johnson, 2003) separately for each inference LM. We remove categories containing few or no ground-truth answers missed during generation, as detailed in Appendix F, to improve semantic prediction set efficiency. Therefore, each dataset–inference-model combination has its own pruned vocabulary $\mathcal{W}$.

Evaluation protocol. After excluding development examples, we evaluate two versions of each dataset: the full version containing all remaining examples, and a recoverable version retaining only examples whose ground-truth answer is generated by the LM or contained in $\mathcal{W}$. We report $\delta_{\mathcal{W}}$ as the proportion of examples in the full version that are not retained in the recoverable version. Each version is randomly divided into calibration and test sets in a $2:1$ ratio. For cell scalarization, the calibration set is further divided equally into cell-construction and recalibration sets. We repeat the evaluation over 100 random splits, calibrating each method separately for each split.

Baselines. We compare against four sampling-based CP methods. ConU (Wang et al., 2024b) combines generation frequencies with semantic similarity. LofreeCP (Su et al., 2024) further incorporates normalized entropy into its nonconformity score. CLM (Quach et al., 2023) jointly calibrates when to stop sampling and which low-quality answers to reject. SConU (Wang et al., 2025) derives a minimum risk level from ground-truth answers missed during generation on calibration examples. Despite these differences, their sampling-based prediction sets remain limited to generated answers.

5.1 COVERAGE AND SET EFFICIENCY

Figure 3 compares empirical coverage across five datasets and two inference LMs at the target level $1 - \alpha = 0.9$. We report the hybrid prediction set with norm scalarization and the explicit semantic prediction sets with norm and cell scalarizations. On the full versions, sampling-based baselines fall below the target coverage, reflecting their dependence on generating a correct answer. Hybrid prediction sets achieve near-target coverage by including missed ground-truth answers through the latent region. The semantic sets also improve coverage over the baselines, but cannot recover ground-truth answers that are neither generated nor contained in $\mathcal{W}$. Their coverage is therefore bounded above by $1 - \delta_{\mathcal{W}}$. This ceiling is independent of whether norm or cell scalarization is used. On the recoverable versions, both norm and cell semantic prediction sets achieve near-target coverage, while sampling-based baselines remain limited by sampling recall. These results demonstrate that semantic sets can recover ungenerated ground-truth when they are available in the vocabulary.

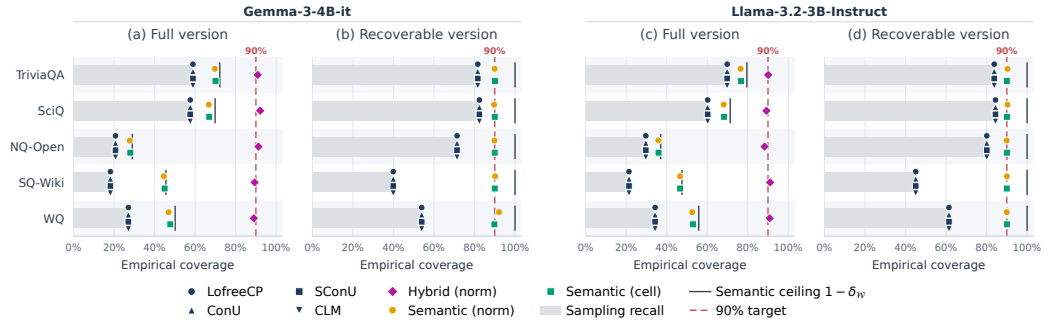

Figure 3: Coverage comparison at a 90% target with $M = 10$ generations. Hybrid prediction sets achieve near-target coverage on the full versions. Explicit semantic prediction sets improve coverage over sampling-based baselines on the full versions and achieve near-target coverage on the recoverable versions.

We compare semantic prediction set sizes under norm and cell scalarizations on the recoverable versions in Figure 4. Baseline sizes are omitted because these methods do not attain the target coverage. Cell scalarization yields lower mean sizes, supporting the effectiveness of the learned ordering. Set sizes vary from tens to thousands of answers, reflecting differences in how many vocabulary answers satisfy the calibrated threshold for each input. While large sets can limit direct usability, lowering the target coverage substantially reduces mean set size, as demonstrated in Table 6, allowing users to balance coverage and set size. Appendix H further reports the corresponding coverage and size results on both full and recoverable versions across different target coverage levels.

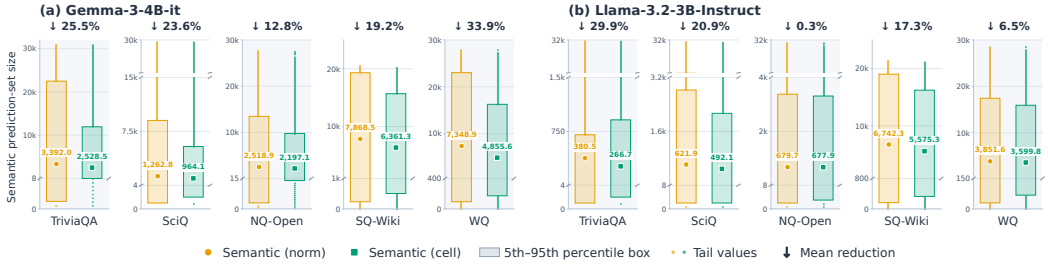

Figure 4: Semantic prediction set sizes on the recoverable versions at $1 - \alpha = 0.9$ and $M = 10$. Cell scalarization yields smaller mean sizes than norm scalarization across inference LMs and datasets.

5.2 Ablation Studies

The hybrid prediction set with norm scalarization maintains near-target coverage across sampling budgets in Figure 5 (a). As the budget M decreases, baseline coverage drops with sampling recall. Semantic prediction sets exhibit a smaller coverage decline, as they can recover ground-truth answers from $\mathcal{W}$ even when those answers are not generated. This provides greater robustness to limited sampling budget. Figure 5 (b) shows that mean semantic prediction set sizes under both norm and cell scalarizations generally decrease as M increases. This trend is consistent with additional generations recovering ground truth directly or providing more favorable latent comparisons, reducing the expansion needed after calibration. These results suggest that increasing the sampling budget can reduce semantic set size, at the cost of additional LM generations. Appendix G examines the contribution of angular and radial discrepancies. Appendix I discusses limitations and extensions concerning prediction set size, vocabulary dependence, and long-form question answering.

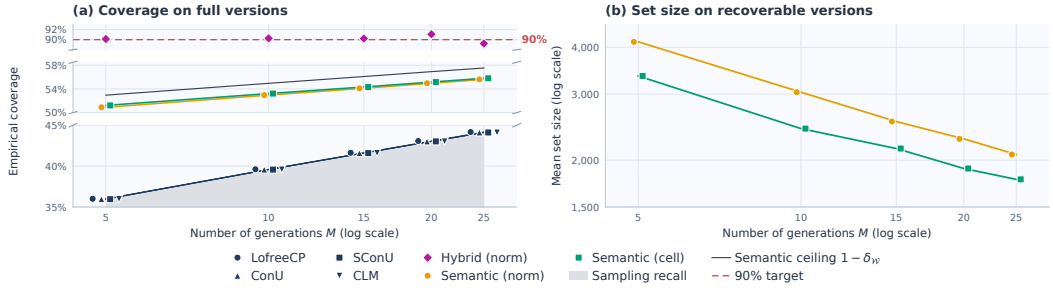

Figure 5: Effect of sampling budget M, weighted by test-set-size across two LMs and five datasets. (a) On full versions, hybrid coverage remains near the target, while vocabulary $\mathcal{W}$ makes semantic coverage more robust to limited sampling than baseline coverage. (b) On recoverable versions, mean semantic set sizes generally decrease with M under both scalarizations, with cell producing smaller sets than norm. Size comparisons also reflect changes in the recoverable versions, which are reconstructed at each M.

6 Conclusion

We proposed a hybrid conformal framework combining generation frequencies with encoder-based discrepancies. Under exchangeability, the hybrid prediction set provides marginal coverage without requiring the ground-truth answer to be generated. A finite vocabulary enables explicit recovery of missed answers, with vocabulary omissions limiting semantic coverage. Learned cell scalarization improves the efficiency of this extension by controlling the order in which additional answers are admitted. Together, these components establish a mechanism for recovery beyond sampling, while producing compact semantic prediction sets for direct use is an important direction for future work.

AI USE STATEMENT

We used generative AI tools to polish the manuscript's language and improve clarity. The authors reviewed all AI-assisted revisions and take responsibility for the final content, including its accuracy and originality.

REFERENCES

Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1533–1544, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1160.

Aaron Chatterji, Thomas Cunningham, David J Deming, Zoe Hitzig, Christopher Ong, Carl Yan Shan, and Kevin Wadman. How people use chatgpt. Technical report, National Bureau of Economic Research, 2025.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. Inside: Llms' internal states retain the power of hallucination detection, 2024. URL https://arxiv.org/abs/2402.03744.

John J Cherian, Isaac Gibbs, and Emmanuel J Candès. Large language model validity via enhanced conformal prediction methods. *Advances in Neural Information Processing Systems*, 37:114812–114842, 2024.

Massimiliano Ciaramita and Mark Johnson. Supersense tagging of unknown nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pp. 168–175, 2003. URL https://aclanthology.org/W03-1022/.

Dennis Diefenbach, Thomas Pellissier Tanon, Kamal Deep Singh, and Pierre Maret. Question answering benchmarks for wikidata. In *Proceedings of the ISWC 2017 Posters & Demonstrations and Industry Tracks co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 23rd - to - 25th, 2017.*, 2017. URL http://ceur-ws.org/Vol-1963/paper555.pdf.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. *arXiv preprint arXiv:2307.01379*, 2023.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, et al. Fact-checking the output of large language models via token-level uncertainty quantification. *arXiv preprint arXiv:2403.04696*, 2024.

Gemma Team. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Aaron Grattafiori et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *ArXiv*, abs/1705.03551, 2017. URL https://api.semanticscholar.org/CorpusID:26501419.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language models hallucinate, 2025. URL https://arxiv.org/abs/2509.04664.

Ramneet Kaur, Colin Samplawski, Adam D. Cobb, Anirban Roy, Brian Matejek, Manoj Acharya, Daniel Elenius, Alexander Michael Berenbeim, John A. Pavlik, Nathaniel D. Bastian, and Susmit Jha. Addressing uncertainty in LLMs to enhance reliability in generative AI. In *Neurips Safe Generative AI Workshop 2024*, 2024. URL https://openreview.net/forum?id=Z3DS4Pcxct.

Michal Klein, Louis Bethune, Eugene Ndiaye, and Marco Cuturi. Multivariate conformal prediction using optimal transport, 2025. URL https://arxiv.org/abs/2502.03609.

Vasily Kostumov, Bulat Nutfullin, Oleg Pilipenko, and Eugene Ilyushin. Uncertainty-aware evaluation for vision-language models. *arXiv preprint arXiv:2402.14418*, 2024.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

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.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6086–6096, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1612. URL https://aclanthology.org/P19-1612/.

Haitao Li, Junjie Chen, Jingli Yang, Qingyao Ai, Wei Jia, Youfeng Liu, Kai Lin, Yueyue Wu, Guozhi Yuan, Yiran Hu, et al. Legalagentbench: Evaluating llm agents in legal domain. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2322–2344, 2025.

Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. Uncertainty quantification and confidence calibration in large language models: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 6107–6117, 2025.

Junyu Luo, Weizhi Zhang, Ye Yuan, Yusheng Zhao, Junwei Yang, Yiyang Gu, Bohan Wu, Binqi Chen, Ziyue Qiao, Qingqing Long, Rongcheng Tu, Xiao Luo, Wei Ju, Zhiping Xiao, Yifan Wang, Meng Xiao, Chenwu Liu, Jingyang Yuan, Shichang Zhang, Yiqiao Jin, Fan Zhang, Xian Wu, Hanqing Zhao, Dacheng Tao, Philip S. Yu, and Ming Zhang. Large language model agent: A survey on methodology, applications and challenges, 2025. URL https://arxiv.org/abs/2503.21460.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*, 2023.

Katerina Margatina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu. Active learning principles for in-context learning with large language models. *arXiv preprint arXiv:2305.14264*, 2023.

Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. *arXiv preprint arXiv:2402.10978*, 2024.

John Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M Rush. Text embeddings reveal (almost) as much as text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12448–12460, 2023.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *Machine learning: ECML 2002: 13th European conference on machine learning Helsinki, Finland, August 19–23, 2002 proceedings 13*, pp. 345–356. Springer, 2002.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. Conformal language modeling. *arXiv preprint arXiv:2306.10193*, 2023.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Allen Z Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, et al. Robots that ask for help: Uncertainty alignment for large language model planners. *arXiv preprint arXiv:2307.01928*, 2023a.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J. Liu. Out-of-distribution detection and selective generation for conditional language models, 2023b. URL https://arxiv.org/abs/2209.15558.

Yaniv Romano, Matteo Sesia, and Emmanuel Candes. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, 33:3581–3591, 2020.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting, 2024. URL https://arxiv.org/abs/2310.11324.

Huanxin Sheng, Xinyi Liu, Hangfeng He, Jieyu Zhao, and Jian Kang. Analyzing uncertainty of llm-as-a-judge: Interval evaluations with conformal prediction, 2025. URL https://arxiv.org/abs/2509.18658.

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 Computing Surveys*, 2025.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *ArXiv*, abs/2004.09297, 2020. URL https://api.semanticscholar.org/CorpusID:215827489.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. LLM-check: Investigating detection of hallucinations in large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=LYx4w3CAgy.

Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. Api is enough: Conformal prediction for large language models without logit-access, 2024. URL https://arxiv.org/abs/2403.01216.

Yam Tawachi and Bracha Laufer-Goldshtein. Multi-dimensional conformal prediction. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=loDppyW7e2.

Rosario Uceda-Sosa, Karthikeyan Natesan Ramamurthy, Maria Chang, and Moninder Singh. Reasoning about concepts with llms: Inconsistencies abound, 2024. URL https://arxiv.org/abs/2405.20163.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

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

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*, 2024a.

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*, pp. 6886–6898, 2024b.

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.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. In Leon Derczynski, Wei Xu, Alan Ritter, and Tim Baldwin (eds.), *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pp. 94–106, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4413. URL https://aclanthology.org/W17-4413/.

Ziqi Yang, Xuhai Xu, Bingsheng Yao, Ethan Rogers, Shao Zhang, Stephen Intille, Nawar Shara, Guodong Gordon Gao, and Dakuo Wang. Talk2care: An llm-based voice assistant for communication between healthcare providers and older adults. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(2):1–35, 2024.

Fanlong Zeng, Wensheng Gan, Zezheng Huai, Lichao Sun, Hechang Chen, Yongheng Wang, Ning Liu, and Philip S. Yu. Large language models for robotics: A survey, 2025. URL https://arxiv.org/abs/2311.07226.

Collin Zhang, John X Morris, and Vitaly Shmatikov. Universal zero-shot embedding inversion. *arXiv preprint arXiv:2504.00147*, 2025.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. Enhancing uncertainty-based hallucination detection with stronger focus. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 915–932, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.58. URL https://aclanthology.org/2023.emnlp-main.58/.

A COVERAGE OF THE SEMANTIC PREDICTION SET

We analyze the semantic prediction set $\mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}$ using the nonconformity score $s_\rho^{\mathcal{Y}}$ and the threshold τ calibrated on $\mathcal{D}_{\text{cal}}$. Since the same generation procedure is independently applied to the recalibration examples and the test example, the corresponding scores are exchangeable. Therefore, the finite-sample conformal quantile satisfies

$$\Pr\big(s_\rho^{\mathcal{Y}}(X_{n+1}, Y_{n+1}) > \tau\big) \le \alpha.$$

Let $\mathcal{E}_{\text{score}}$ denote the event $\{s_\rho^{\mathcal{Y}}(X_{n+1}, Y_{n+1}) > \tau\}$, and let $\mathcal{E}_{\mathcal{W}}$ denote the event $\{Y_{n+1} \notin \widehat{\mathcal{Y}}(X_{n+1}) \text{ and } Y_{n+1} \notin \mathcal{W}\}$. Thereby, we can also denote $\delta_{\mathcal{W}} = \Pr(\mathcal{E}_{\mathcal{W}})$ and $\alpha \ge \Pr(\mathcal{E}_{\text{score}})$.

By construction, generated answers remain eligible for inclusion, while only ungenerated answers are restricted to $\mathcal{W}$. Consequently,

$$Y_{n+1} \notin \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1}; \tau) \quad \Longleftrightarrow \quad \mathcal{E}_{\text{score}} \text{ or } \mathcal{E}_{\mathcal{W}}.$$

Hence,

$$\begin{aligned}\Pr\big(Y_{n+1} \notin \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1}; \tau)\big) &= \Pr(\mathcal{E}_{\text{score}} \cup \mathcal{E}_{\mathcal{W}}) \\ &\le \Pr(\mathcal{E}_{\text{score}}) + \Pr(\mathcal{E}_{\mathcal{W}}) \\ &\le \alpha + \delta_{\mathcal{W}}.\end{aligned}$$

Taking complements gives

$$\Pr\big(Y_{n+1} \in \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1}; \tau)\big) \ge 1 - \alpha - \delta_{\mathcal{W}}.$$

The additive bound can be conservative because the two failure events may overlap. Indeed,

$$\Pr\big(Y_{n+1} \in \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1}; \tau)\big) = 1 - \Pr(\mathcal{E}_{\text{score}}) - \delta_{\mathcal{W}} + \Pr(\mathcal{E}_{\text{score}} \cap \mathcal{E}_{\mathcal{W}}).$$

Thus, the overlap contributes directly to the actual coverage.

In particular, if $\mathcal{E}_{\mathcal{W}} \subseteq \mathcal{E}_{\text{score}}$, then

$$\Pr\big(Y_{n+1} \in \mathcal{C}_\rho^{\mathcal{Y}\to\mathcal{W}}(X_{n+1}; \tau)\big) = 1 - \Pr(\mathcal{E}_{\text{score}}) \ge 1 - \alpha.$$

Since this containment also implies $\delta_{\mathcal{W}} \le \alpha$, the corresponding worst-case bound can equivalently be written as $1 - \max\{\alpha, \delta_{\mathcal{W}}\} = 1 - \alpha$.

In summary, the distribution-free guarantee is $1 - \alpha - \delta_{\mathcal{W}}$. When $\delta_{\mathcal{W}} = 0$, the explicit semantic prediction set retains the target marginal coverage $1 - \alpha$. When such a containment assumption $\mathcal{E}_{\mathcal{W}} \subseteq \mathcal{E}_{\text{score}}$ holds, $1 - \alpha$ coverage is still attainable with $\delta_{\mathcal{W}} \ne 0$.

B STATEMENT GENERATION PROCESS

In Eq. (4), we denote by g a statement-generation model that transforms a question-answer pair (X, Y) into a query-conditioned statement $g(X, Y)$. Here we provide additional details on how the statement is generated in practice.

B.1 DIRECT STATEMENT GENERATION

Given a question X and its corresponding answer Y, we prompt a language model to rewrite the question-answer pair as a concise declarative statement. Specifically, the model is instructed to preserve the semantic content of both the question and the answer while expressing them as a natural-language statement. The generated statement is then used as the semantic representation of the corresponding prediction.

For example, the generation process can be illustrated as follows:

$$(X, Y) = (\text{“What is the capital of France?”}, \text{“Paris”}),$$

which is provided to the language model using the following prompt:

> **Instruction:** Convert the following question and answer into a single concise declarative statement. The statement should naturally incorporate the answer as the response to the question, while preserving the original meaning.
> **Question:** What is the capital of France?
> **Answer:** Paris
> **Statement:**

The language model g may then generate:

> **Statement:** Paris is the capital of France.

Thus, in this example, $g(X, Y) =$ "Paris is the capital of France."

The same procedure is applied to both the ground-truth answer Y and the model predictions $\widehat{Y}^{(k)}$ for $k = 1, ..., K$. This query-conditioned formulation is important because the same answer can have different semantic interpretations depending on the question. By explicitly incorporating the question into the generated statement, the resulting representation captures the relationship between the answer and its corresponding query rather than evaluating the answer in isolation.

B.2 Template-based statement generation

In Section 4.2, for each sample $(X_i, Y_i) \in \mathcal{D}_{\text{cell}}$, we need to compute $g(X_i, y)$ for all $y \in \mathcal{W} \cup \widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}$ to conduct cell construction and ordering. Directly generating a statement for each candidate y would require tens of thousands of separate calls to the language model g, since the vocabulary $\mathcal{W}$ contains tens of thousands of words. To avoid this substantial computational overhead, we generate the statement template only once for each query X_i, with a `PLACEHOLDER` token in place of the answer. We then replace `PLACEHOLDER` with each candidate $y \in \mathcal{W} \cup \widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}$ to obtain the corresponding query-conditioned statement.

Specifically, for a given query, we first prompt the language model g using the following template:

> **Instruction:** Convert the following question and answer into a single concise declarative statement. Preserve the semantic meaning of the question and answer. Use `PLACEHOLDER` as a placeholder for the answer.
> **Question:** What is the capital of France?
> **Answer:** `PLACEHOLDER`
> **Statement:**

The model g therefore generates:

> **Statement:** `PLACEHOLDER` is the capital of France.

The resulting template is then instantiated for each candidate answer. For example, for $y =$ Paris, we obtain 'Paris is the capital of France,' while for another candidate $y =$ London, we obtain 'London is the capital of France.' Thus, the statement-generation model g can be efficiently evaluated for all candidates through a single template-generation step followed by simple string replacement:

$$g(X, y) = \text{Replace}\big(g(X, \texttt{PLACEHOLDER}), \texttt{PLACEHOLDER}, y\big).$$

This procedure substantially reduces the number of language-model calls while preserving the query-conditioned structure of the generated statements.

The template-based statement generation is also applied to the recalibration set $\mathcal{D}_{\text{recal}}$. Specifically, for each $(X_i, Y_i) \in \mathcal{D}_{\text{recal}}$, we generate a statement for every $y \in \widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}$ using the same template-based procedure.

C ALGORITHMS

Algorithm 1 summarizes the hybrid conformal prediction procedure. The prediction set is maintained in a hybrid form: generated answers satisfying the cumulative-frequency threshold are represented explicitly, whereas ungenerated answers are represented implicitly by the latent inclusion rule in Eq. (12). Consequently, constructing the prediction set for X_{n+1} does not require immediately computing the embeddings of its distinct generated answers. Given a particular answer $y \notin \widehat{\mathcal{Y}}(X_{n+1})$, its membership is evaluated on demand by computing $h(X_{n+1}, y)$ and $h(X_{n+1}, \widehat{Y}_{n+1}^{(k)})$ for $k = 1, \ldots, K_{n+1}$, and checking whether

$$\min_{1 \le k \le K_{n+1}} \rho\Big(\mathbf{d}^{(k)}(X_{n+1}, y)\Big) \le \tau - 1.$$

The embeddings of the distinct generated answers can be cached and reused when membership is queried for multiple answers.

Algorithm 1 Hybrid conformal prediction

Require: Calibration set $\mathcal{D}_{\mathrm{cal}} = \{(X_i, Y_i)\}_{i=1}^{n}$, test input X_{n+1}, number of generations M, encoder f, statement generator g, scalarization ρ, and miscoverage level α.
Ensure: Hybrid prediction set $\mathcal{C}_\rho^{\mathcal{Y}}(X_{n+1}; \tau)$.

for $i = 1, \ldots, n$ **do**
  Independently sample $A_i^{(1)}, \ldots, A_i^{(M)} \sim \Pr_{\mathrm{LM}}(\cdot \mid X_i)$.
  Deduplicate the outputs as $\widehat{\mathcal{Y}}(X_i) = \{\widehat{Y}_i^{(1)}, \ldots, \widehat{Y}_i^{(K_i)}\}$, with $p_i^{(1)} \ge \cdots \ge p_i^{(K_i)}$.
  for $k = 1, \ldots, K_i$ **do**
    Compute the cumulative frequency $P_i^{(k)} = \sum_{\ell=1}^{k} p_i^{(\ell)}$.
    Compute $\mathbf{d}^{(k)}(X_i, Y_i)$ using Eq. (5) and Eq. (6), where $h(X_i, y) = f(g(X_i, y))$.
  end for
  Construct the nested family $\mathcal{C}_\rho^{\mathcal{Y}}(X_i; q)$ for $q \ge 0$ according to Eq. (9).
  Compute the nonconformity score $V_i \leftarrow s_\rho^{\mathcal{Y}}(X_i, Y_i) = \inf\left\{q \ge 0 : Y_i \in \mathcal{C}_\rho^{\mathcal{Y}}(X_i; q)\right\}$.
end for
Obtain τ as the $\lceil (1-\alpha)(n+1) \rceil$-th order statistic of $\{V_i\}_{i=1}^{n}$.
Independently sample $A_{n+1}^{(1)}, \ldots, A_{n+1}^{(M)} \sim \Pr_{\mathrm{LM}}(\cdot \mid X_{n+1})$,
Deduplicate the outputs as $\widehat{\mathcal{Y}}(X_{n+1}) = \{\widehat{Y}_{n+1}^{(1)}, \ldots, \widehat{Y}_{n+1}^{(K_{n+1})}\}$, with $p_{n+1}^{(1)} \ge \cdots \ge p_{n+1}^{(K_{n+1})}$.
Compute $P_{n+1}^{(k)} = \sum_{\ell=1}^{k} p_{n+1}^{(\ell)}$ for $k = 1, \ldots, K_{n+1}$.
Construct hybrid prediction set $\mathcal{C}_\rho^{\mathcal{Y}}(X_{n+1}; \tau)$ using Eq. (12).
return $\mathcal{C}_\rho^{\mathcal{Y}}(X_{n+1}; \tau)$.

Algorithm 2 provides pseudocode for constructing the cell-ordered semantic prediction set. It first uses $\mathcal{D}_{\mathrm{cell}}$ to construct and order the cells, thereby defining ρ_{cell}. The learned scalarization is then recalibrated on the independent set $\mathcal{D}_{\mathrm{recal}}$ through the unrestricted nested family $\mathcal{C}_{\rho_{\mathrm{cell}}}^{\mathcal{Y}}$. Finally, for a test input, the algorithm restricts the addition of ungenerated answers to $\mathcal{W}$ and returns the explicit semantic prediction set $\mathcal{C}_{\rho_{\mathrm{cell}}}^{\mathcal{Y} \to \mathcal{W}}(X_{n+1}; \tau_{\mathrm{cell}})$.

Algorithm 2 Cell-ordered semantic conformal prediction

Require: Cell-construction set $\mathcal{D}_{\text{cell}} = \{(X_i, Y_i)\}_{i=1}^{c}$, recalibration set $\mathcal{D}_{\text{recal}} = \{(X_i, Y_i)\}_{i=c+1}^{n}$, test input X_{n+1}, finite vocabulary $\mathcal{W}$, number of generations M, encoder f, statement generator g, and miscoverage level α.

Ensure: Explicit semantic prediction set $\mathcal{C}^{\mathcal{Y}\to\mathcal{W}}_{\rho_{\text{cell}}}(X_{n+1}; \tau_{\text{cell}})$.

for $i = 1, \ldots, c$ **do**
  Sample, deduplicate, and order the outputs for X_i, obtaining $\widehat{\mathcal{Y}}(X_i) = \{\widehat{Y}_i^{(1)}, \ldots, \widehat{Y}_i^{(K_i)}\}$.
  if $Y_i \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)$ **then**
    Compute $\mathbf{d}^{(k)}(X_i, Y_i)$ for $k = 1, \ldots, K_i$ and retain these vectors as candidate centers.
  end if
end for
Remove duplicate candidate centers to obtain $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_L$.
Construct the Voronoi cells $\mathcal{B}_1, \ldots, \mathcal{B}_L$ using Eq. (18).
For every construction input X_i and $y \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)$, compute $\Gamma_i(y)$ using Eq. (19).
Initialize $\mathcal{I}_0 \leftarrow \varnothing$.
for $t = 1, \ldots, L$ **do**
  for $\ell \in \{1, \ldots, L\} \setminus \mathcal{I}_{t-1}$ **do**
    Compute $T(\mathcal{I}_{t-1} \cup \{\ell\})$ and $F(\mathcal{I}_{t-1} \cup \{\ell\})$ using Eq. (20) and Eq. (21), respectively.
    Compute candidate inefficiency $Q_\ell^{(t)} \leftarrow \dfrac{F(\mathcal{I}_{t-1} \cup \{\ell\})}{T(\mathcal{I}_{t-1} \cup \{\ell\})}$.
  end for
  Select $\pi(t) \leftarrow \arg\min_{\ell \notin \mathcal{I}_{t-1}} Q_\ell^{(t)}$.
  Update $\mathcal{I}_t \leftarrow \mathcal{I}_{t-1} \cup \{\pi(t)\}$.
end for
Define ρ_{cell} using Eq. (24).
for $i = c+1, \ldots, n$ **do**
  Apply the same sampling and deduplication procedure to X_i, obtaining $\widehat{\mathcal{Y}}(X_i)$, empirical frequencies $p_i^{(k)}$, and cumulative frequencies $P_i^{(k)}$.
  Compute the unrestricted score $V_i \leftarrow s^{\mathcal{Y}}_{\rho_{\text{cell}}}(X_i, Y_i)$ using Eq. (10).
end for
Set τ_{cell} to the $\lceil (1-\alpha)(n-c+1) \rceil$-th smallest value among $V_{c+1}, \ldots, V_n$.
Apply sampling and deduplication to X_{n+1}, to get $\widehat{\mathcal{Y}}(X_{n+1})$, $\{p_{n+1}^{(k)}\}_{k=1}^{K_{n+1}}$, and $\{P_{n+1}^{(k)}\}_{k=1}^{K_{n+1}}$.
Construct $\mathcal{C}^{\mathcal{Y}\to\mathcal{W}}_{\rho_{\text{cell}}}(X_{n+1}; \tau_{\text{cell}})$ using Eq. (14).
return $\mathcal{C}^{\mathcal{Y}\to\mathcal{W}}_{\rho_{\text{cell}}}(X_{n+1}; \tau_{\text{cell}})$.

D COMPUTATION COST

D.1 COST OF HYBRID PREDICTION SET

Conformal calibration. For each of the n calibration examples in $\mathcal{D}_{\text{cal}}$, repeated generation requires M LM generations, for a total of nM.

The statement-generation model and encoder are applied only to the distinct generated answers and the ground-truth answer. Thus, the number of statement-generation and encoding operations is

$$\sum_{i=1}^{n} \left| \widehat{\mathcal{Y}}(X_i) \cup \{Y_i\} \right| \leq n(M+1), \tag{25}$$

so the workload for statement generation and encoding is bounded by $O(n(M+1))$.

For each example, we compute K_i discrepancy vectors between Y_i and the distinct generated answers and evaluate the hybrid score. The total number of discrepancy-vector computations is

$$\sum_{i=1}^{n} K_i \leq nM. \tag{26}$$

Hence, the computation cost for obtaining discrepancy vectors is bounded by $O(nM)$.

Test-time hybrid prediction set. For a test input X_{n+1}, the LM is sampled M times and the outputs are deduplicated into K_{n+1} distinct answers. This requires M LM generations.

The statement-generation model and encoder are applied once to each distinct generated answer, giving K_{n+1} statement-generation and encoding operations, which is bounded by $O(M)$ cost. If membership of a specified answer y in the latent region is queried, its statement and embedding are computed once, followed by K_{n+1} discrepancy-vector computations against the generated answers, which is bounded by $O(M)$ as well. If $\tau \leq 1$, the latent region is empty, and these additional embedding and discrepancy computations can be skipped.

D.2 COST OF SEMANTIC PREDICTION SET CONSTRUCTION

Cell construction and ordering. The c cell-construction examples require cM LM generations. For each input X_i, we compute embeddings for the distinct answers in $\mathcal{W} \cup \widehat{\mathcal{Y}}(X_i)$. Under per-answer statement generation, the total number of evaluations of each of g and f is

$$\sum_{i=1}^{c} \left| \mathcal{W} \cup \widehat{\mathcal{Y}}(X_i) \right| = O(c(|\mathcal{W}| + M)) \,. \tag{27}$$

Each embedding is computed once and reused across comparisons. Ungenerated ground-truth answers outside $\mathcal{W}$ require no embedding at this stage because they contribute neither cell centers nor recoverable true answers.

For each input, every ungenerated vocabulary answer is compared with all K_i distinct generated answers. The total number of discrepancy-vector computations is therefore

$$\sum_{i=1}^{c} K_i \left| \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i) \right| = O(cM|\mathcal{W}|). \tag{28}$$

These comparisons already include the missed ground-truth answers in $\mathcal{W}$, so their discrepancy vectors are reused as candidate centers without additional computation. All comparisons use cached embeddings and require no further evaluations of g or f.

After retaining the distinct centers, we assign each discrepancy vector to its nearest center to obtain $\Gamma_i(y)$. A direct search over the L centers requires

$$O(L \sum_{i=1}^{c} K_i |\mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)|) = O(cM|\mathcal{W}|L) \tag{29}$$

two-dimensional distance evaluations. The associations $\Gamma_i(y)$ are then cached throughout ordering.

Sequential ordering requires repeated evaluation of candidate inefficiency, but no new embeddings or discrepancy vectors. With incremental count updates, we maintain the accumulated true and false counts and, for each unselected cell, the numbers of additional answers it would admit. When an input–answer pair is first admitted, its contribution is removed from these additional counts for every remaining cell associated with it. Each pair triggers this update only once, so the total update work is bounded by

$$\sum_{i=1}^{c} \sum_{y \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)} |\Gamma_i(y)| \leq \sum_{i=1}^{c} K_i \left| \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i) \right|. \tag{30}$$

Scanning the remaining cells at each of the L selection steps requires $O(L^2)$ evaluations of candidate inefficiency. Thus, with cached associations and incremental updates, the ordering stage requires $O(L^2 + cM|\mathcal{W}|)$ time.

Conformal recalibration. The recalibration stage applies the same per-example computation as the calibration stage for constructing hybrid prediction sets: repeated LM generation, statement generation and encoding of the distinct generated answers and Y_i, followed by the computation of the K_i discrepancy vectors and the hybrid score. The only difference is that recalibration uses the independent set $\mathcal{D}_{\text{recal}} = \{(X_i, Y_i)\}_{i=c+1}^{n}$, which contains $n - c$ examples. Consequently, the recalibration bounds are obtained by replacing n with $n - c$: under $K_i \leq M$, the procedure requires $O((n - c)(M + 1))$ statement-generation and encoding operations and $O((n - c)M)$ discrepancy-vector computations.

Cell scalarization additionally assigns each nonzero discrepancy vector to its nearest cell center. Direct comparison with all L centers requires $O\big(L \sum_{i=c+1}^{n} K_i\big) = O((n - c)ML)$ distance evaluations in the two-dimensional discrepancy space. The assigned cell's rank then determines the scalarized value. These assignments require no additional evaluations of g or f.

Test-time semantic prediction set. For a test input X_{n+1}, the LM is sampled M times, yielding K_{n+1} distinct generated answers after deduplication. The statement-generation model g and encoder f are each applied once to every unique answer in $\mathcal{W} \cup \widehat{\mathcal{Y}}(X_{n+1})$.

Each answer in $\mathcal{W} \setminus \widehat{\mathcal{Y}}(X_{n+1})$ is compared with all K_{n+1} distinct generated answers, requiring $O(K_{n+1}|\mathcal{W} \setminus \widehat{\mathcal{Y}}(X_{n+1})|)$ discrepancy-vector computations. For cell scalarization, assigning these vectors to cells by direct nearest-center search additionally requires $O(LK_{n+1}|\mathcal{W} \setminus \widehat{\mathcal{Y}}(X_{n+1})|)$ distance evaluations. Once assigned, the cell ranks are used to check the calibrated inclusion condition, without further statement generation or encoding.

D.3 Practical computation cost

For $M = 10$ and $1 - \alpha = 0.9$, we consider one split of the recoverable versions, comprising 109,537 examples across both inference LMs and all five datasets. Table 1 summarizes a computational budget for four NVIDIA RTX A6000 GPUs and eight CPU workers. We use template-based statement generation. Both scalarizations use the same test-time LM generation and encoding workload; cell scalarization additionally requires cell construction and ordering and nearest-center assignments.

Table 1: Practical computation cost across the ten dataset–inference-model combinations. Each stage includes repeated LM generation and deduplication, followed by statement generation and encoding for the indicated answer set. Test-time costs cover the entire test set.

Stage	**Examples**	**Computations**	norm scalarization (h)	cell scalarization (h)
D.1 Cost of hybrid prediction set				
Conformal calibration	73,023	$\widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}$; discrepancy-vector computation; hybrid nonconformity scores; finite-sample corrected $(1-\alpha)$-quantile.	$\approx$ 8.8	–
Test-time hybrid prediction set	36,514	$\widehat{\mathcal{Y}}(X_{n+1})$; additional statement generation, encoding and discrepancy-vector computation when membership of a specified answer y is queried and $\tau > 1$.	$\approx$ 3.8	–
D.2 Cost of semantic prediction set construction				
Cell construction and ordering	36,511	$\mathcal{W} \cup \widehat{\mathcal{Y}}(X_i)$; discrepancy-vector computation; distinct centers from missed ground-truth answers in $\mathcal{W}$; nearest-center assignments to obtain $\Gamma_i(y)$; sequential ordering by $Q_\ell^{(t)}$ with incremental count updates.	–	$\approx$21.5
Conformal recalibration	36,512	$\widehat{\mathcal{Y}}(X_i) \cup \{Y_i\}$; discrepancy-vector computation; nearest-center assignments and cell ranks; hybrid nonconformity scores; finite-sample corrected $(1-\alpha)$-quantile.	–	$\approx$ 4.2
Test-time semantic prediction set	36,514	$\mathcal{W} \cup \widehat{\mathcal{Y}}(X_{n+1})$; discrepancy vectors for ungenerated vocabulary answers; norm scalarization or nearest-center assignments and cell ranks; calibrated inclusion condition.	$\approx$20.6	$\approx$21.0
Total: hybrid prediction set			$\approx$12.6	–
Total: semantic prediction set			$\approx$29.4	$\approx$46.7

E Comparison with multi-dimensional conformal prediction

Different prediction problems. Tawachi & Laufer-Goldshtein (2025) consider classification over a known finite label space. Their objective is to obtain smaller prediction sets by combining multiple nonconformity scores and selecting regions that contain relatively few incorrect labels. Our work addresses open-ended LM generation, where repeated generation may fail to produce the ground-truth answer. We therefore first construct a hybrid nonconformity score that supplements empirical generation frequencies with encoder-based discrepancies. Cell scalarization subsequently improves the efficiency of converting the latent region into an explicit semantic prediction set. Thus, the cell ordering operates within a framework that distinguishes generated answers from additional answers recovered through a finite vocabulary.

Different representations and cell construction. To describe the prior method using our notation, let $\mathbf{s}(X, y)$ denote its multidimensional nonconformity score for input X and label y. The method constructs Voronoi cells around the distinct vectors $\mathbf{s}(X_i, Y_i)$ from $\mathcal{D}_{\text{cell}}$. Each input–label pair produces one vector and is therefore assigned to one cell.

In our setting, comparing an answer with the K_i distinct generated answers produces $\mathbf{d}^{(1)}(X_i, y), \ldots, \mathbf{d}^{(K_i)}(X_i, y)$. The angular and radial discrepancies form the coordinates of each

vector; different generated answers produce different vectors for the same input–answer pair. Consequently, these vectors may occupy several cells, whose indices are recorded by $\Gamma_i(y)$ in Eq. (19). We also restrict candidate centers to discrepancy vectors of ground-truth answers satisfying $Y_i \in \mathcal{W} \setminus \widehat{\mathcal{Y}}(X_i)$. This restriction concentrates the partition on answers that can be newly recovered through vocabulary conversion: generated answers are already handled by frequency-based scoring, while ungenerated answers outside $\mathcal{W}$ cannot appear in the explicit semantic prediction set.

Different cell-ordering criteria. In the prior method, let T_ℓ and F_ℓ denote the numbers of true-label and false-label vectors in $\mathcal{B}_\ell$, respectively. Cells are ranked once by

$$Q_\ell = \frac{T_\ell + F_\ell}{T_\ell} = 1 + \frac{F_\ell}{T_\ell}. \tag{31}$$

Because each input–label pair occupies one cell, its contribution is counted once when counts are summed over a collection of cells.

This additivity does not generally hold in our setting. An answer is admitted as soon as any cell indexed by $\Gamma_i(y)$ is selected. For example, if $\Gamma_i(y) = \{1, 2\}$, selecting $\mathcal{B}_1$ admits y; subsequently selecting $\mathcal{B}_2$ cannot admit another copy of y for the same input. Counting that occurrence in both cells would therefore overestimate either ground-truth recovery or the number of incorrect answers, depending on whether $y = Y_i$. The Voronoi cells remain disjoint, but the collections of answers they admit can overlap.

We account for this overlap through the accumulated counts $T(\mathcal{I}_t)$ and $F(\mathcal{I}_t)$ defined in Eq. (20) and Eq. (21). At step t, the candidate inefficiency of adding $\mathcal{B}_\ell$ is

$$Q_\ell^{(t)} = \frac{F(\mathcal{I}_{t-1} \cup \{\ell\})}{T(\mathcal{I}_{t-1} \cup \{\ell\})}, \qquad \ell \notin \mathcal{I}_{t-1}. \tag{32}$$

This criterion evaluates the entire proposed collection, counting each input–answer occurrence only once. After a cell is selected, the remaining candidates are re-evaluated because some of the answers they admit may already have been included. The distinction from Eq. (31) is therefore the use of accumulated counts and their dependence on previous selections; omitting the additive constant of one does not itself change an ordering.

Calibration and vocabulary conversion. Both methods use separate data for cell construction and recalibration. In the prior setting, prediction sets are formed directly over the complete finite label space. Our recalibration instead uses the unrestricted hybrid score $s^{\mathcal{Y}}_{\rho_{\text{cell}}}$, including for ground-truth answers outside $\mathcal{W}$. The vocabulary restriction is applied only when constructing the explicit semantic prediction set in Eq. (14). Accordingly, generated answers remain eligible regardless of vocabulary membership, while additional answers must belong to $\mathcal{W}$. This separation preserves unrestricted conformal calibration and identifies vocabulary omission as a distinct source of coverage loss, quantified by $\delta_{\mathcal{W}}$.

F Vocabulary selection and pruning

The answer vocabulary should contain ground-truth answers missed during generation while remaining small enough for explicit enumeration. We therefore select complete WordNet categories using an independent development set $\mathcal{D}_{\text{dev}}$, separately for each dataset–inference-model combination. Development examples are excluded from cell construction, conformal calibration, and testing.

Candidate categories. Our initial WordNet vocabulary contains 146,740 normalized lexical entries, without filtering by word frequency or phrase length. We construct candidate categories from WordNet's native supersense categories (Ciaramita & Johnson, 2003) and its noun hierarchy. For the latter, each category contains the lexical entries associated with a noun synset and all its descendants, using root synsets whose `min_depth` is 3, 5, or 7. Retaining categories with at least 100 distinct entries gives 399 candidates.

Category membership is fixed before examining task-specific ground-truth answers. Selecting a category retains all its entries, including those that do not occur as development answers. We neither remove individual entries from a selected category nor insert ground-truth answers individually. Each candidate vocabulary $\mathcal{W}$ is the deduplicated union of its selected categories.

Selection criterion. For each development input, we use the first $M = 10$ LM generations to obtain $\widehat{\mathcal{Y}}(X_i)$. Vocabulary selection focuses on answers absent from this generated-answer set, since these are the answers for which the vocabulary provides an additional recovery opportunity. We define the recovery count and recovery efficiency as

$$\mathrm{Rec}(\mathcal{W}) = \sum_{(X_i,Y_i)\in\mathcal{D}_{\mathrm{dev}}} \mathbf{1}\Big\{Y_i \notin \widehat{\mathcal{Y}}(X_i),\ Y_i \in \mathcal{W}\Big\}, \tag{33}$$

$$\mathrm{Eff}(\mathcal{W}) = \frac{\mathrm{Rec}(\mathcal{W})}{|\mathcal{W}|}. \tag{34}$$

The recovery rate divides $\mathrm{Rec}(\mathcal{W})$ by the total number of development examples whose ground-truth answers were not generated, including answers absent from the initial WordNet vocabulary.

Each development example contributes at most once to the recovery count, even when its answer belongs to several selected categories. The same answer appearing in different examples contributes separately, whereas each lexical entry is counted only once in $|\mathcal{W}|$. These quantities measure whether missed answers are available in the vocabulary; their inclusion in a semantic prediction set additionally depends on the calibrated nonconformity score.

Initial category selection. We search for category combinations that recover as many missed development answers as possible subject to $|\mathcal{W}| < 40{,}000$. Starting from multiple initial combinations, we prioritize additions with a larger ratio of newly recovered examples to newly introduced lexical entries, accounting for overlap between categories. We then consider whole-category replacements and remove redundant categories. Among feasible candidates recovering at least 95% of the largest recovery count found during the search, we select the one with the highest recovery efficiency $\mathrm{Eff}(\mathcal{W})$.

Joint refinement. Starting from the initially selected vocabulary, we use local integer programming to jointly modify multiple categories, accepting only changes that reduce vocabulary size and increase recovery. A proposal is eligible only if it reduces vocabulary size and increases the recovery count relative to the current combination. Among eligible proposals, we retain those achieving at least 80% of the largest recovery gain found in that round and select the one with the highest recovery efficiency. Otherwise, the current vocabulary is retained. This procedure searches over complete categories and does not establish global optimality.

Validation and reuse. We verify that the exported vocabulary equals the deduplicated union of the selected categories and recompute its recovery count directly from the development examples. The vocabulary is then fixed across all random splits and shared by norm and cell scalarizations. The sampling-budget experiments reuse the vocabulary selected at $M = 10$ for all other budgets, without repeating category selection.

Table 2 shows that pruning substantially reduces vocabulary size while largely preserving the semantic coverage ceiling. The reported vocabularies contain approximately 79.4–86.1% fewer entries, while the ceiling decreases by only 2.35–4.73 percentage points. This limits the pool of additional answers, helping control semantic prediction set size while retaining most recoverable ground-truth answers.

G Contribution of discrepancy coordinates

We examine the contributions of angular and radial discrepancies under both norm and cell scalarizations. For norm scalarization, we compare the joint score $\sqrt{d_\theta^2 + d_r^2}$ with $|d_\theta|$ and $|d_r|$ separately. For cell scalarization, we compare the two-dimensional partition with one-dimensional Voronoi partitions constructed using either coordinate alone. Each variant is recalibrated separately and evaluated on the recoverable versions at a target coverage of $1 - \alpha = 0.9$, using the same vocabulary, sampling budget, and data splits.

Norm scalarization. Table 3 shows that using angular discrepancy alone closely matches the joint norm. Using radial discrepancy alone, however, produces much larger prediction sets. These results

Table 2: Vocabulary size and empirical semantic coverage ceiling before and after pruning, with $M = 10$ generations. Coverage ceilings are evaluated on the full versions.

		Original vocabulary		Pruned vocabulary	
Inference model	Dataset	$\lvert\mathcal{W}\rvert$	$1 - \delta_{\mathcal{W}}$ (%)	$\lvert\mathcal{W}\rvert$	$1 - \delta_{\mathcal{W}}$ (%)
Gemma-3-4B-it	TriviaQA	146,740	76.93	30,161	72.20
	SciQ	146,740	73.13	28,931	69.87
	Natural Questions-Open	146,740	31.62	27,545	28.99
	SimpleQuestions-Wiki	146,740	48.65	20,457	45.59
	WebQuestions	146,740	53.85	27,778	50.15
Llama-3.2-3B-Instruct	TriviaQA	146,740	83.22	30,193	79.58
	SciQ	146,740	74.56	29,552	71.36
	Natural Questions-Open	146,740	39.40	29,423	37.05
	SimpleQuestions-Wiki	146,740	50.44	21,149	47.53
	WebQuestions	146,740	58.91	28,296	55.80

Table 3: Effect of discrepancy coordinates on norm scalarization. Coverage is reported in percentages; size is the mean number of answers in the semantic prediction set on the recoverable versions. **Bold** identifies the smallest mean size in each row before rounding.

	Angular and radial		Angular only		Radial only	
Dataset	Coverage (%)	Size	Coverage (%)	Size	Coverage (%)	Size
Gemma-3-4B-it						
TriviaQA	89.93	3,392	89.52	**3,268**	89.94	6,317
SciQ	89.68	**1,263**	89.23	1,266	90.18	5,872
Natural Questions-Open	89.87	**2,519**	90.36	2,627	89.96	9,045
SimpleQuestions-Wiki	90.19	7,869	89.92	**7,707**	89.39	11,200
WebQuestions	92.12	**7,349**	92.28	7,591	89.39	14,547
Llama-3.2-3B-Instruct						
TriviaQA	90.34	**381**	90.44	388	89.97	2,080
SciQ	90.30	622	90.38	**595**	89.82	5,019
Natural Questions-Open	89.91	680	89.23	**593**	89.36	6,961
SimpleQuestions-Wiki	90.02	**6,742**	90.55	6,837	90.21	11,614
WebQuestions	89.95	3,852	89.91	**3,707**	90.00	14,669

indicate that angular discrepancy provides most of the discrimination achieved by norm scalarization in these experiments. Adding the radial term yields little additional improvement after recalibration. This is consistent with the radial term having limited influence on which answers satisfy the calibrated norm threshold.

Cell scalarization. Table 4 shows a different pattern: using either coordinate alone produces larger mean prediction sets than using both, with radial-only cells generally yielding the largest sets. The two-dimensional partition can distinguish discrepancy vectors with similar angular coordinates but different radial coordinates, allowing the learned ordering to prioritize regions containing recoverable ground-truth answers. Removing the radial coordinate eliminates this distinction and can place more incorrect answers in the same cells as ground-truth answers. The observed size increases support the usefulness of radial information for learning cell boundaries, even though its additional benefit under the fixed norm is small.

H Performance across target coverage levels

We fix the sampling budget at $M = 10$ and vary the target coverage $1 - \alpha \in \{0.6, 0.7, 0.8, 0.9\}$. We evaluate both the full and recoverable versions, aggregating results across five datasets and two inference LMs using the number of test examples in each split as weights. All methods use 100 random splits.

Table 4: Effect of discrepancy coordinates on cell scalarization. Coverage is reported in percentages; size is the mean number of answers in the semantic prediction set on the recoverable versions. **Bold** identifies the smallest mean size in each row before rounding.

	Angular and radial		Angular only		Radial only	
Dataset	Coverage (%)	Size	Coverage (%)	Size	Coverage (%)	Size
Gemma-3-4B-it						
TriviaQA	90.10	**2,529**	89.61	4,130	90.66	6,974
SciQ	89.99	**964**	90.14	2,093	89.99	7,126
Natural Questions-Open	89.97	**2,197**	90.56	3,201	89.47	10,152
SimpleQuestions-Wiki	89.99	**6,361**	90.29	8,431	88.02	11,166
WebQuestions	89.82	**4,856**	89.91	6,556	90.01	15,631
Llama-3.2-3B-Instruct						
TriviaQA	89.98	**267**	89.90	400	89.74	1,796
SciQ	90.05	**492**	90.59	840	89.67	5,425
Natural Questions-Open	90.04	**678**	89.04	703	90.11	8,372
SimpleQuestions-Wiki	89.99	**5,575**	90.32	7,261	89.71	11,107
WebQuestions	90.04	**3,600**	89.63	4,115	87.96	14,269

Coverage. Table 5 compares empirical coverage across target levels on both the full and recoverable versions. On the full versions, baseline coverage initially increases but soon plateaus: requesting higher coverage cannot recover answers absent from the generated outputs. Semantic prediction sets continue to improve coverage by admitting missed ground-truth answers from $\mathcal{W}$. However, with the sampling budget and vocabulary fixed, increasing the target does not change which answers are recoverable. Consequently, semantic coverage remains limited by $1 - \delta_{\mathcal{W}}$ even as the calibrated threshold increases.

On the recoverable versions, vocabulary omissions are absent, and both norm and cell scalarizations closely track all four target levels. Sampling-based baselines approach the lowest target but increasingly fall short at higher targets, despite the ground-truth answers being available in the vocabulary. This difference highlights the role of semantic inclusion: access to $\mathcal{W}$ allows higher target levels to translate into additional ground-truth recovery beyond sampling recall.

Table 5: Empirical coverage (%) across target levels with $M = 10$. Norm and Cell refer to semantic prediction sets. Results are weighted by test-set size across five datasets and two inference LMs.

$1 - \alpha$	LofreeCP	ConU	SConU	CLM	Norm	Cell
Full versions						
0.6	38.26	37.76	37.85	38.27	44.34	44.51
0.7	39.61	39.58	39.57	39.58	47.96	48.21
0.8	39.61	39.58	39.57	39.58	50.59	50.84
0.9	39.61	39.58	39.57	39.58	52.96	53.22
Recoverable versions						
0.6	56.34	56.72	58.10	57.16	60.08	60.02
0.7	63.48	63.84	65.32	63.89	70.16	70.00
0.8	69.84	70.18	70.99	70.01	80.21	80.02
0.9	72.09	72.02	72.03	71.99	90.12	90.01

Prediction set size. Table 6 compares semantic prediction set sizes at target levels from 60% to 90%. On the full versions, both scalarizations produce substantially larger sets as the target increases, while cell ordering provides only modest size reductions. Cell ordering prioritizes recovery of missed ground-truth answers within $\mathcal{W}$, whereas recalibration on the full versions also includes examples whose missed ground-truth answers lie outside $\mathcal{W}$. This mismatch can weaken the size advantage of the learned ordering, while vocabulary omissions continue to limit semantic coverage.

On the recoverable versions, both scalarizations attain comparable coverage, making their size differences informative about efficiency. Cell ordering provides increasingly larger reductions as the target rises, from 1.91% at 60% coverage to 20.02% at 90%. This pattern suggests that prioritizing

cells with fewer incorrect answers becomes more beneficial when higher coverage requires recovering more ground-truth answers beyond the generated outputs.

Table 6: Mean semantic prediction set size across target levels with $M = 10$, weighted by test-set size. Reduction is relative to norm scalarization; a negative value indicates an increase. The smaller mean size in each row is shown in **bold**.

$1 - \alpha$	Norm	Cell	Reduction (%)
Full versions			
0.6	973.93	**963.68**	1.05
0.7	2,161.65	**2,130.36**	1.45
0.8	4,785.16	**4,731.77**	1.12
0.9	10,510.06	**10,376.87**	1.27
Recoverable versions			
0.6	71.30	**69.94**	1.91
0.7	282.24	**244.96**	13.21
0.8	790.49	**663.03**	16.12
0.9	3,035.48	**2,427.77**	20.02

I Limitations and future directions

I.1 Prediction set size

Although cell scalarization improves efficiency, semantic prediction sets can still contain thousands of answers, limiting their usefulness for direct presentation to users. These large sets reflect the limited selectivity of the current discrepancy representation: recovering missed ground-truth answers can also require admitting many incorrect vocabulary answers. Table 6 further shows that this cost increases substantially at higher target coverage. Improving set efficiency therefore remains an important practical challenge.

One possible extension is to combine complementary representations from J encoders, yielding a $2J$-dimensional discrepancy space. The additional coordinates may help distinguish ground-truth answers from incorrect candidates that overlap in the current representation. Alternatively, query–answer relevance scores learned on independent training data could provide a more discriminative ordering before conformal calibration.

However, higher dimensionality also makes it harder to learn a reliable partition from a finite cell-construction set $\mathcal{D}_{\text{cell}}$. Estimated candidate inefficiencies may become sensitive to individual examples, causing the ordering to favor patterns that do not generalize to unseen inputs. Independent recalibration does not ensure that an overfitted ordering produces smaller semantic prediction sets. Future work could therefore combine complementary representations with regularized cell construction, for example by limiting partition complexity or merging cells supported by few construction examples. Any efficiency gains should be evaluated at comparable coverage and balanced against additional training and computational costs.

I.2 Vocabulary dependence

Explicit semantic prediction sets cannot recover answers that are neither generated nor contained in $\mathcal{W}$, limiting their coverage to $1 - \delta_{\mathcal{W}}$. Vocabulary selection therefore affects both answer recoverability and computational cost. A restricted vocabulary may omit rare entities or domain-specific answers, whereas enlarging it increases enumeration costs and may admit more incorrect candidates. Our pruning procedure balances these competing objectives, but cannot recover answers absent from the original vocabulary.

One possible extension is to construct query-dependent candidate vocabularies through retrieval from broader knowledge sources. By concentrating on answers relevant to each input, retrieval could improve recoverability without requiring exhaustive enumeration of a large global vocabulary. However, retrieval introduces its own omissions: ground-truth answers absent from both the

generated outputs and retrieved candidates remain unrecoverable. Evaluating such an extension would therefore require considering candidate recall alongside prediction set size and retrieval cost.

I.3 LONG-FORM QUESTION ANSWERING.

Our evaluation and explicit vocabulary construction focus on short-form QA. For longer responses, variation in wording makes exact-response frequencies less informative, while a finite vocabulary cannot adequately represent complete answers. One possible extension is embedding inversion (Morris et al., 2023; Zhang et al., 2025), which could generate candidate responses from the latent prediction region without a fixed dictionary. However, sampled embeddings may not correspond to valid responses, and approximate inversion can alter their meaning. Even when decoded candidates satisfy the latent inclusion rule, a finite collection need not contain the ground-truth answer; the original coverage guarantee therefore does not automatically transfer to the enumerated set.

A complementary direction is to combine our construction with conformal factuality and claim-filtering methods (Mohri & Hashimoto, 2024; Cherian et al., 2024). Responses could be decomposed into atomic claims, with retrieval or latent-space generation supplying additional candidates and calibrated filtering controlling the factuality of retained content. This would shift the objective from including a complete ground-truth answer to controlling factual errors in the response. Establishing such a guarantee requires response-level calibration that accounts for multiple dependent claims, while retaining sufficient informative content. We leave these extensions to future work.