Conformal Language Modeling

Haeyoung Lee

February 26, 2025

Seoul National University

Outline

Introduction

- LMs generate text based on probabilistic distributions. While effective, they can still produce incorrect or unreliable outputs.
- Quantifying uncertainty in LM responses remains a major challenge.
- Conformal Prediction is a model-agnostic method that ensures predictions contain correct responses with high probability.
- However, direct application to LMs is difficult due to their vast, unbounded output space.
- Unlike traditional models, LMs rely on approximate sampling rather than exhaustive enumeration.

Our Approach: Conformal Prediction for LMs

- We propose a method that calibrates a stopping rule for sampling LMs until confidence is met.
- A rejection mechanism filters out low-quality or redundant responses while maintaining theoretical guarantees.
- This ensures reliable and efficient prediction sets without requiring exhaustive search.

Challenges in Applying CP to LMs

Three major challenges:

- **Infinite Output Space** $\rightarrow$ Impossible to enumerate all possible text responses.
- Some outputs are **redundant or incorrect**.
- Need a **rejection rule** $\rightarrow$ Remove low-quality responses while maintaining coverage guarantees.

Conformal Language Modeling (CLM)

Solution: Sampling-Based Conformal Prediction

Conformal Language Modeling (CLM) Approach:

- **Sampling**: Sample responses from LLM.
- **Acceptance/Rejection**: Accept/reject based on confidence diversity.
- **Stopping Rule**: Stop sampling once certainty threshold is met.

Notation

- x : Input prompt.
- $p_\theta(y \mid x)$: Conditional output distribution defined by the language model.
- C_λ : Prediction set.
- $Q(x, y_k)$: Sample quality estimator.
- $S(y_k, y_j)$: Text similarity function.
- $\mathcal{F}$: Set-based confidence function.
- λ : Threshold configuration.
- λ_1 : Similarity threshold for filtering redundant samples.
- λ_2 : Quality threshold for rejecting low-quality samples.
- λ_3 : Confidence threshold for stopping criterion.
- $k_{\max}$: Sampling budget.

Algorithm 1 - Conformal Sampling with Rejection

Algorithm 1 Conformal sampling with rejection

Definitions: x is an input prompt, $\mathcal{F}$ is our set-based confidence function, $\mathcal{S}$ is our text similarity function, $\mathcal{Q}$ is our sample quality estimator, λ is our threshold configuration, and $k_{\max}$ is our sampling budget. $p_\theta(y \mid x)$ is the conditional output distribution defined by our language model.

function SAMPLE(x, $\mathcal{F}$, $\mathcal{S}$, $\mathcal{Q}$, λ, $k_{\max}$)
    $\mathcal{C}_\lambda \leftarrow \{\}$ ▷ Initialize an empty output set.
    for $k = 1, 2, \ldots, k_{\max}$ **do**
        $y_k \leftarrow y \sim p_\theta(y \mid x)$. ▷ Sample a new response.
        if $\mathcal{Q}(x, y_k) < \lambda_2$ **then** ▷ Reject if its estimated quality is too low.
            continue
        if $\max\{\mathcal{S}(y_k, y_j) \colon y_j \in \mathcal{C}_\lambda\} > \lambda_1$ **then** ▷ Reject if it is too similar to other samples.
            continue
        $\mathcal{C}_\lambda = \mathcal{C}_\lambda \cup \{y_k\}$. ▷ Add the new response to the output set.
        if $\mathcal{F}(\mathcal{C}_\lambda) \geq \lambda_3$ **then** ▷ Check if we are confident enough to stop.
            break
    return $\mathcal{C}_\lambda$

- **Initialize:** Start with an empty prediction set.
- **Sampling:** Generate candidate responses iteratively.
- **Filtering:** Reject low-quality or redundant responses.
- **Stopping:** Stop when confidence is sufficient.

Algorithm 1 - Conformal Sampling with Rejection

Input:

- x : Prompt
- S : Similarity function
- Q : Quality estimator
- λ : Threshold
- $k_{\max}$: Max samples

Loop until stopping criterion is met:

1. Sample response y_k from LLM.
2. **Reject** if $Q(x, y_k) < \lambda_2$ (low quality).
3. **Reject** if $\max S(y_k, y_j) > \lambda_1$.
4. **Add** y_k to prediction set C_λ.
5. **Stop** if confidence score $\mathcal{F}(C_\lambda) \geq \lambda_3$.

Output: C_λ (Prediction Set)

Algorithm 1 Conformal sampling with rejection

Definitions: x is an input prompt, $\mathcal{F}$ is our set-based confidence function, $\mathcal{S}$ is our text similarity function, $\mathcal{Q}$ is our sample quality estimator, λ is our threshold configuration, and $k_{\max}$ is our sampling budget. $p_\theta(y \mid x)$ is the conditional output distribution defined by our language model.

function SAMPLE(x, $\mathcal{F}$, $\mathcal{S}$, $\mathcal{Q}$, λ, $k_{\max}$)
  $\mathcal{C}_\lambda \leftarrow \{\}$ ▷ Initialize an empty output set.
  for $k = 1, 2, \ldots, k_{\max}$ **do**
    $y_k \leftarrow y \sim p_\theta(y \mid x)$. ▷ Sample a new response.
    if $\mathcal{Q}(x, y_k) < \lambda_2$ **then** ▷ Reject if its estimated quality is too low.
      continue
    if $\max\{\mathcal{S}(y_k, y_j) \colon y_j \in \mathcal{C}_\lambda\} > \lambda_1$ **then** ▷ Reject if it is too similar to other samples.
      continue
    $\mathcal{C}_\lambda = \mathcal{C}_\lambda \cup \{y_k\}$. ▷ Add the new response to the output set.
    if $\mathcal{F}(\mathcal{C}_\lambda) \geq \lambda_3$ **then** ▷ Check if we are confident enough to stop.
      break
  return $\mathcal{C}_\lambda$

Optimizing λ with Learn Then Test (LTT)

Goal: Find the optimal threshold configuration λ to ensure reliable prediction sets while maintaining efficiency.

Key Challenges:

- Prediction sets must maintain a controlled risk level ϵ.
- Searching for valid λ values is computationally expensive.

Solution: The **Learn Then Test (LTT)** framework finds the best λ values through statistical risk control.

Steps of LTT Calibration

1. Define Candidate λ Values (Λ) : A set of possible threshold configurations is predefined.

2. Compute Empirical Risk $R_{\mathbf{n}}(\lambda)$:

$$R_{\mathrm{n}}(\lambda) = \frac{1}{n}\sum_{i=1}^{n} L_i(\lambda), \quad \text{where} \quad L_i(\lambda) = \mathbf{1}\left\{ \nexists y \in C_\lambda(X_i) : A_i(y) = 1 \right\}$$

$L_i(\lambda)$ checks if no valid prediction exists in $C_\lambda(X_i)$.

3. Calculate p-values p_λ

$$p_\lambda^{BT} = P(\mathrm{Binom}(n, \epsilon) \leq nR_{\mathrm{n}}(\lambda))$$

This controls the statistical risk.

Selecting the Optimal λ

4. Identify Valid λ Configurations (Λ_{valid})

- Select λ values that satisfy the risk control condition.
- If no valid λ exists, abstain from making predictions.

5. Optimize λ to Balance Set Size and Efficiency

$$\hat{\lambda} = \arg \min_{\lambda \in \Lambda_{\text{valid}}} \frac{1}{n} \sum_{i=1}^{n} \left(\rho_1 |C_\lambda(X_i)| + \rho_2 \frac{[S_\lambda(X_i) - S_\lambda^*(X_i)]^+}{S_\lambda(X_i)} \right)$$

where $S_\lambda(X_i)$ is the total number of samples taken, and $S_\lambda^*(X_i)$ is the index of the first valid generation.

Theorem 4.2: Risk-Controlled Sampling

The selected $\hat{\lambda}$ ensures that the final prediction set satisfies:

$$P(Y \in C_\lambda(X)) \geq 1 - \epsilon$$

Algorithm 2 - Conformal Component Selection

Algorithm 2 Conformal component selection

Definitions: $\mathcal{C}_\lambda$ is a prediction set, $\mathcal{E}$ is an algorithm for splitting candidates y into components, $\mathcal{F}^{\mathrm{c}}$ is a confidence estimator for individual components, γ is our threshold configuration.

function SELECT($\mathcal{C}_\lambda, \mathcal{E}, \mathcal{F}^{\mathrm{c}}, \gamma$)
 $\mathcal{C}_\gamma^{\mathrm{inner}} \leftarrow \{\}$ ▷ Initialize an empty output set.
 for $y \in \mathcal{C}_\lambda$ **do**   ▷ Iterate over full predictions.
  for $e \in \mathcal{E}(y)$ **do**   ▷ Iterate over individual components.
   if $\mathcal{F}^{\mathrm{c}}(e) \geq \gamma$ **then**
    $\mathcal{C}_\gamma^{\mathrm{inner}} \leftarrow \mathcal{C}_\gamma^{\mathrm{inner}} \cup \{e\}$ ▷ Keep only high-confidence components.
 return $\mathcal{C}_\gamma^{\mathrm{inner}}$

Motivation: Long responses contain both correct & incorrect information. Need to identify reliable subcomponents.

Steps:

1. Split text into components (sentences, phrases).
2. Evaluate each component independently using function $\mathcal{F}^c$.
3. Select high-confidence components into $C_\gamma^{\mathrm{inner}}$.

Experiments

Experimental Setup - Tasks & Datasets

Task	Dataset	Model	Evaluation Criteria
Radiology Report Generation	MIMIC-CXR	ViT (Image Encoder) + GPT-2 (Text Decoder)	Clinical Efficacy (CheXbert) + ROUGE-L ≥ 0.4
News Summarization	CNN/DailyMail	Fine-tuned T5-XL	ROUGE-L ≥ 0.35
Open-domain QA	TriviaQA	LLaMA-13B (Few-shot, No Fine-tuning)	Exact Match (Reference vs. Answer)

Experimental Setup - Scoring Functions

Conformal Prediction uses three key scoring functions:

- **Quality Function (Q)**: Evaluates the quality of individual responses.
- **Similarity Function (S)**: Ensures diversity by detecting duplicates.
- **Set Scoring Function (F)**: Measures confidence in the final prediction set.

Quality Function (Q)

Definition: Measures how good an individual response y is.

Defined as:

$$Q(x, y) = p_\theta(y \mid x)$$

but varies by task.

Task-Specific Evaluation Metrics

Task	Quality Metric	Threshold
Radiology Report Generation	ROUGE-L	≥ 0.4
News Summarization	ROUGE-L	≥ 0.35
Open-Domain QA	Exact Match	$= 1$

Similarity Function (S)

Definition: Prevents redundant responses in the prediction set.

- Uses ROUGE-L to compare new samples against existing ones.
- Ensures each new sample is distinct:

$$\max S(y_k, y_j) \leq \lambda_1$$

Set Scoring Function (F)

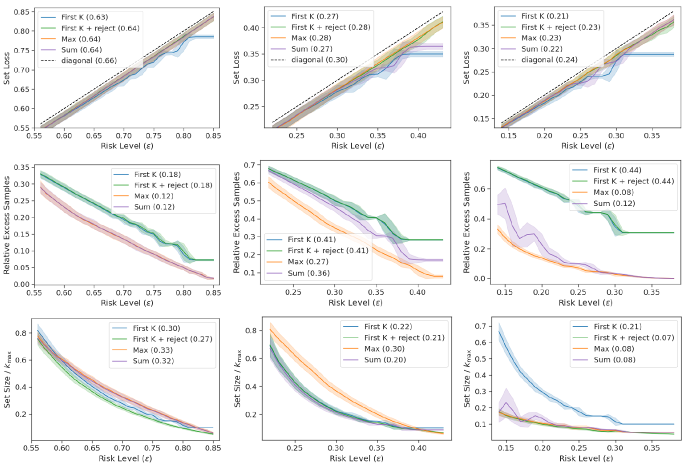
Determines when to stop sampling.

Scoring Function	Definition
FIRST-K	Number of samples taken: $F_{\text{FIRST-K}}(C)$
FIRST-K+REJECT	Same as FIRST-K, but filters duplicates.
MAX	Best individual response: $$F_{\text{MAX}}(C) = \max(Q(y))$$
SUM	Total quality score: $$F_{\text{SUM}}(C) = \sum_{y \in C} Q(y)$$

Experimental Setup - Metrics

- **Set Loss:** Measures the probability that the prediction set fails to contain a correct answer.
  - Ensures the loss does not exceed the predefined risk threshold ϵ.
  - Example: If $\epsilon = 0.05$, the model guarantees 95% coverage of correct answers.
- **Excess Samples:** Evaluates unnecessary sampling beyond the first correct response.
  - Over-sampling increases computational cost and inefficiency.
  - Includes redundant responses or continued sampling after a correct answer is found.
- **Final Set Size:** Assesses the size of the final prediction set.
  - Large sets may contain diverse answers but reduce interpretability.
  - Small sets are more precise but risk missing correct answers.
  - The goal is to maintain an optimal balance between accuracy and efficiency.
- **Computes Area Under the Curve (AUC) over ϵ or α.**

Experimental Results



(a) MIMIC-CXR (b) CNN/DM (c) TriviaQA

Figure 2: Conformal sampling results for C_λ as a function of ϵ. We report the loss, relative excess samples, and overall size (normalized by $k_{\max}$). We also report the AUC over achieved/non-trivial ϵ.

Experimental Results

Conformal Sampling Validity

- Set Loss must not exceed the target risk level.
- All methods remain below the diagonal line, confirming theoretical validity.

Sampling Efficiency

- **TriviaQA:** MAX and SUM reduce set size significantly.
- **Long-text tasks:** MAX is more efficient than SUM and FIRST-K.
- FIRST-K+REJECT reduces redundancy but lacks full efficiency.

Component-Based Selection (Appendix G)

- Long text responses may mix correct and incorrect info.
- Selecting the most reliable components improves response quality.

Conclusion

Conclusion

This study proposes a method to enhance the reliability of Language Models by constructing statistically guaranteed prediction sets.

Key Contributions:

- Bridges conformal prediction and LMs by calibrating output set sampling.
- Extends multi-label conformal prediction to identify reliable components in long texts.
- Achieves valid risk control across diverse tasks while ensuring efficient and precise output sets.

Thank You