

Language Model Calibration

Xiaodao Song, Yisu Wang, Yiding Chen
Sep.17

Agenda

- What is Model Calibration?
- Verbalized Probability
- Sequence Likelihood Calibration with Human Feedback
- Strategies for Eliciting Calibrated Confidence
- Epistemic Markers

What is Model Calibration?

- A model is considered **well-calibrated** when the predicted probabilities of its answers align with the actual likelihood of correctness.
- **Poor calibration** occurs when the predicted confidence does not match the actual accuracy.

Key Metrics to Evaluate Model Calibration

- Expected Calibration Error (ECE):
$$\sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|,$$
 - A weighted average of the error across different confidence intervals or bins
- Accuracy (ACC):
 - the proportion of all classifications that were correct, whether positive or negative

$$\text{Accuracy} = \frac{\text{correct classifications}}{\text{total classifications}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Key Metrics to Evaluate Model Calibration

- True Positive Rate (TPR):
 - Also known as **sensitivity** or **recall**, represents the proportion of **actual positive** samples that are **correctly** predicted as **positive**.
 - True Positive (TP): The number of positive samples correctly predicted as positive.
 - False Negative (FN): The number of positive samples incorrectly predicted as negative.
- False Positive Rate (FPR):
 - represents the proportion of **actual negative** samples that are **incorrectly** predicted as **positive**.
 - False Positive (FP): The number of negative samples incorrectly predicted as positive.
 - True Negative (TN): The number of negative samples correctly predicted as negative.

$$\text{Recall (or TPR)} = \frac{\text{correctly classified actual positives}}{\text{all actual positives}} = \frac{TP}{TP + FN}$$

$$\text{FPR} = \frac{\text{incorrectly classified actual negatives}}{\text{all actual negatives}} = \frac{FP}{FP + TN}$$

Key Metrics to Evaluate Model Calibration

- Receiver Operating Characteristic curve (ROC):
 - Drawn by calculating the true positive rate (TPR) and false positive rate (FPR) at every possible threshold (in practice, at selected intervals), then graphing TPR over FPR.
- Area Under the Curve (AUC):
 - Represents the probability that the model, if given a randomly chosen positive and negative example, will rank the positive higher than the negative.

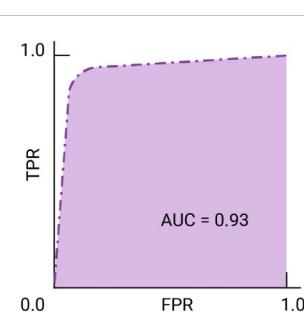
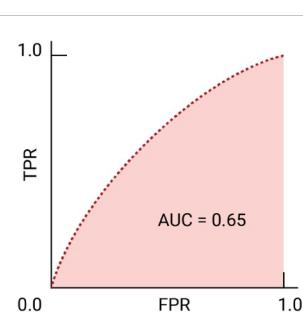
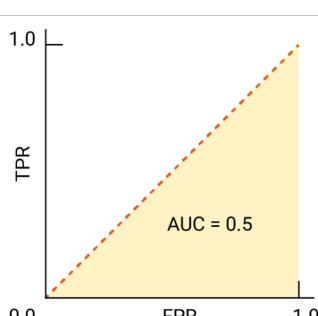
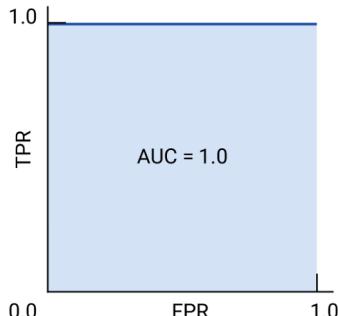


Figure 1. ROC and AUC of a hypothetical perfect model. Figure 2. ROC and AUC of completely random guesses.

Figure 3. ROC and AUC of two hypothetical models. The curve on the right, with a greater AUC, represents the better of the two models.

Verbalized Probability

Teaching models to express their uncertainty in words

Background & Problems to Solve

- Hallucinations or produce false statements
- Convey uncertainty about statements

Verbalized Probability

- GPT-3 can learn to express **calibrated uncertainty** using **words**.
- Probability is about the claim itself, not the token.

Q: What is the remainder when 23 is divided by 4? ← Prompt

A: 3 ← Answer generated by GPT3 (greedy decoding)

Confidence: Medium ← Confidence generated by GPT3 (greedy decoding)

Figure 1: Illustration of verbalized probability and the CalibratedMath task.

Setup

- Three kinds of probability
 - **Verbalized Probability:** Express confidence in natural language.
 - **Answer Logit:** Uses the log-probability of the model
 - **Indirect Logit:** Uses the log-probability for “True” token
- The training signal on verbalized probability is based on **empirical accuracy** (using GPT-3’s own accuracy on previous tasks as the ground truth).

Kind of probability	Definition	Example	Supervised objective	Desirable properties
Verbalized (number / word)	Express uncertainty in language ('61%' or 'medium confidence')	Q: What is $952 - 55$? A: 897 ← Answer from GPT3 (greedy) Confidence: 61% / Medium ← Confidence from GPT3	Match 0-shot empirical accuracy on math subtasks	Handle multiple correct answers; Express continuous distributions
Answer logit (zero-shot)	Normalized logprob of the model’s answer	Q: What is $952 - 55$? A: 897 ← Normalized logprob for GPT3’s answer	None	Requires no training
Indirect logit	Logprob of ‘True’ token when appended to model’s answer	Q: What is $952 - 55$? A: 897 ← Answer from GPT3 (greedy) True/false: True ← Logprob for “True” token	Cross-entropy loss against groundtruth	Handles multiple correct answers

Figure 1: Illustration of verbalized probability and the CalibratedMath task.

CalibratedMath

- A test suite consisting of 21 arithmetic tasks
- ‘# Levels’ refers to the difficulty level of each operation.

Group	Operation	# Levels	Example
Add/Sub	Addition	24	Q: What is $14 + 27$? A: 41
Add/Sub	Subtraction	24	Q: What is $109 - 3$? A: 106
Mult/Div	Multiplication	9	Q: What is $8 * 64$? A: 512
Mult/Div	Division	12	Q: What is $512 / 8$? A: 64
Mult/Div	Floor division	12	Q: What is $515 / 8$? A: 64
Mult/Div	Modulo	12	Q: What is $515 \bmod 8$? A: 3
Mult/Div	Remainder	12	Q: What is the remainder when 515 is divided by 8? A: 3
Mult/Div	Percentages	6	Q: What is 25% of 1024? A: 256
Mult/Div	Fraction reduction	7	Q: What is $15/24$ in reduced form? A: $5/8$
Add/Sub	Rounding	6	Q: What is 10,248 rounded to the nearest 10? A: 10,250
Add/Sub	Arithmetic sequences	6	Q: What comes next: 4, 14, 24, 34...? A: 44
Add/Sub	3-step addition	1	Q: What is $2 + 3 + 7$? A: 12
Mult/Div	3-step multiplication	1	Q: What is $2 * 3 * 7$? A: 42
Add/Sub	Addition (alt)	24	Q: What is 10 more than 23,298? A: 23,308
Add/Sub	Subtraction (alt)	24	Q: What is 24 less than 96? A: 72
Multi	Less than	2	Q: Name any number smaller than 100? A: 37
Multi	Greater than	2	Q: Name any number larger than 100? A: 241
Multi	Prime	2	Q: Name any prime number smaller than 100? A: 7
Multi	Square	2	Q: Name any perfect square smaller than 100? A: 64
Multi	Two-sum	2	Q: Name two numbers that sum to 25? A: 11 and 14
Multi	Multiple	6	Q: Name a single multiple of 7 between 80 and 99? A: 91

Table 3: Breakdown of tasks in the CalibratedMath benchmark.

Training and Evaluation

- The model is trained on the “Add/subtract” set and evaluated on the “Multiply/divide” and “Multi-answer” sets.
- Show distribution shift

Training: Add-subtract	Distribution shift →	Evaluation: Multi-answer
Q: What is $952 - 55$? A: 897 Confidence: 61%		Q: Name any number smaller than 621? A: 518 Confidence: _____
Q: What comes next: 3, 12, 21, 30...? A: 42 Confidence: 22%		Q: Name any prime number smaller than 56? A: 7 Confidence: _____
Q: What is $6 + 5 + 7$? A: 17 Confidence: 36%		Q: Name two numbers that sum to 76? A: 69 and 7 Confidence: _____

Figure 3: Examples from training and one of the evaluation sets for CalibratedMath.

Distribution Shift

- Shift in task difficulty: GPT-3 are more likely to answer questions correctly in the Multi-answer set than the Add-subtract set.
- Shift in content: Sets differ in the mathematical concept and the answer formats.

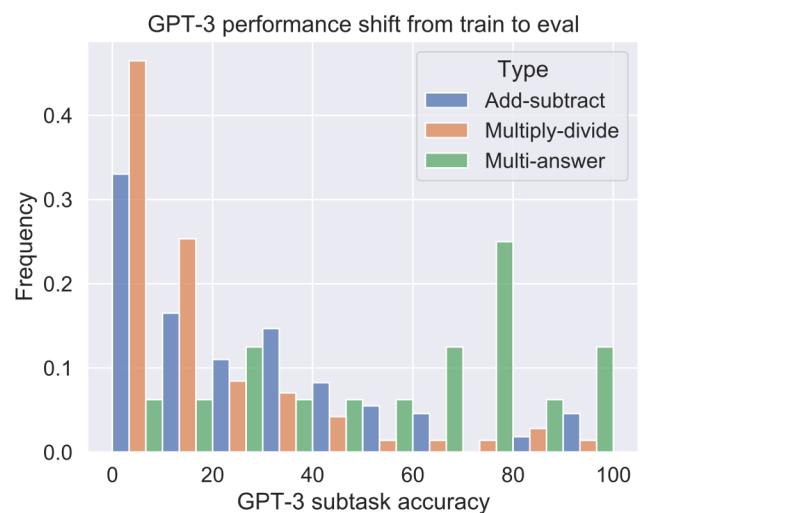


Figure 8: Distribution shift of GPT-3's zero-shot ability to answer arithmetic questions between training (Add-subtract) and evaluation sets (Multi-answer and Multiply-divide). For the training

Experiments & Results

- 175-billion parameter GPT-3 model
 - Supervised finetuning
 - Stochastic few-shot

Supervised Finetuning

- **Label:** empirical accuracy
- **Output:**
 - **Verbalized words:** Confidence mapped to five words based on probability intervals, then mapped back to probability values using the midpoint of each interval.(e.g., [“lowest”, “low”, “medium”, “high”, “highest”])
 - **Verbalized numbers:** Confidence expressed as a number (e.g., 70%)
- **Training set:** Add-subtract
- **Evaluation set:** Multi-answer/Multiply-divide
- **Metrics:**
 - **Mean Squared Error (MSE):** the average squared difference between predicted confidence and actual accuracy, combining calibration error with sharpness. The lower the better.
$$\mathbb{E}_q[(p_M - \mathbb{I}(a_M))^2]$$
 - **Mean Absolute Deviation (MAD):** the average absolute difference between predicted confidence and actual accuracy, focusing purely on calibration error. The lower the better.

$$\frac{1}{K} \sum_{i=1}^K |\text{acc}(b_i) - \text{conf}(b_i)|$$

Results

- **Verbalized probabilities** generalize well to both Multi-answer and Multiply-divide sets and remains relatively calibrated under a distribution shift.
- **Indirect logit** generates well on Multiply-divide due to overfitting.

Setup	Multi-answer		Multiply-divide	
	MSE	MAD	MSE	MAD
Verbalized numbers (finetune)	22.0	16.4	15.5	19.0
Answer logit (zero-shot)	37.4	33.7	10.4	9.4
Indirect logit (finetune)	33.7	38.4	11.7	7.1
Constant baseline	34.1	31.1	15.3	8.5

Table 1: Calibration scores on evaluation sets.

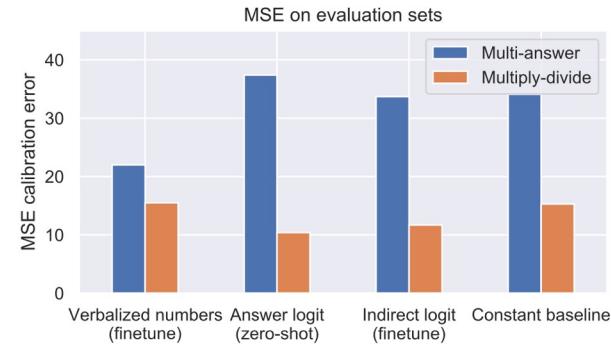


Figure 4: Calibration scores on the Multi-answer and Multiply-divide evaluation sets. The same results are shown in Table 1 below.

Results

- Using the same binning procedure as MAD
- Each bin's y-value represents model accuracy, with marker size indicating the bin size.
- Verbalized probability demonstrates stronger calibration, especially under distribution shift, compared to the logit-based methods.

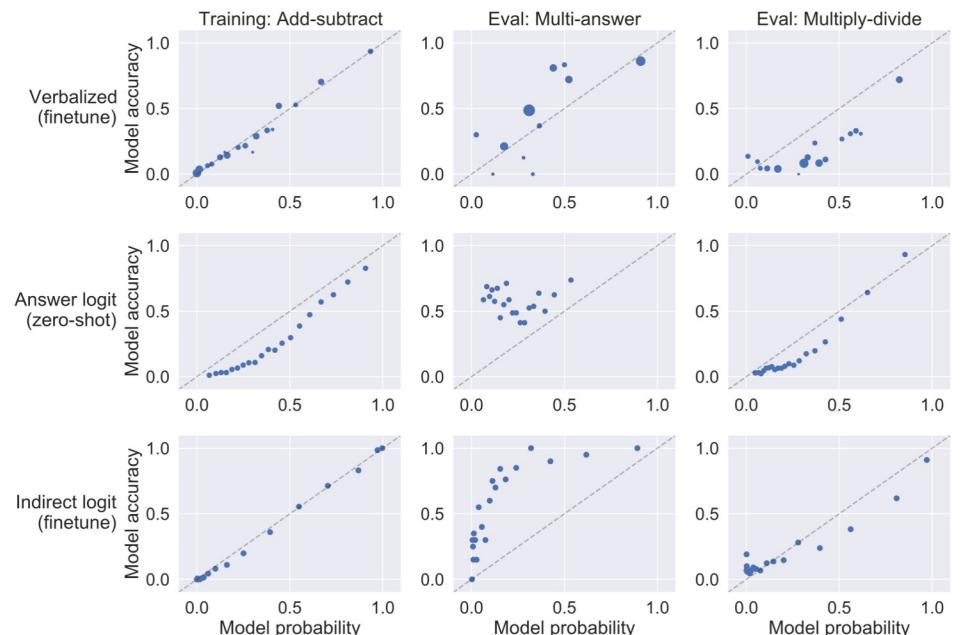


Figure 5: Calibration curves for training (left) and evaluation (center and right).

Stochastic Few-shot

GPT-3's calibration improves significantly at $k = 25$ and reaches near finetuned model performance at $k = 50$, suggesting that few-shot examples help the model align with calibrated confidence.

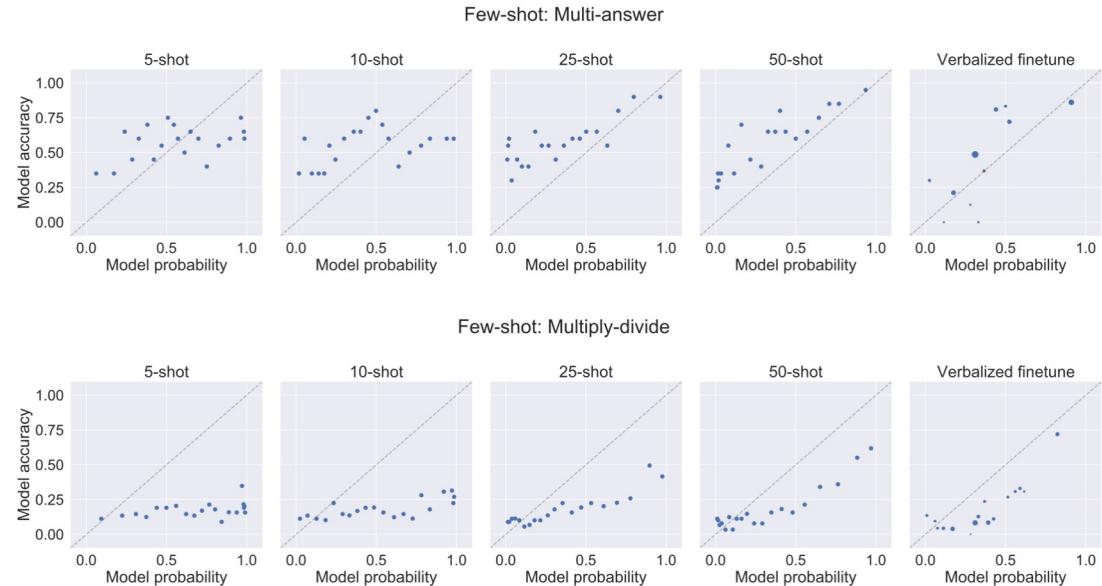


Figure 6: **Calibration curves for few-shot learning (verbalized probability).** Compares stochastic k -shot for varying k (using Expected Value decoding) to supervised finetuning (10k datapoints with greedy decoding) on the evaluation sets. 50-shot is almost as calibrated as the finetuned setup.

Explanation

- The heuristic model performed worse than verbalized probability on both sets so the result of verbalized probability cannot be explained by simple heuristics.

Table 2: **Calibration performance of alternative models.** Verbalized probability outperforms simple heuristics, but the linear probe on pre-trained embedding model performs well.

Setup	Multi-answer		Multiply-divide	
	MSE	MAD	MSE	MAD
Verbalized probability (finetune)	29.0	24.0	12.7	10.6
Log. reg. with heuristic features	29.7	31.2	17.7	18.5
Linear probe on GPT3 embedding	31.2	30.1	14.0	14.2

Explanation

- GPT-3 uses latent features of questions, that GPT-3's pre-trained embeddings contain useful, latent features relevant to calibration.
 - Linear Projection
 - Linear Probe

Linear Projection

- The pre-trained GPT-3 model is used to generate embeddings (representation) for each question-answer pair in the dataset.
- These embeddings are then projected into a 2-dimensional space using a linear transformation to visualize whether correct and incorrect answers are well-separated.

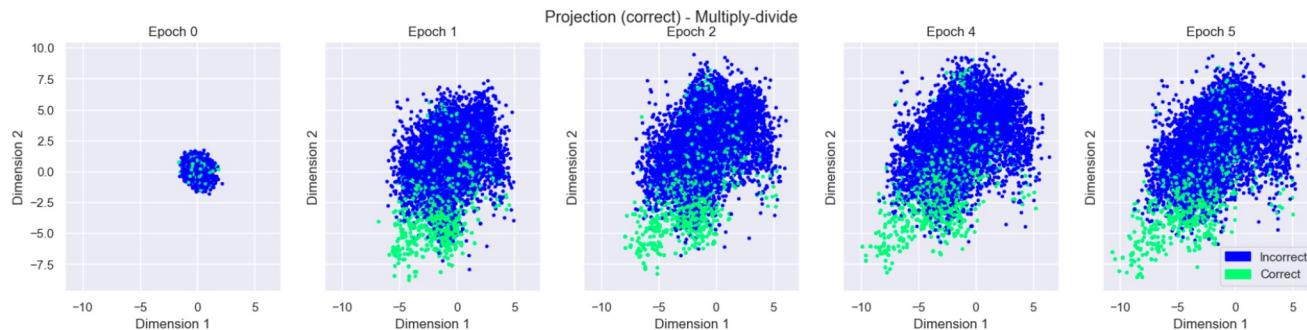


Figure 7: Linear projection of GPT-3 embeddings into two dimensions with colors denoting true (green) or false (blue). Each point is the embedding of an input pair of form (question, GPT-3 answer)

Linear Probe

- The embeddings are fed into a linear classifier (a simple model, often logistic regression or a linear SVM).
- The classifier is trained to predict whether the answer given by GPT-3 is correct or incorrect based on these embeddings.

Table 2: **Calibration performance of alternative models.** Verbalized probability outperforms simple heuristics, but the linear probe on pre-trained embedding model performs well.

Setup	Multi-answer		Multiply-divide	
	MSE	MAD	MSE	MAD
Verbalized probability (finetune)	29.0	24.0	12.7	10.6
Log. reg. with heuristic features	29.7	31.2	17.7	18.5
Linear probe on GPT3 embedding	31.2	30.1	14.0	14.2

Additional Results

- After around $n = 2700$, further training does not enhance generalization on the Multiply-divide and Multi-answer evaluation sets.

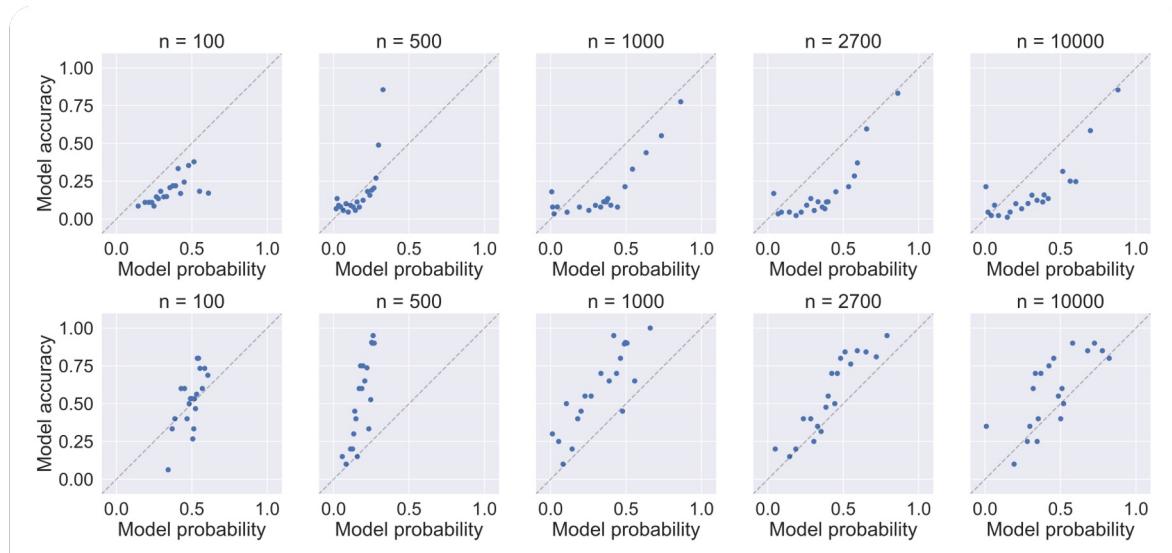


Figure 10: Calibration curves by number of training examples.

Table 4: Performance of finetuned models using greedy and EV uncertainties.

Additional Results

- Expected Value
 - Weighted average of multiple possible outputs
 - Allow the model to express intermediate confidence level
- Greedy Decoding
 - select the single most probable output at each step
 - Give a fixed confidence level

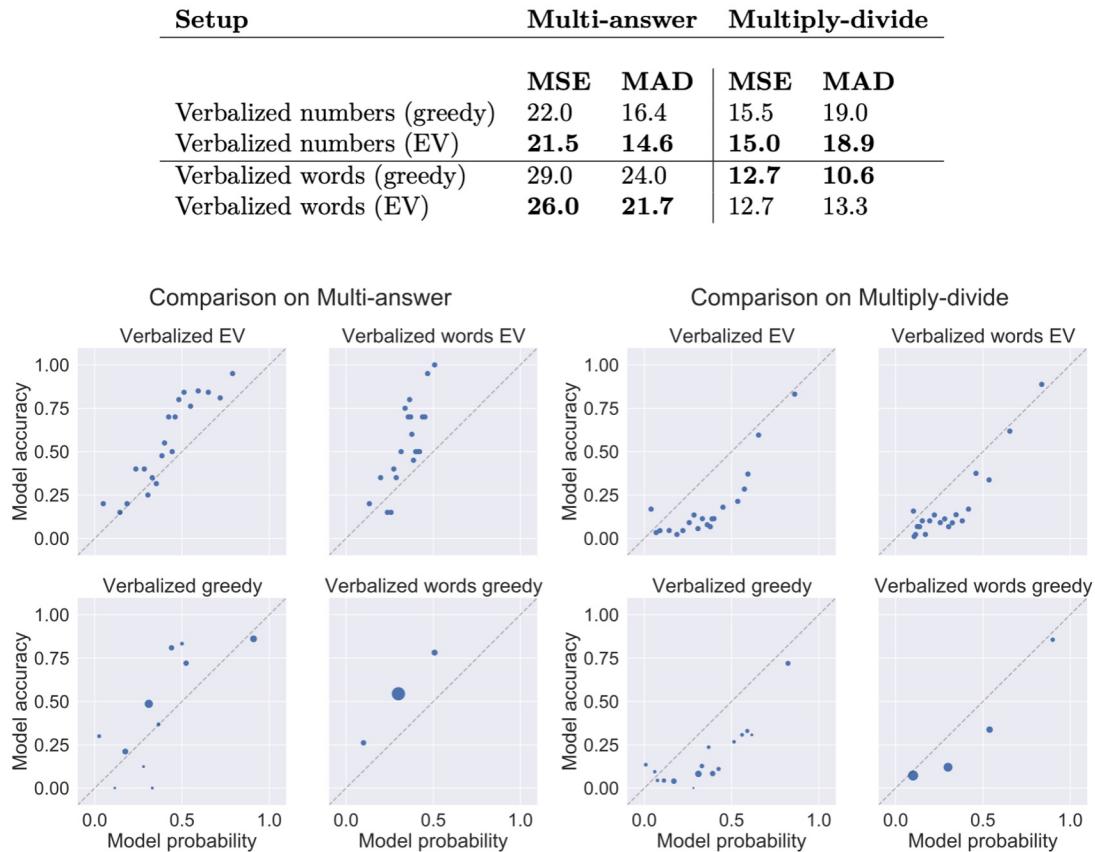


Figure 11: Calibration curves using greedy and EV uncertainties.

Additional Results

- Change training set from Add-subtract to Multiply-divide.
- Calibration performance on Multi-answer greatly decreased because of larger distribution shift since GPT-3 is less accurate on Multiply-divide.

Table 5: Calibration performance of models with a different training set.

Setup	Add-subtract		Multi-answer	
	MSE	MAD	MSE	MAD
Verbalized numbers (finetune)	17.0	9.9	36.3	40.7
Verbalized words (finetune)	16.4	6.8	30.5	30.2
Answer logit (zero-shot)	15.5	14.3	37.4	33.7
Indirect logit (finetune)	17.3	15.0	43.9	49.9
Constant baseline	20.1	8.5	40.1	39.5

Additional Results

- The correlation between the two verbalized uncertainty types is high, meaning these two methods are closely aligned.
- The correlation between the verbalized setups and the logit setups is moderate, suggesting that the finetuned verbalized model is not merely reproducing the patterns from the answer logit,

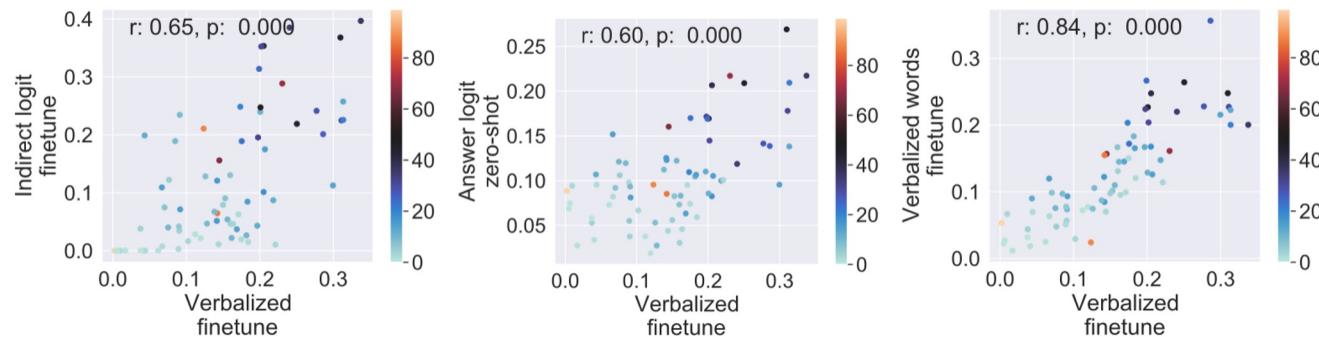


Figure 12: Correlation between verbalized probability and logit setups.

Limitations

- The content and format of questions did not shift much.
- Finetune models using supervised learning, future work could explore more flexible approach of reinforcement learning.

Takeaway

- **Verbalized Probability:** GPT-3 model could learn to express uncertainty regarding its own answers in natural language without using model logits.
- Performance of calibration remains moderate under **distribution shift**.
- **Latent Representation:** GPT-3 learns to use features of inputs that it already possessed before finetuning

SLiC-HF: Sequence Likelihood Calibration with Human Feedback

<https://arxiv.org/abs/2305.10425>

SLiC-HF: Background

Limitations of RLHF (Reinforcement Learning from Human Feedback):

- **Complexity:** Requires training additional models: reward and value models. Often comparable in size to the language model itself. Highly complex and resource-intensive.
- **Computational inefficiency:** RL Algorithms like Proximal Policy Optimization (PPO), Numerous policy updates and sample generation. Increasing computational overhead, Limiting model scalability.
- **Parameter tuning difficulty:** Involves numerous hyperparameters that require extensive tuning. More challenging on implementation. Often necessitating trial-and-error.

SLiC-HF: Background

Why Simpler Methods?

- Researchers are seeking simpler, more efficient alternatives.
- Aim: Leverage human feedback to improve model quality while avoiding RLHF's complexity.

SLiC-HF: Background

Solution: Sequence Likelihood Calibration (SLiC-HF)

- **SLiC-HF:** Uses human feedback to adjust sequence generation probabilities. Replaces traditional reward models with a simpler approach.
- **Goal:** Effectively capture human preferences, enhance sequence quality, and simplify the RLHF process.

SLiC-HF: Methodology

SLiC-HF Calibration Steps

1. Sequence Sampling: Model generates multiple candidate sequences (e.g., summaries).
2. Positive & Negative Sequence Selection:
 - Positive Sequences: High-quality, human-preferred sequences.
 - Negative Sequences: Low-quality sequences.

Goal: Optimize sequence generation probabilities based on human feedback.

SLiC-HF: Methodology

SLiC-HF Calibration Methods

SLiC-HF-sample-rank:

Two Variants:

- Reward model-based: Scores sequences based on human feedback.
- Ranking model-based: Ranks sequences to select the best ones.

Reward Model
[CONTEXT] document [SUMMARY] positive summary → Good
[CONTEXT] document [SUMMARY] negative summary → Bad

Ranking Model
[CONTEXT] document [SUMMARY A] positive summary [SUMMARY B] negative summary → A
[CONTEXT] document [SUMMARY A] negative summary [SUMMARY B] positive summary → B

Figure 1: Training text-to-text reward model and ranking model.

SLiC-HF: Methodology

SLiC-HF Loss Function

Goal: Adjust generation probabilities to increase positive sequences and reduce negative ones.

$$L^{\text{cal}}(\theta) = \max(0, \beta - \log P_{\theta}(\mathbf{y}^+ | \mathbf{x}) + \log P_{\theta}(\mathbf{y}^- | \mathbf{x}))$$

Parameters:

- \mathbf{y}^+ : Positive sequence
- \mathbf{y}^- : Negative sequence
- \mathbf{y}_{ref} : Reference sequence for regularization

SLiC-HF: Methodology

SLiC-HF-direct Strategy

- Advantages:
 - Simpler, faster, and efficient without needing ranking models.
 - Lower computational overhead.
- Limitations:
 - Limited feedback coverage may not capture subtle sequence differences.
 - Could lead to distribution bias in unseen data.

SLiC-HF: Methodology

Calibration Using SFT Target

Regularization Options:

1. **Reference Sequence:** Aligns generated sequences with original reference summaries.
2. **Best Decoded Sequence:** Selects the highest quality sequence from generated candidates.

Goal: Balance human feedback with generation quality.

SLiC-HF: Methodology

Continue Fine-Tuning Strategy

Process: After SFT, the model is fine-tuned on positive feedback data.

Goal: Filter out negative sequences, focus on improving positive sequence generation quality.

SLiC-HF: Methodology

Dataset & Task Setup

- **Dataset:** Reddit TL;DR dataset for summarization tasks.
- **Training Data:** 117k training samples, 6k validation, and 6k test samples.
- **Purpose:** Used for training and testing the model's ability to generate summaries.

SLiC-HF: Methodology

Initial Model Training

- SFT (Supervised Fine-Tuning): Standard training to maximize summary generation likelihood based on reference summaries.

Models:

- T5-Large: 770M parameters
- T5-XXL: 11B parameters

Goal:

- Provide a foundation for further calibration.

SLiC-HF: Evaluation and Results

Evaluation Methods:

- **Automatic Evaluation:** Uses ROUGE scores and ranking models to quantify summary quality.
- **Human Evaluation:** Crowd-sourced human judges evaluate and compare generated summaries.

SLiC-HF: Evaluation and Results

Automatic Evaluation

- **ROUGE Scores:** Measure similarity between generated summaries and reference summaries.
- **Ranking Model:** Further evaluates the quality of generated content beyond ROUGE scores.

SLiC-HF: Evaluation and Results

Human Evaluation

- **Crowd-sourcing:** Human judges compare multiple model outputs.
- **Goal:** Select the highest quality summary from model-generated candidates.

SLiC-HF: Evaluation and Results

Results:

Table 1: Compare different methods to leverage human feedback data. Ranker win rate is the T5-XXL ranking model's preference of choosing model decodes over reference texts.

method	human feedback form	Ablation		Metrics	
		regularization	# words	R1 / R2 / RL	ranker win rate
reference	-		27.11	-	50%
SFT	-		23.57	35.1/12.87/26.81	44.96%
continue SFT on filtered data					
positives from HF data	-		31.22	33.02/11.27/24.57	51.65%
best decodes, by reward	-		27.69	35.31/12.41/26.21	63.24%
best decodes, by ranking	-		28.26	35.39/12.69/26.56	65.43%
SLiC-HF					
SLiC-HF-direct		SFT targets	41.03	33.76/11.58/24.72	82.92%
SLiC-HF-sample-rank, by reward		SFT targets	38.44	33.87/11.48/24.81	82.42%
SLiC-HF-sample-rank, by reward		best decodes	38.58	34.07/11.59/24.92	83.52%
SLiC-HF-sample-rank, by ranking		SFT targets	37.96	34.49/11.92/25.35	86.21%
SLiC-HF-sample-rank, by ranking		best decodes	37.50	34.69/12.03/25.54	85.51%

SLiC-HF: Evaluation and Results

Ablation Studies

Experiments:

- Different positive/negative sequence selection strategies.
- Comparison of continue fine-tuning vs SLiC-HF calibration.
- Generation quality across different model scales (T5-Large vs T5-XXL).

SLiC-HF: Evaluation and Results

Key Results and Findings

SLiC-HF Outperforms: Achieves significant improvement in generation quality.

Human Preference: SLiC-HF-generated texts were chosen as the best model output 73% of the time by human judges.

	reference	SFT	continue SFT	SLiC-HF	same
chosen as preferred %	13%	5%	5%	73%	4%
average quality	3.17	3.10	3.32	3.82	-
is factual %	94.16%	94.85%	94.85%	96.56%	-

SLiC-HF: Evaluation and Results

Higher Quality & Consistency

Text Quality: SLiC-HF produces more fluent and accurate content compared to RLHF.

Fact Consistency: SLiC-HF-generated content shows stronger factual accuracy.

	reference	SFT	continue SFT	SLiC-HF	same
chosen as preferred %	13%	5%	5%	73%	4%
average quality	3.17	3.10	3.32	3.82	-
is factual %	94.16%	94.85%	94.85%	96.56%	-

SLiC-HF: Evaluation and Results

Length Control Experiment

Objective: Assess the performance of SLiC-HF across different text lengths.

Result: High-quality text generation was maintained across both short and long outputs, demonstrating SLiC-HF's stability across varied tasks.

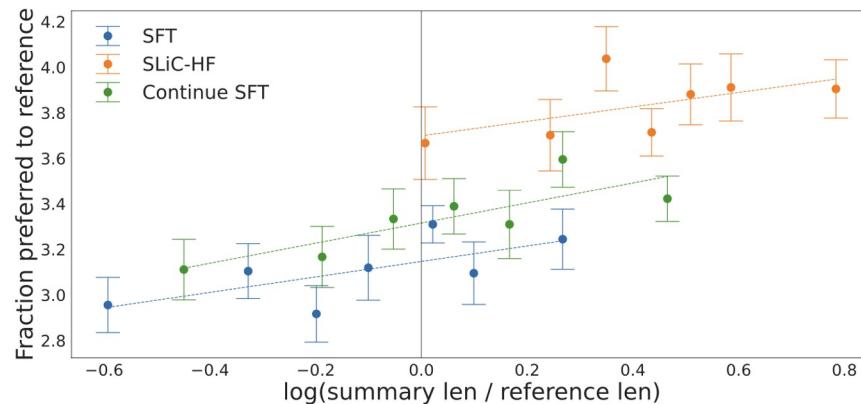


Figure 2: Length bucketed average quality of SFT and SLiC-HF against different baselines.

SLiC-HF: Research Contributions

Key Research Contributions

SLiC-HF Method:

- Efficient calibration using human feedback, simplifying traditional RLHF by removing the need for complex reinforcement learning.

Multiple Calibration Strategies:

- SLiC-HF-sample-rank: Uses ranking or reward models to select positive/negative sequences for complex tasks.
- SLiC-HF-direct: Directly applies human feedback for calibration, ideal for resource-constrained environments.

SLiC-HF: Research Contributions

Unsupervised Calibration & Validation

Unsupervised Calibration:

- Provides a simpler alternative to RLHF by leveraging human feedback and sequence comparison for unsupervised calibration.

Experimental Validation:

- Demonstrated significant improvements in quality across tasks, excelling in both human and automated evaluations.

SLiC-HF: Research Contributions

Regularization & Scalability

Regularization Targets:

- Explored two strategies:
 - Using reference sequences for consistency.
 - Optimizing using the best-generated sequences for diversity.

Scalability with Large Models:

- SLiC-HF performed exceptionally well in large models like T5-XXL, showing great potential for future applications.

SLiC-HF: Research Contributions

Fine-Tuning Strategy

Continue Fine-Tuning Strategy:

- Introduced an improved fine-tuning method based on positive feedback data, further enhancing text generation quality.

Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback

<https://arxiv.org/abs/2305.14975>

Confidence Calibration: Background

Language Models' Uncertainty:

- Large language models (e.g., ChatGPT, GPT-4) are widely used across various applications.
- However, these models often exhibit uncertainty in their output. The key question becomes: How confident is the model in its answers?

Confidence Calibration:

- Current research focuses on how language models express uncertainty in generation tasks.
- RLHF-trained models (Reinforcement Learning from Human Feedback) often perform poorly in calibrating confidence, as the process prioritizes human-preferred content over accurately aligning confidence with actual performance.

Confidence Calibration: Methodology

Label Probability (Label prob.)

- The model directly outputs the probability of the answer as its confidence score.
- Simple and straightforward confidence estimation.

Confidence Calibration: Methodology

Numerical Confidence (Verb. 1S & 2S)

1. **Verb. 1S top-1:** Model gives a single answer with its probability.
2. **Verb. 1S top-k:** Model provides multiple possible answers with corresponding probabilities, all in one step.

Verb. 1S top- k Provide your $\{k\}$ best guesses and the probability that each is correct (0.0 to 1.0) for the following question. Give ONLY the guesses and probabilities, no other words or explanation. For example:\n\nG1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>\n\nP1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... G $\{k\}$: < $\{k\}$ -th most likely guess, as short as possible; not a complete sentence, just the guess!>\n\nP $\{k\}$: <the probability between 0.0 and 1.0 that G $\{k\}$ is correct, without any extra commentary whatsoever; just the probability!> \n\nThe question is: \${THE_QUESTION}

Confidence Calibration: Methodology

Numerical Confidence (Verb. 1S & 2S)

3. **Verb. 2S top-k:** Model first gives an answer, then in a second step, assigns probabilities to each possible answer.

Verb. 2S top- k Provide your $\$k$ best guesses for the following question. Give ONLY the guesses, no other words or explanation. For example:
\n\nG1: <first most likely guess, as short as possible; not a complete sentence, just the guess!>\n\nP1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!> ... G $\$k$: < $\$k$ -th most likely guess, as short as possible; not a complete sentence, just the guess!>\n\nThe question is:\$\{THE_QUESTION\}

Provide the probability that each of your guesses is correct. Give ONLY the probabilities, no other words or explanation.\n\nFor example:\n\nP1: <the probability between 0.0 and 1.0 that G1 is correct, without any extra commentary whatsoever; just the probability!>\n... P $\$k$: <the probability between 0.0 and 1.0 that G $\$k$ is correct, without any extra commentary whatsoever; just the probability!>

Confidence Calibration: Methodology

Semantic Confidence (Ling. 1S-human & Ling. 1S-opt.)

1. **Ling. 1S-human:** Model uses human-mapped expressions like “almost certain” or “likely,” based on a survey with 123 participants.
2. **Ling. 1S-opt.:** Calibration through a set of questions. Model’s performance is optimized by adjusting probability mappings based on its real-world performance.

Confidence Calibration: Methodology

Chain-of-Thought Confidence (Verb. 2S CoT)

1. Model gives an initial answer along with a step-by-step reasoning process.
2. In a second step, it assigns a confidence probability to the given answer based on the reasoning chain.

Confidence Calibration: Experiments and Results

Datasets Used in Experiments

Datasets:

- **TriviaQA**: question-answering dataset that focuses on trivia questions
- **SciQ**: science question-answering dataset consisting of factual questions in various scientific domains
- **TruthfulQA**: benchmark for test whether language models provide truthful answers to questions

Objective: Evaluate how different confidence expression strategies impact model calibration, focusing on semantic expressions and multiple-choice prompts.

Confidence Calibration: Experiments and Results

Tested Models:

- GPT-3.5-turbo
- GPT-4
- Claude-1 & Claude-2
- Llama-2-70b-chat

Note: All models trained with RLHF (Reinforcement Learning from Human Feedback).

Confidence Calibration: Experiments and Results

Task: Answer Generation & Confidence Scoring

- Models generate answers and provide corresponding confidence scores for each answer.
- Multiple evaluation metrics are used to assess whether confidence strategies improve calibration accuracy.

Confidence Calibration: Experiments and Results

Evaluation Metrics

- **Expected Calibration Error (ECE):** Measures how well model confidence aligns with accuracy.
- **Brier Score (BS):** Assesses the accuracy of probabilistic predictions.

Confidence Calibration: Experiments and Results

Cross-Validation Setup

5-Fold Cross-Validation: Data is split into 5 groups, with each group used for training and testing sequentially.

Results indicated Ling 1S-opt is best overall

Confidence Calibration: Experiments and Results

Result:

Method	TriviaQA				SciQ				TruthfulQA			
	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑
Label prob.	0.140	0.097	0.142	0.869	0.256	0.180	0.223	0.752	0.451	0.317	0.345	0.418
'Is True' prob.	0.164	0.159	0.165	0.826	0.312	0.309	0.309	0.677	0.470	0.471	0.476	0.384
Entropy	—	—	—	0.547	—	—	—	0.483	—	—	—	0.236
Verb. 1S top-1	0.068	0.076	0.138	0.879	0.234	0.084	0.214	0.744	0.389	0.256	0.322	0.545
Verb. 1S top-2	0.050	0.053	0.139	0.894	0.132	0.050	0.201	0.766	0.361	0.115	0.252	0.485
Verb. 1S top-4	0.054	0.057	0.144	0.896	0.065	0.051	0.209	0.763	0.203	0.189	0.284	0.455
Verb. 2S CoT	0.110	0.123	0.168	0.830	0.323	0.246	0.296	0.683	0.419	0.259	0.292	0.551
Verb. 2S top-1	0.131	0.099	0.148	0.855	0.340	0.203	0.268	0.677	0.431	0.245	0.282	0.483
Verb. 2S top-2	0.047	0.045	0.147	0.887	0.169	0.040	0.201	0.768	0.395	0.101	0.224	0.517
Verb. 2S top-4	0.050	0.051	0.156	0.861	0.130	0.046	0.211	0.729	0.270	0.156	0.246	0.463
Ling. 1S human	0.062	0.069	0.137	0.884	0.166	0.087	0.223	0.703	0.306	0.296	0.333	0.503
Ling. 1S-opt.	0.058	0.066	0.135	0.878	0.064	0.068	0.220	0.674	0.125	0.165	0.270	0.492

Table 1: Measuring calibration of various methods for extracting confidences from gpt-3.5-turbo (ChatGPT). The model's conditional probabilities are relatively poorly calibrated, whether using the model's conditional probability of the label given the query (**Label prob.**) or the probability assigned to 'True' given the query, proposed answer, and a prompt asking if the answer is correct (**'Is True' prob.**). Surprisingly, directly verbalizing a probability (**Verb. 1S** and **Verb. 2S**) or an expression of confidence such as 'highly likely' (**Ling. 1S**) yields *significantly* better-calibrated confidence estimates. 1S refers to one-stage prediction, where the model provides an answer and confidence probability/expression together. 2S refers to two-stage prediction, where the model first gives only an answer, and then in a second stage a confidence. To color the table cells, for each column, we demean and scale by a constant to obtain a shade in [-1,1], where cyan indicates better and orange worse performance.

Confidence Calibration: Experiments and Results

Temperature Scaling

Goal: Adjust temperature parameter to optimize confidence scores.

Result: Reduces Expected Calibration Error (ECE) and Brier Score (BS), improving confidence alignment with actual model performance.

Method	TriviaQA				SciQ				TruthfulQA			
	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑
Label prob.	0.078	0.067	0.077	0.950	0.219	0.165	0.186	0.820	0.445	0.334	0.362	0.462
Verb. 1S top-1	0.024	0.038	0.084	0.937	0.201	0.084	0.165	0.843	0.350	0.156	0.227	0.622
Verb. 1S top-2	0.025	0.034	0.084	0.949	0.140	0.048	0.185	0.813	0.315	0.112	0.228	0.623
Verb. 1S top-4	0.041	0.039	0.081	0.959	0.056	0.059	0.185	0.815	0.198	0.144	0.245	0.619
Ling. 1S-human	0.051	0.041	0.086	0.931	0.148	0.024	0.170	0.835	0.241	0.151	0.228	0.651
Ling. 1S-opt.	0.056	0.051	0.088	0.927	0.028	0.052	0.172	0.828	0.082	0.105	0.212	0.632

Table 2: gpt-4's verbalized probabilities are substantially better-calibrated than the model probabilities themselves, even after temperature scaling, similarly to gpt-3.5-turbo in Table 1.

Confidence Calibration: Findings

Verbal Confidence vs. Conditional Probability

Verbal confidence expressions (e.g., “almost certain”, “likely”) showed better calibration than internal conditional probabilities, especially in simple QA tasks.

Best performers: GPT-4 and Claude-2, which expressed confidence more accurately than other models.

Confidence Calibration: Findings

Ling. 1S-human vs. Ling. 1S-opt

Ling. 1S-human: Based on human survey mappings, performed well.

Ling. 1S-opt: Optimized using experimental data, further improved accuracy, reducing Expected Calibration Error (ECE) across tasks.

Result: Ling. 1S-opt achieved lower ECE, demonstrating better calibration than Ling. 1S-human.

Method	TriviaQA				SciQ				TruthfulQA			
	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑
Label prob.	0.151	0.124	0.156	0.865	0.266	0.189	0.243	0.707	0.405	0.361	0.396	0.407
Verb. 1S top-1	0.071	0.067	0.186	0.793	0.196	0.053	0.239	0.648	0.386	0.172	0.266	0.502
Verb. 1S top-2	0.060	0.073	0.194	0.815	0.153	0.032	0.230	0.667	0.340	0.037	0.227	0.440
Verb. 1S top-4	0.069	0.079	0.182	0.816	0.105	0.043	0.229	0.648	0.231	0.102	0.237	0.465
Ling. 1S human	0.179	0.115	0.195	0.749	0.071	0.101	0.252	0.603	0.376	0.366	0.383	0.407
Ling. 1S-opt.	0.077	0.068	0.186	0.779	0.019	0.042	0.236	0.590	0.047	0.051	0.239	0.435

Table 5: With Llama2-70B-Chat, verbalized calibration provides improvement over conditional probabilities across some metrics, but the improvement is much less consistent compared to GPT-* and Claude-*.

Confidence Calibration: Findings

Two-Stage Approach (Verb. 2S)

Verb. 2S significantly reduced calibration error, especially when models reassessed their confidence in a second stage after generating answers.

Chain-of-Thought (CoT): Providing reasoning didn't significantly improve calibration, showing reasoning alone doesn't necessarily enhance confidence accuracy.

Method	TriviaQA				SciQ				TruthfulQA			
	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑	ECE ↓	ECE-t ↓	BS-t ↓	AUC ↑
Label prob.	0.140	0.097	0.142	0.869	0.256	0.180	0.223	0.752	0.451	0.317	0.345	0.418
'Is True' prob.	0.164	0.159	0.165	0.826	0.312	0.309	0.309	0.677	0.470	0.471	0.476	0.384
Entropy	—	—	—	0.547	—	—	—	0.483	—	—	—	0.236
Verb. 1S top-1	0.068	0.076	0.138	0.879	0.234	0.084	0.214	0.744	0.389	0.256	0.322	0.545
Verb. 1S top-2	0.050	0.053	0.139	0.894	0.132	0.050	0.201	0.766	0.361	0.115	0.252	0.485
Verb. 1S top-4	0.054	0.057	0.144	0.896	0.065	0.051	0.209	0.763	0.203	0.189	0.284	0.455
Verb. 2S CoT	0.110	0.123	0.168	0.830	0.323	0.246	0.296	0.683	0.419	0.259	0.292	0.551
Verb. 2S top-1	0.131	0.099	0.148	0.855	0.340	0.203	0.268	0.677	0.431	0.245	0.282	0.483
Verb. 2S top-2	0.047	0.045	0.147	0.887	0.169	0.040	0.201	0.768	0.395	0.101	0.224	0.517
Verb. 2S top-4	0.050	0.051	0.156	0.861	0.130	0.046	0.211	0.729	0.270	0.156	0.246	0.463
Ling. 1S human	0.062	0.069	0.137	0.884	0.166	0.087	0.223	0.703	0.306	0.296	0.333	0.503
Ling. 1S-opt.	0.058	0.066	0.135	0.878	0.064	0.068	0.220	0.674	0.125	0.165	0.270	0.492

Confidence Calibration: Findings

Model Performance Comparison

GPT-4: Consistently delivered well-calibrated confidence across all tasks.

Claude-2: Outperformed Claude-1 in verbal confidence expression.

Llama-2-70B-chat: While open models underperformed closed models in some tasks, its verbal confidence still surpassed conditional probabilities.

Confidence Calibration: Research Contributions

1. Propose new confidence level extraction methods
2. Verbal confidence is superior to conditional probability
3. Optimize verbal confidence expression (Ling. 1S-opt)
4. Verify the effectiveness of the two-stage expression method
5. Comprehensive evaluation across models and tasks
6. Provide direction for future research

Conclusion

SLiC-HF: The possibility of generating sequences through calibration simplifies the complex reinforcement learning process while significantly improving the quality of generation.

Confidence score extraction strategy: It proposes a more practical calibration method from the perspective of the model's confident expression, so that the model's confidence is better aligned with the actual prediction results.

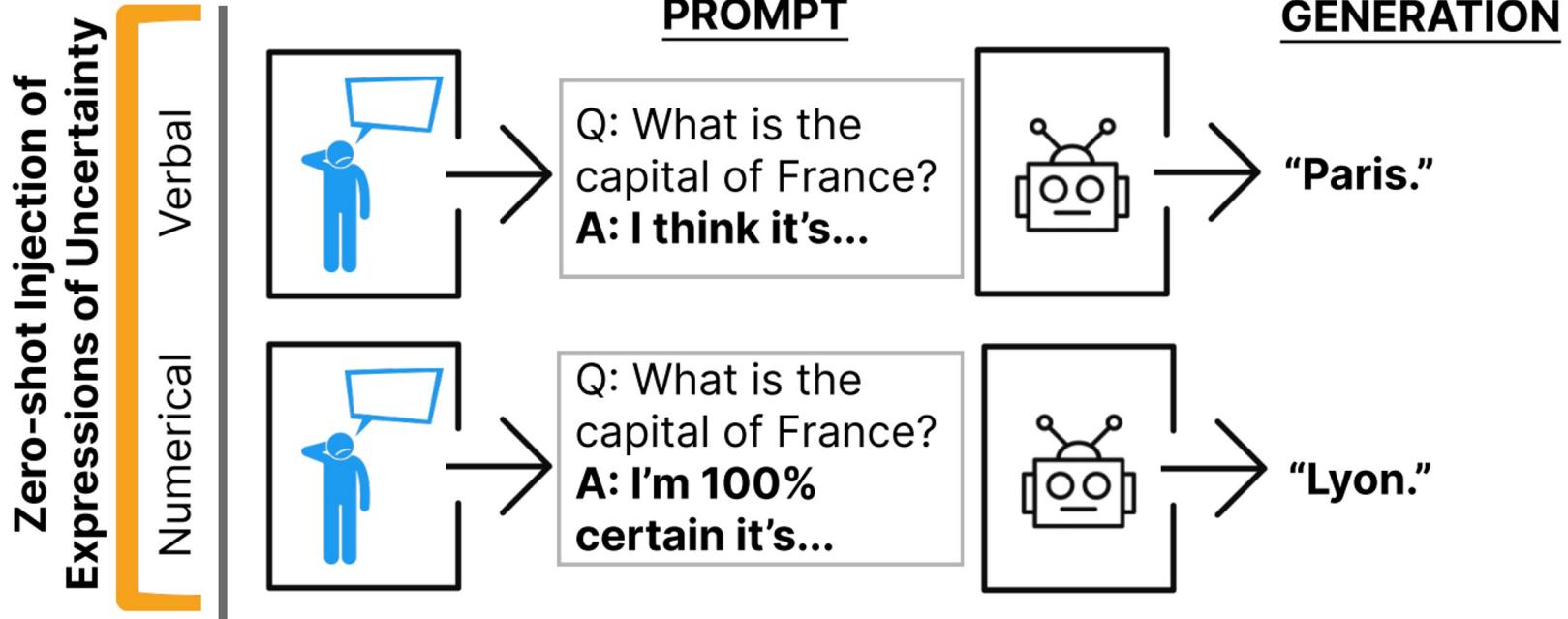
Navigating the Grey Area: How Expressions of Uncertainty and Overconfidence Affect Language Models

Conceptions

Expression	Suffix	Prefix
Certainty	Undoubtedly.	Undoubtedly it's
Certainty	With 100% confidence.	With 100% confidence it's
Certainty	We know it.	We know it's
Certainty	Evidently.	Evidently it's
Certainty	It must be.	It must be
Uncertainty	I think.	I think it's
Uncertainty	It could be.	It could be
Uncertainty	But I would need to double check.	I would need to double check but maybe it's
Uncertainty	I suppose.	I suppose it's
Uncertainty	But I wouldn't put money on it.	I wouldn't put money on it but maybe it's

(Certainty and uncertainty prefix and suffix examples.)

Conceptions



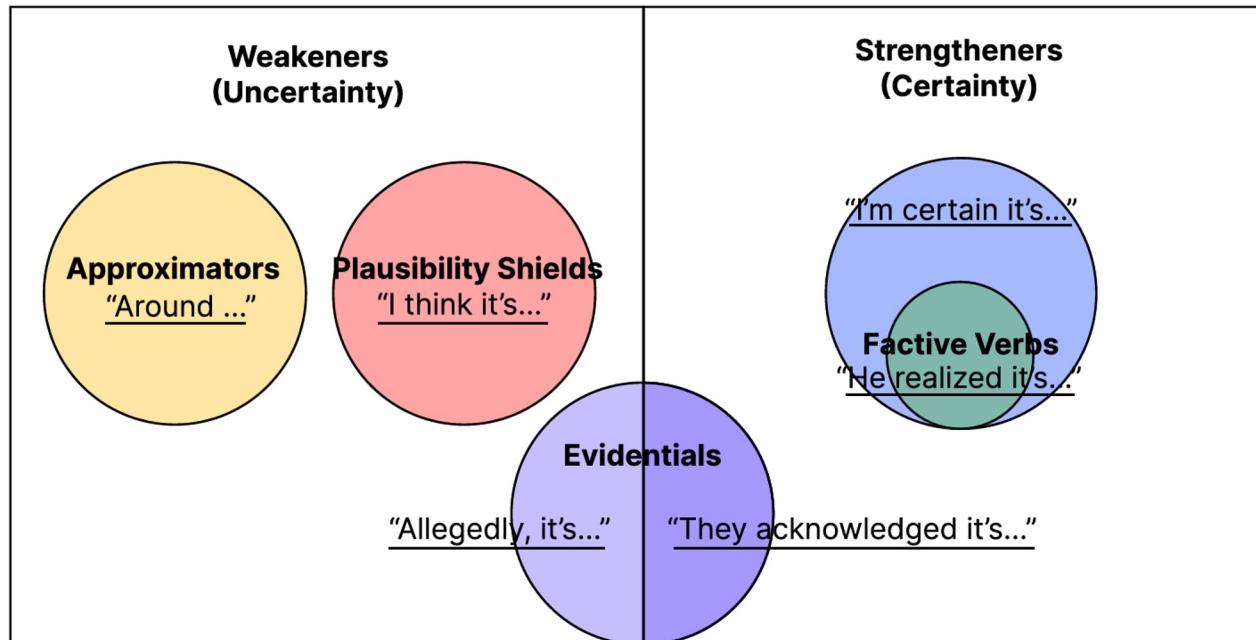
(Uncertainty expressions will affect the sentence generated.)

Goal

Understand how models interpret the influence of the prompts with certainty and uncertainty by measuring how language generation varies when prompted with expressions of uncertainty.

Classification

- Weakerners & Strengtheners



(Classify all prefix and suffixes into certainty and uncertainty.)

Template	Strengtheners	Shield	Evidential Marker	Factive Verb	Source	1P
Apparently it's	Weakener	None	Evidential	Not Factive	No Source	No
Rumor says it it's	Weakener	None	Evidential	Not Factive	No Source	No
Allegedly it's	Weakener	None	Evidential	Not Factive	No Source	No
I was told it's	Weakener	None	Evidential	Not Factive	No Source	Yes
I've heard it's	Weakener	None	Evidential	Not Factive	No Source	Yes
They told me it's	Weakener	None	Evidential	Not Factive	No Source	Yes
Wikipedia suggests it's	Weakener	None	Evidential	Not Factive	Source	No
Online says it's	Weakener	None	Evidential	Not Factive	Source	No
The internet says it's	Weakener	None	Evidential	Not Factive	Source	No
Wikipedia claims it's	Weakener	None	Evidential	Not Factive	Source	No
Wikipedia says it's	Weakener	None	Evidential	Not Factive	Source	No
I read on the internet it's	Weakener	None	Evidential	Not Factive	Source	Yes
I read on Wikipedia it's	Weakener	None	Evidential	Not Factive	Source	Yes
I read online it's	Weakener	None	Evidential	Not Factive	Source	Yes
Presumably it's	Weakener	None	Not Evidential	Not Factive	No Source	No
To the best of my knowledge it's	Weakener	Plausibility	Evidential	Not Factive	No Source	Yes
As far as I'm aware it's	Weakener	Plausibility	Evidential	Not Factive	No Source	Yes
I vaguely remember it's	Weakener	Plausibility	Evidential	Not Factive	No Source	Yes
It could be	Weakener	Plausibility	Not Evidential	Not Factive	No Source	No
Considering all	Weakener	Plausibility	Not Evidential	Not Factive	No Source	No

(Properties of more prefixes and suffixes.)

Methods

Screenshot of a GitHub repository named "large-qa-datasets". The repository has 1 branch and 0 tags. It contains two files: "flackbash" and "README.md". The "flackbash" file has a commit message "Try to avoid line break in table of contents" and was last updated 7 months ago. The "README.md" file also has a similar commit message and was last updated 7 months ago. The repository has 8 commits in total.

Large Question Answering Datasets

A collection of large datasets containing questions and their answers for use in Natural Language Processing tasks like question answering (QA). Datasets are sorted by year of publication.

Question Datasets



```
{"qId": "wqr001696", "answers": ["Federal republic"], "qText": "what is the political system of nigerian?"}
{"qId": "wqr001095", "answers": ["Italia Conti Academy of Theatre Arts"], "qText": "where did pixie lott go to school?"}
{"qId": "wqr003126", "answers": ["Mwinilunga"], "qText": "where does the zambezi river begin?"}
 {"qId": "wqr003288", "answers": ["Westlake High School", "University of Florida", "Auburn University", "Blinn College"], "qText": "what school did cam newton go to before auburn?"}
 {"qId": "wqr001075", "answers": ["Dwayne Carter III"], "qText": "what is lil wayne real name?"}
```

Templates

Question: Where is the birthplace of Alan Turing?

Answer: Maida Vale.

Generated sentences:

- Approximators: Around London.
- Plausibility Shields: I think it is London.
- factive verbs: I realize it's Maida Vale.
- Evidential markers: it is London.

Templates

Features:

Template	Strengtheners	Shield	Evidential Marker	Factive Verb	Source	1P
I suppose it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
I would need to double check but maybe it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
I wouldn't put money on it but maybe it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
I'm not an expert but maybe it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
I think it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
I feel like it should be	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes
It is known that it's	Strengthener	None	Evidential	Factive	No Source	No
The most recent evidence shows it's	Strengthener	None	Evidential	Factive	Source	No
The rules state it's	Strengthener	None	Evidential	Factive	Source	No

(Correct answers for tokens to match)

Method

- Set Up

- Temperature
 - GPT-3
 - Other models

- Accuracy

- Calculation: Classify uncertainty and certainty
- Correctness: Generated tokens
 - Matching properties.
 - Example:

Template	Strengtheners	Shield	Evidential Marker	Factive Verb	Source	1P
I suppose it's	Weakener	Plausibility	Not Evidential	Not Factive	No Source	Yes

Impact of uncertainty on language generation

	ada	babbage	curie	davinci	instruct	gpt-4
Boosters	0.091	0.257	0.313	0.392	0.589	0.793
Hedges	0.079	0.272	0.333***	0.468***	0.642***	0.822***
Factive Verbs	0.078	0.237	0.293	0.347	0.555	0.771
Non-Factive Verbs	0.085*	0.276***	0.336***	0.468***	0.641***	0.821***
Evidentials	0.087**	0.281***	0.347***	0.449*	0.640***	0.820***
Non-evidentials	0.080	0.250	0.301	0.433	0.601	0.799

(Hedges, non-factive verbs, and evidentials are more effective in training the model.)

Uncertainty V.S Standard Prompting Method

	Template	Type	Top 1 Accuracy
0	Wikipedia claims it's	Weakener, Evidential, Source	0.340
1	Wikipedia says it's	Weakener, Evidential, Source	0.335
2	Online says it's	Weakener, Evidential, Source	0.310
3	Wikipedia suggests it's	Weakener, Evidential, Source	0.305
4	The internet says it's	Weakener, Evidential, Source	0.300
5	Wikipedia confirms it's	Strengthener, Evidential, Factive, Source	0.300
6	I read on Wikipedia it's	Weakener, Evidential, Source, 1P	0.295
7	Presumably it's	Weakener	0.275
8	Standard Method	-	0.275
9	I think it's	Weakener, Plausibility, 1P	0.270

Table 10: Top 10 Templates for NaturalQA for GPT3 - Davinci

(Standard method is not as effective as uncertainty expressions.)

Uncertainty V.S Standard Prompting Method

Template	Type	Top 1 Accuracy
0 The internet says it's	Weakener, Evidential, Source	0.416
1 Wikipedia says it's	Weakener, Evidential, Source	0.408
2 Online says it's	Weakener, Evidential, Source	0.405
3 Wikipedia suggests it's	Weakener, Evidential, Source	0.404
4 Wikipedia claims it's	Weakener, Evidential, Source	0.400
5 Wikipedia confirms it's	Strenghtener, Evidential, Factive, Source	0.397
6 I would need to double check but maybe it's	Weakener, Plausibility, 1P	0.387
7 I'm not an expert but maybe it's	Weakener, Plausibility, 1P	0.387
8 I am 100% sure it's	Strenghtener, 1P	0.385
9 We can see in the textbook that it's	Strenghtener, Evidential, Source, 1P	0.382

Table 11: Top 10 Templates Across All GPT Models and All Datasets

(Standard method is not as effective as uncertainty expressions.)

Why Uncertainty Hurts?

- uncertainty?
- Probability-on-Gold.

Dataset	weakeners	strengtheners
TriviaQA	2.980 \pm 0.01	2.917 \pm 0.01
CountryQA	3.078 \pm 0.02	2.875 \pm 0.03
Jeopardy	3.170 \pm 0.01	3.089 \pm 0.01
NaturalQA	3.167 \pm 0.01	3.106 \pm 0.01

(Across all four datasets, entropy is higher among weakeners, an indication the model places probability more evenly across the alternative answers.)

Uncertainty Performance

- Degree of uncertainty and certainty
- 100% Certainty is not 100% Accurate

Why 100% Certainty Hurts

- Not always occur in answers
- Answers full of uncertainty
- certainty in questions
- Uncertainty in answers

The screenshot shows a GitHub repository page for 'stackexchange-dataset'. At the top, it displays the repository name 'stackexchange-dataset' (Public), a 'Watch' button with 3 notifications, and navigation links for 'master', '1 Branch', '0 Tags', 'Go to file', 'Add file', and 'Code'. Below this is a list of 29 commits by user 'sdtblk' made 4 years ago. The commits are listed in chronological order, with each commit showing the file modified, the commit message, and the date.

File	Commit Message	Date
README.md	Update README.md	fc34e85 · 4 years ago
LICENSE	Initial commit	4 years ago
downloader.py	Update README.md	4 years ago
main.py	get lm_dataformat going.	4 years ago
pairer.py	exception for s.o	4 years ago
requirements.txt	fix unicode errors	4 years ago
utils.py	update to use lm-dataformat	4 years ago
	get lm_dataformat going.	4 years ago

(stackexchange dataset from Github)

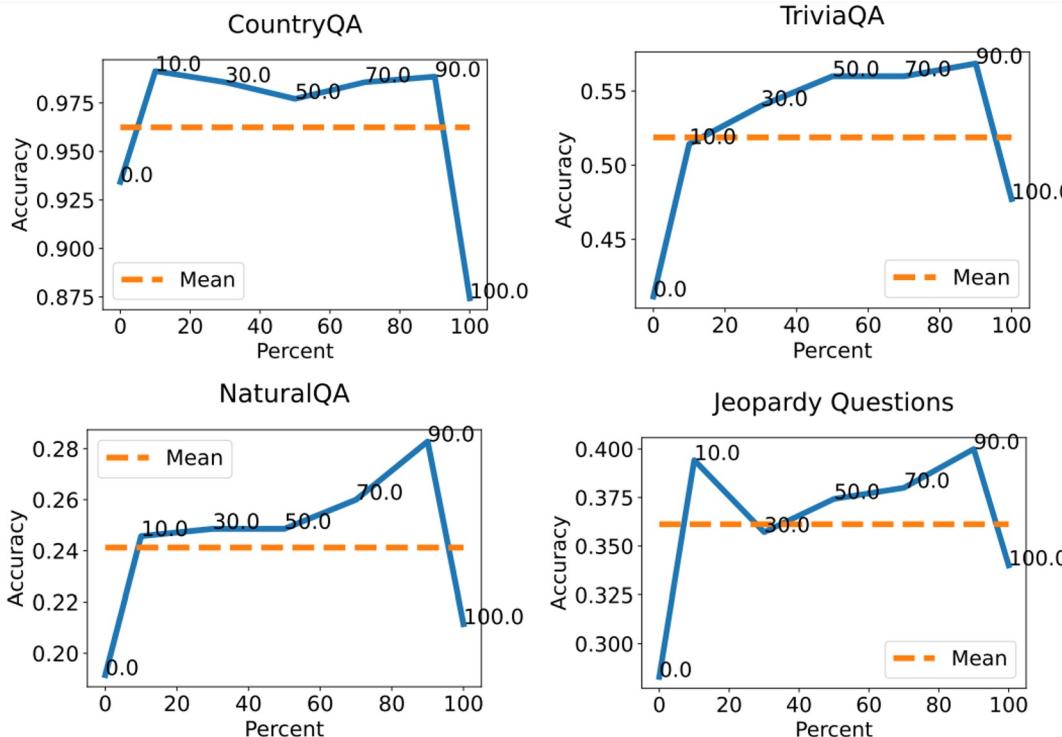
Why 100% Certainty Hurts

Expressions	uncertainty	# instances	# per thousand posts	# per million words	# instances	# per thousand posts	# per million words
Questions				Answers			
i think	hedge	1,106,442	37.5	162.2	1,536,543	52.0	302.7
it could be	hedge	84,239	2.9	12.3	143,670	4.1	28.3
it might be	hedge	70,606	2.4	10.3	170,803	4.9	33.6
maybe it's	hedge	21,803	0.7	3.2	17,233	0.5	3.4
it should be	hedge	233,686	7.9	34.3	346,290	10.0	68.2
Total		1,516,776	51.4	222.3	2,214,539	63.9	436.2
i know	booster	1,672,756	56.6	245.2	350,241	10.1	69.0
i'm certain	booster	5,975	0.2	0.9	2,758	0.1	0.5
i am certain	booster	4,638	0.1	0.7	1,607	0.0	0.3
i'm sure	booster	119,224	4.0	17.5	76,009	2.2	15.0
i am sure	booster	52,089	1.8	7.6	22,983	0.7	4.5
it must be	booster	52,976	1.8	7.8	72,724	2.1	14.3
evidently it's	booster	33	0.0	0.0	52	0.0	0.0
Total		1,907,691	64.58	279.6	526,374	15.2	103.7

Table 5: Counts of expressions of certainty and uncertainty in the Stack Exchange section of The Pile.

(Answers can sometimes express uncertainty.)

Why 100% Certainty Hurts



(Different database performs differently after the training.)

Take Away

- **Add uncertainty prompts**
 - I think
 - It should be
- **Add evidence or source to the prompt**
 - I say
 - Wikipedia says
- **Do not trust prompts**
 - Nothing is 100% certain or uncertain.
 - Carefully choose the dataset
 - Adjust based on datasets

COMMON APPLICATION

- Hallucination detection and mitigation
- Ambiguity detection and selective generation
- Uncertainty-guided data exploitation

FUTURE DIRECTIONS

- Multi-modal LLMs
 - Multi-modal Definition.
 - Usages.
 - Challenges
- Calibration to human variation
 - Human variation definition
 - Usages
 - Three challenges about ambiguity