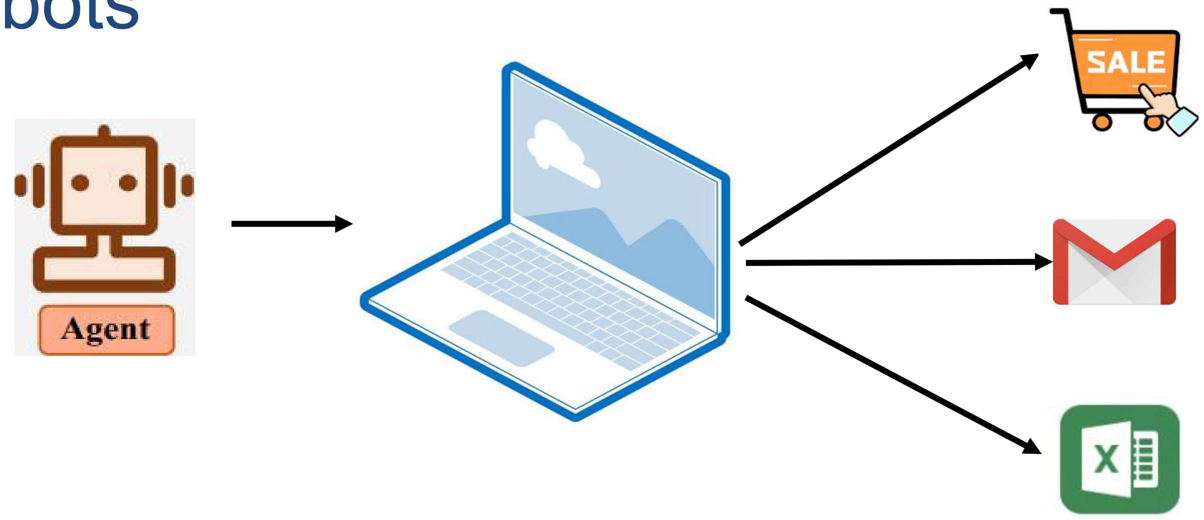
ConfTuner
Training Large Language Models to Express Their Confidence Verbally

Yibo Li, Miao Xiong, Jiaying Wu, Bryan Hooi
National University of Singapore

Trustworthy Is Important for LLMs

- LLMs are increasingly incorporated into our lives in a deeper way beyond chatbots

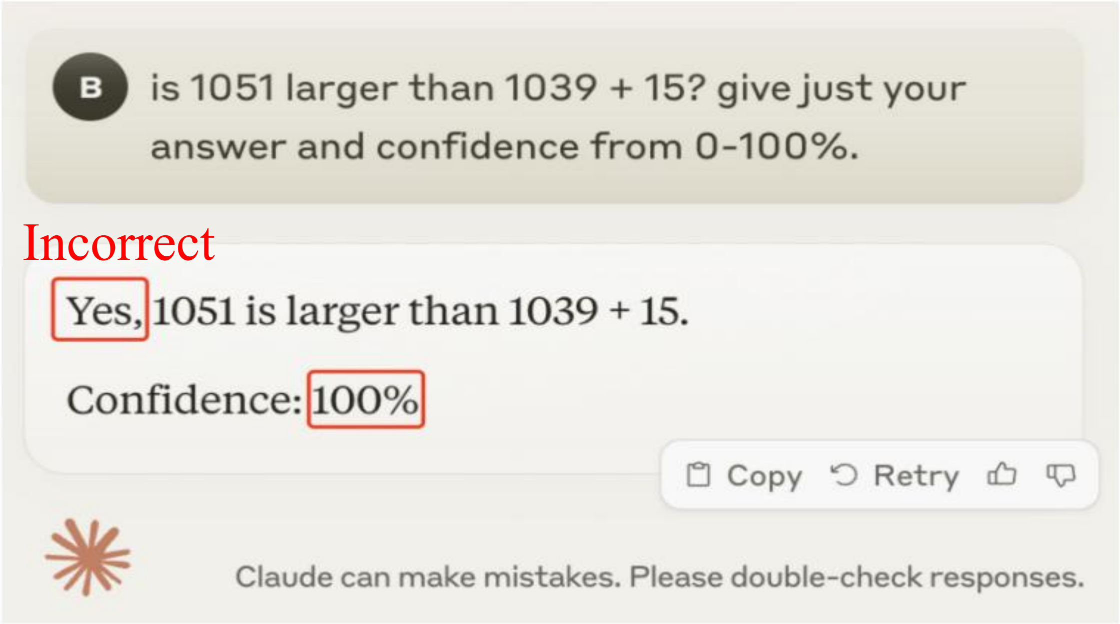


Computer-Using Agent

- It is crucial to develop general approaches for human and LLM to work together effectively

Overconfidence of LLMs

- A fundamental weakness of LLMs is they are all overconfident

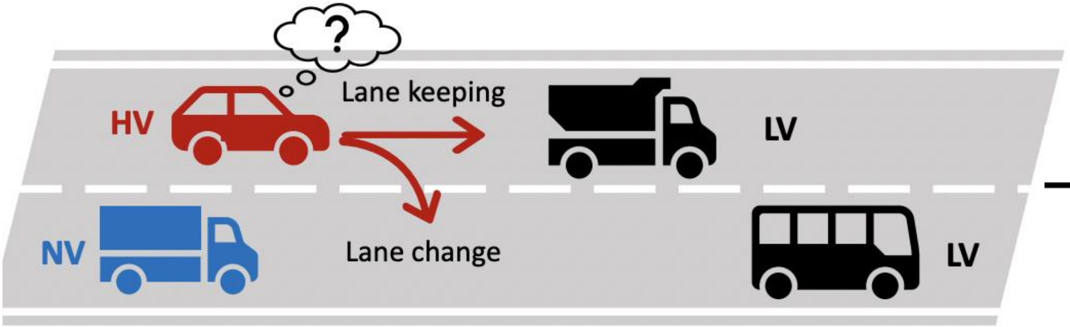


Overconfidence of LLM

Trustworthy LLM Emable rational decisions

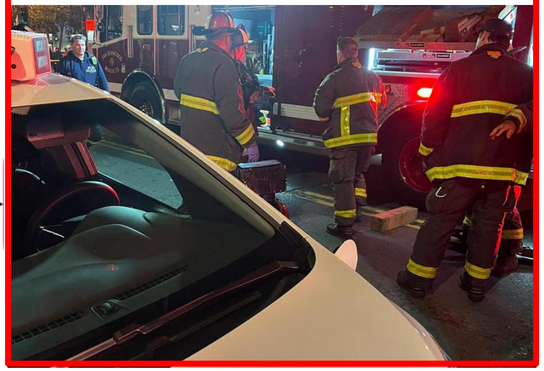
- Developing models that can accurately express their confidence is important, particularly in **Human-Computer Interaction**, as it enables humans to make more rational and informed decisions.

Medical Treatment



A: Lane Change (100%)

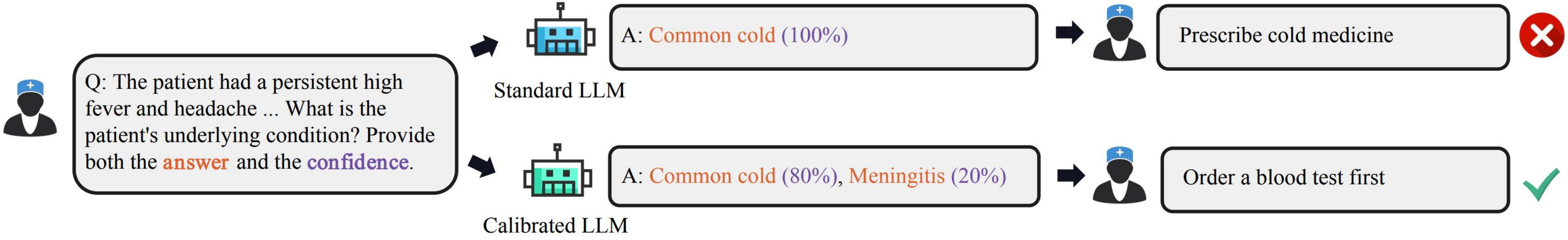
San Francisco self-driving car involved in serious accident



Autonomous driving

Calibrate Verbalized Confidence of LLMs

- We want to calibrate LLM to expresses appropriate uncertainty.



How to Calibrate Verbalized Confidence?

Is 1051 larger than 1039+15? → LLM —response 10 times→

Yes
Yes
Yes
No
No
No
No
No
Yes
No

Accuracy: 0.6

—train LLM to output→ Answer: No. Confidence: 60%

Increase both computational costs and random noise

Can LLMs be naturally calibrated without ground-truth confidence?

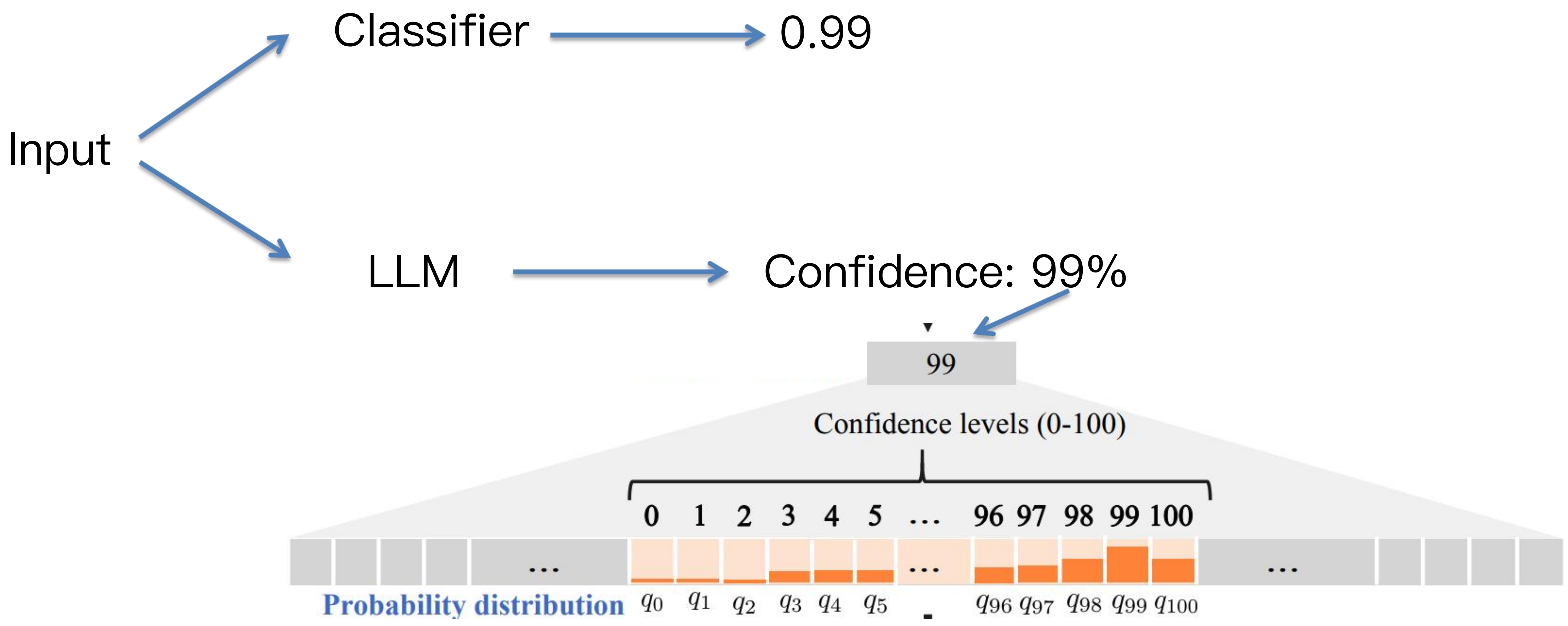
- Classical machine learning classifiers naturally become well-calibrated during training when optimized with loss functions that are proper scoring rules. such as Brier score.

incentivize the classifier to output probabilities that reflect the model's true likelihood of correctness

- Can we use proper scoring rules to calibrate LLMs?

Proper Scoring Rule on LLMs

- We cannot directly use proper scoring rules to LLMs.

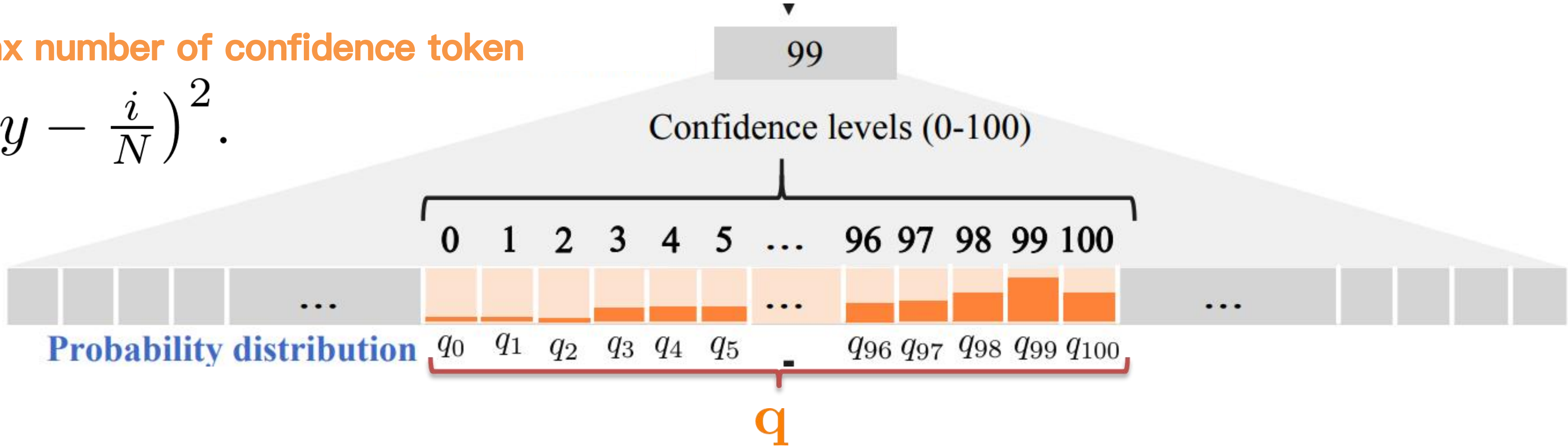


Tokenized Brier Score

correct indicator; N → max number of confidence token

$$\ell(\mathbf{q}, y) \ := \sum_{i=0}^{N} q_i \left(y - \tfrac{i}{N}\right)^2.$$

vector of logits 0–100



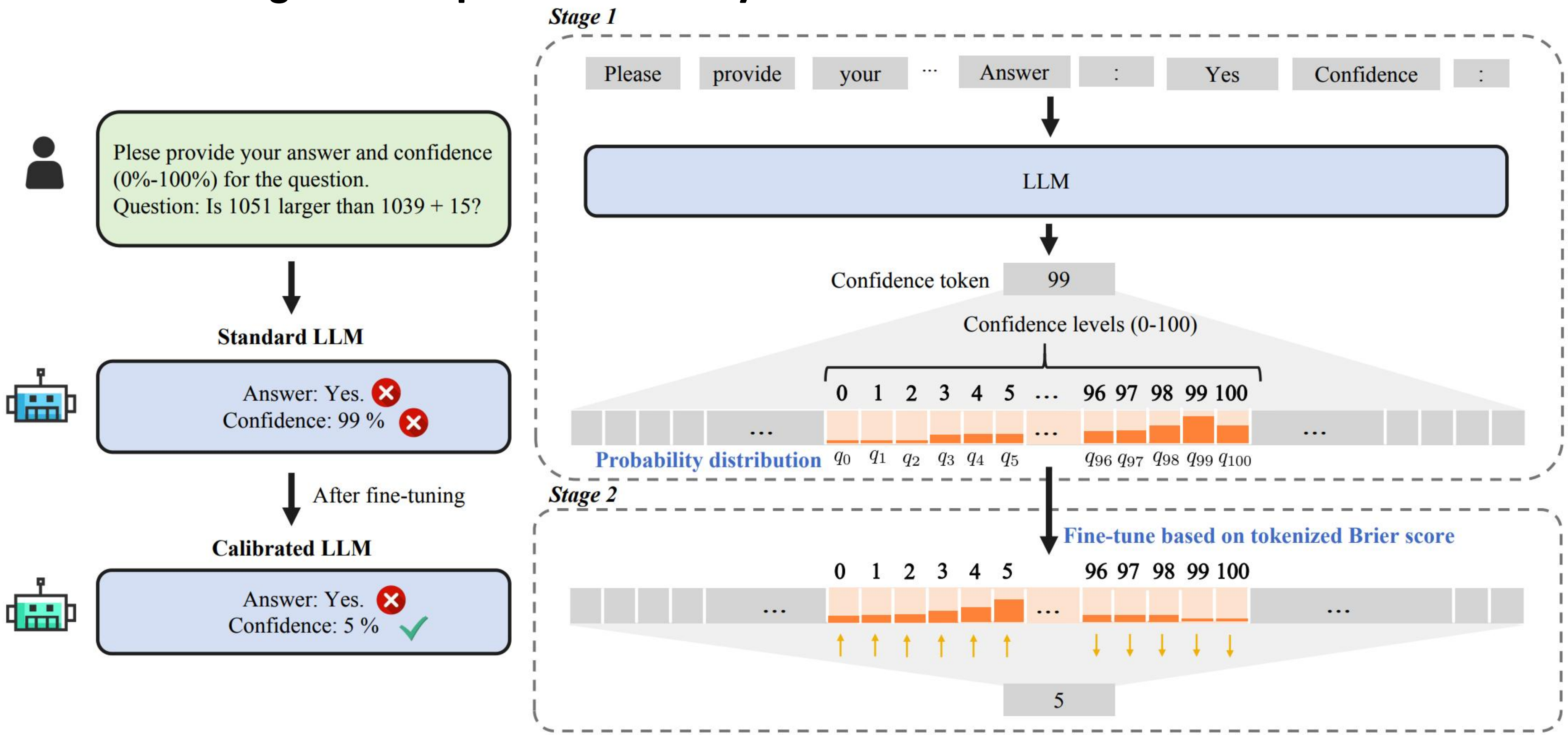
- Encourages model to express larger confidence levels when the answer is correct, and vice versa.
  - e.g. when N=100, y=1:

$$\ell(\mathbf{q}, 1) \ := \ q_0 \underbrace{\left(1 - \tfrac{0}{100}\right)^2}_{1} + q_1\left(1 - \tfrac{1}{100}\right)^2 + \cdots + q_{99}\left(1 - \tfrac{99}{100}\right)^2 + q_{100}\underbrace{\left(1 - \tfrac{100}{100}\right)^2}_{0}$$

We theoretically prove it to be a proper scoring rule.

ConfTuner

Stage 1: Compute Probability Distribution Over Confidence Tokens



Stage 2: Fine-Tune Based on Tokenized Brier Score

Experiments

- Datasets: HotpotQA, TriviaQA, StrategyQA, GSM8K, TruthfulQA
- Base LLMs: LLaMA, Qwen, Ministral
- Evaluation Metrics: AUROC , ECE

Can ConfTuner Learn Effective Verbalized Confidence Estimation Capabilities?

- Generalization to unseen datasets.

AUROC ↑

		In-distribution	Out-of-distribution				
LLM	Method	HotpotQA	GSM8K	TriviaQA	StrategyQA	TruthfulQA	Average
LLaMA	Base	0.6884	0.5028	0.6023	0.6249	0.5433	0.5923
	Ensemble	0.6035	0.5210	0.6323	0.6022	**0.6038**	0.5926
	LACIE	0.7233	0.5117	0.6818	0.6525	0.5452	0.6229
	SaySelf	0.6596	0.5425	0.6202	0.5493	0.5890	0.5921
	ConfTuner	**0.7383**	**0.7007**	**0.6821**	**0.6750**	0.5739	**0.6740**
Qwen	Base	0.6863	0.5114	0.6224	0.6059	0.6517	0.6155
	Ensemble	0.6259	0.5683	0.6287	0.5959	0.6460	0.6130
	LACIE	0.7141	0.5473	0.6951	0.6312	0.6397	0.6455
	SaySelf	0.6972	0.5247	0.6133	0.6265	0.6312	0.6186
	ConfTuner	**0.7180**	**0.5841**	**0.7664**	**0.6692**	**0.6926**	**0.6861**
Ministral	Base	0.5198	0.5133	0.5078	0.5129	0.5541	0.5216
	Ensemble	0.5679	0.6696	0.5004	**0.6222**	0.6153	0.5951
	LACIE	0.6505	0.5126	0.5128	0.6134	0.6098	0.5798
	SaySelf	0.6482	0.5133	0.5477	0.5555	0.6060	0.5740
	ConfTuner	**0.7907**	**0.6700**	**0.7389**	0.5147	**0.6906**	**0.6810**

ECE↓

		In-distribution	Out-of-distribution				
LLM	Method	HotpotQA	GSM8K	TriviaQA	StrategyQA	TruthfulQA	Average
LLaMA	Base	0.4803	0.1896	0.1904	0.1469	0.3770	0.2768
	Ensemble	0.4254	0.2365	0.1652	0.1474	0.4035	0.2756
	LACIE	0.2954	0.1613	0.1396	0.1577	0.4394	0.2387
	SaySelf	0.3358	0.2217	0.2185	0.1453	0.3245	0.2492
	ConfTuner	**0.0405**	**0.1276**	**0.0388**	**0.1387**	**0.1955**	**0.1082**
Qwen	Base	0.6312	0.1306	0.4302	0.2199	0.4786	0.3781
	Ensemble	0.5909	0.2428	0.3595	**0.1226**	0.4626	0.3597
	LACIE	0.5519	**0.1240**	0.4060	0.1775	0.4422	0.3403
	SaySelf	0.5401	0.1244	0.4024	0.1883	0.4509	0.3412
	ConfTuner	**0.4212**	0.1302	**0.3549**	0.1815	**0.3484**	**0.2872**
Ministral	Base	0.6767	0.2926	0.3715	0.2813	0.5746	0.4393
	Ensemble	0.5887	0.3357	0.3966	0.1948	0.5670	0.4166
	LACIE	0.5627	0.2745	0.2503	0.3321	0.4221	0.3683
	SaySelf	0.5536	0.2893	0.3668	0.2784	0.5438	0.4064
	ConfTuner	**0.1027**	**0.2128**	**0.1736**	**0.1815**	**0.2715**	**0.1884**

Can ConfTuner Learn Effective Verbalized Confidence Estimation Capabilities?

- Generalization to different format of confidence scores.
  - Specifically, we test ConfTuner's performance on **high/medium/low** confidence levels.

AUROC

LLM	Method	In-distribution	Out-of-distribution				
		HotpotQA	GSM8K	TriviaQA	StrategyQA	TruthfulQA	Average
LLaMA	Base	0.5859	0.5541	0.5564	0.6280	0.5345	0.5718
	LACIE	0.6013	0.3940	0.5337	0.5105	0.5236	0.5126
	SaySelf	0.6497	0.5841	0.5775	0.6379	0.5453	0.5989
	ConfTuner	**0.7203**	**0.6524**	**0.6820**	**0.6494**	**0.5515**	**0.6511**
Qwen	Base	0.5664	0.5257	0.5204	0.5959	0.5517	0.5520
	LACIE	0.5052	0.4758	0.5442	0.6059	0.5167	0.5296
	SaySelf	0.5814	0.5342	0.5423	0.6148	0.5618	0.5669
	ConfTuner	**0.7116**	**0.6050**	**0.5957**	**0.6385**	**0.5926**	**0.6287**
Ministral	Base	0.5167	0.5181	0.5055	0.5346	0.5177	0.5185
	LACIE	0.5239	0.5535	0.5136	0.5190	0.5620	0.5344
	SaySelf	0.5449	0.5536	0.5427	**0.5370**	0.5478	0.5452
	ConfTuner	**0.7520**	**0.7018**	**0.7517**	0.5000	**0.6123**	**0.6636**

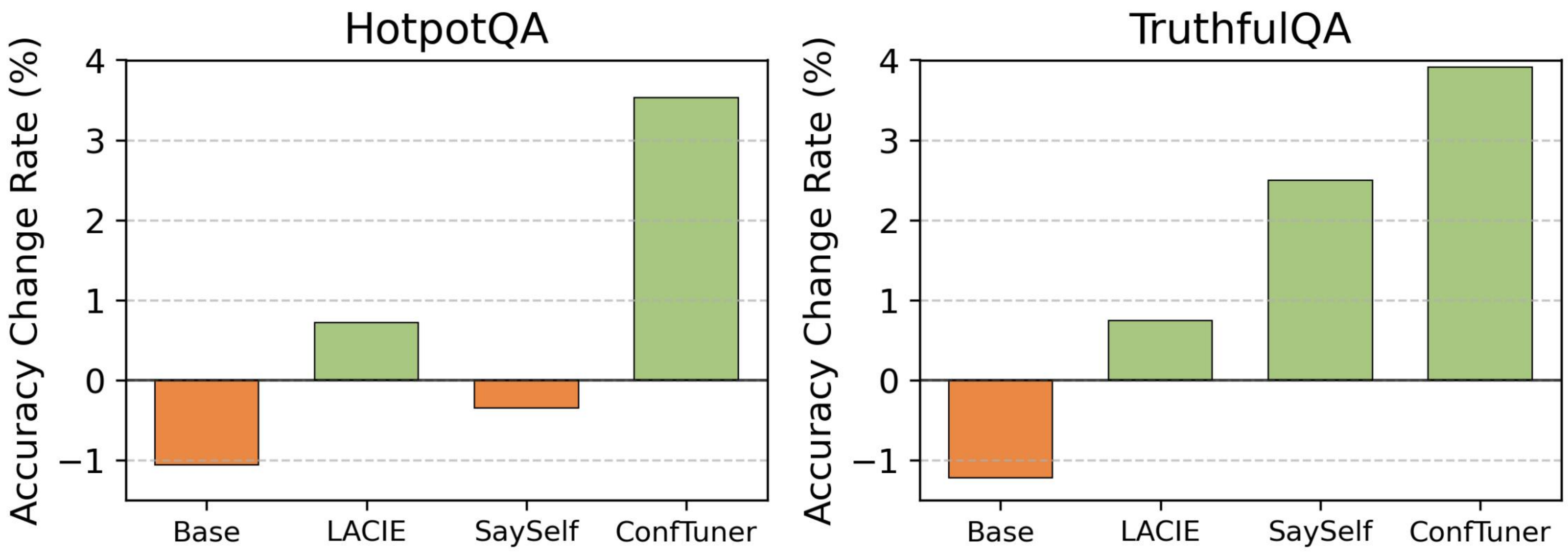
Can ConfTuner Learn Effective Verbalized Confidence Estimation Capabilities?

- Generalization to implicit confidence expressions.
  - e.g. I'm fairly certain, but there's a chance I could be mistaken

		In-distribution		Out-of-distribution			
Metric	Method	HotpotQA	GSM8K	TriviaQA	StrategyQA	TruthfulQA	Average
ECE ↓	Base (i)	0.2808	0.1179	0.1232	**0.1098**	0.3250	0.1913
	ConfTuner (e)	**0.0405**	0.1276	**0.0388**	0.1387	**0.1955**	**0.1082**
	ConfTuner (i)	0.1639	**0.0950**	0.1088	0.1721	0.2019	0.1483
AUROC ↑	Base (i)	0.7047	0.5422	0.6342	0.6489	0.5895	0.6239
	ConfTuner (e)	**0.7383**	**0.7007**	0.6821	**0.6750**	0.5739	0.6740
	ConfTuner (i)	0.7239	0.6869	**0.7024**	0.6751	**0.6217**	**0.6820**

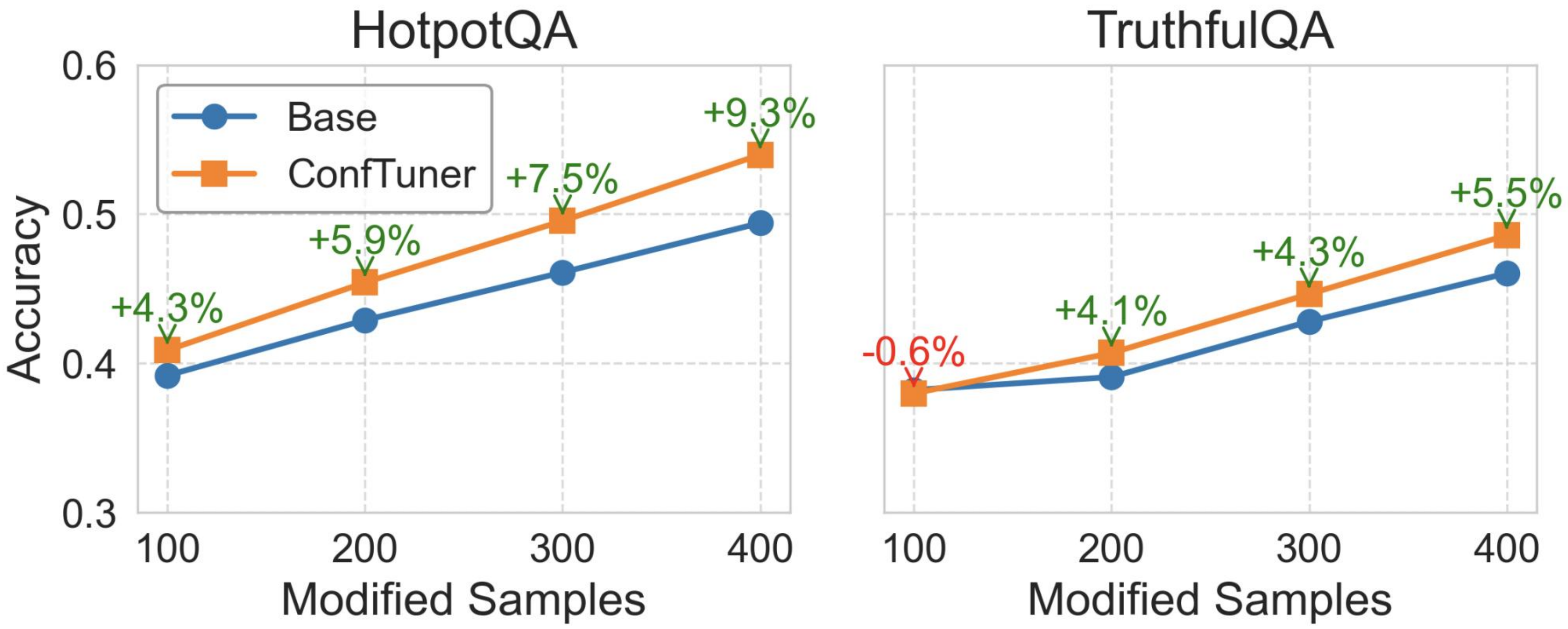
Can ConfTuner Help Build More Reliable and Cost-Effective LLM Systems?

- ConfTuner improves the self-correction ability of LLM
  - We first instruct LLM to generate answers and confidences, then retain initial responses with high confident answers, and instruct LLM to refine low-confident answers



Can ConfTuner Help Build More Reliable and Cost-Effective LLM Systems?

- ConfTuner achieves higher performance gain at same cost in confidence-based model cascade systems
  - When base models are uncertain, use a strong model to generate the answer.



Running Time and Training Dataset Size

Method	Time		Training Data		
	Training	Inference	Data size	Sample times	Total number
LACIE	26 min	1 min	10,000	10	100,000
SaySelf	120 min	1 min	90,000	100	9,000,000
Ensemble	-	10 min	-	-	-
ConfTuner	**4 min**	**1 min**	**2,000**	**1**	**2,000**

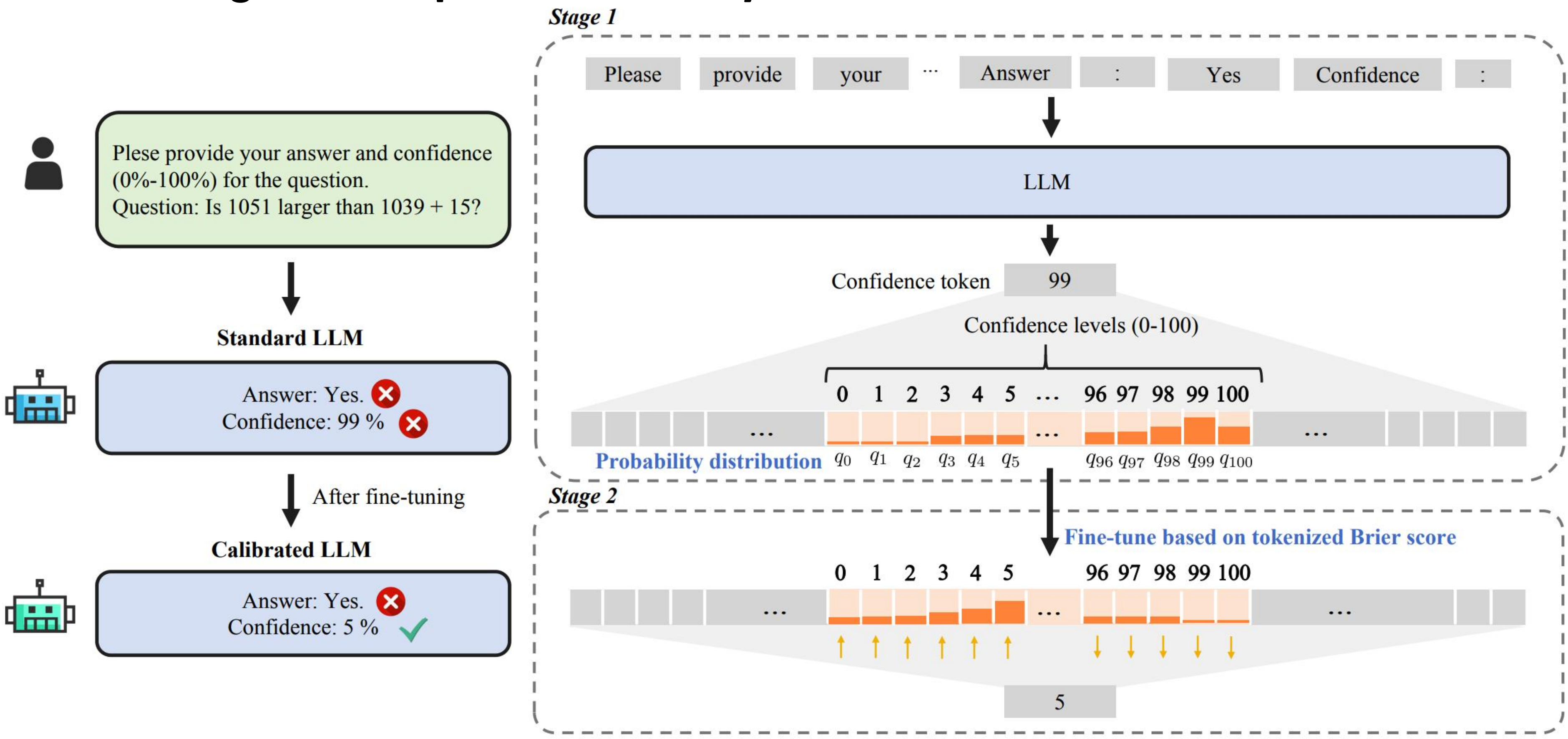
ConfTuner: Training Large Language Models to Express Their Confidence Verbally

Yibo Li, Miao Xiong, Jiaying Wu, Bryan Hooi

Tokenized Brier score

$$\ell(\mathbf{q}, y) \ := \ \sum_{i=0}^{N} q_i \left(y - \frac{i}{N}\right)^2.$$

Stage 1: Compute Probability Distribution Over Confidence Tokens



Stage 2: Fine-Tune Based on Tokenized Brier Score