AIAA 5047
Responsible AI
2025 Fall

Sihong Xie, AI Thrust, Information Hub

Lecture 9

W2 201, 9-11:50 AM F

Introduction

What is uncertainty in classification?

In classification, the value of predicted probability is value of the class with maximum probability:

$$\hat{p}(\mathbf{x}) = \max_{c}(\mathbf{y} = c|\mathbf{x})$$

It leads to Maximum Probability uncertainty measure:

$$U_{MP} = 1 - \hat{p}(\mathbf{x}) = 1 - \max_{c}(\mathbf{y} = c|\mathbf{x})$$

Introduction

What if applying the classification UQ to language models?

The vast majority of modern language models generate text in the autoregressive way:

$$y_l \sim P(y_l \mid \mathbf{y}_{<l},\ \mathbf{x},\ \theta), l = 1, \ldots, L,$$

where $\mathbf{x}$ is a prompt, $\mathbf{y}_{<l} = [y_1, \ldots, y_{l-1}]$, θ is the parameter of the language model.

The resulting probability of the generated sequence $\mathbf{y}$:

$$P(\mathbf{y} \mid \mathbf{x},\ \theta) = \prod_{l=1}^{L} P(y_l \mid \mathbf{y}_{<l},\ \mathbf{x},\ \theta)$$

Introduction

Challenges of uncertainty quantification in language models

- **How to define y_l?**

  subword level: [We] [learn] [un] [certain] [ty] [quantifi] [cation][.]

  Word level: [We] [learn] [uncertainty] [quantification][.]

  Phrase level: [We] [learn] [uncertainty quantification][.]

- **How about semantically equivalent sentences?**

  We learn uncertainty quantification. = We study uncertainty quantification.

- **Length bias: longer outputs tend to be more uncertain as**

$$P(y_l \mid \mathbf{y}_{<l},\ \mathbf{x},\ \theta) \leq 1.$$

- **Not all words are equally important in an output.**

Introduction

Types of unsupervised UQ methods

Black-box methods

- Verbalized uncertainty
  Directly asking the model about its confidence in a generated answer
- Consistency-based
  Sample multiple generations and measure their (semantic) consistency

White-box methods

- Information-theoretic
  Assess uncertainty as measured by probabilities given by the model
- Introspective
  Analyze model embeddings and/or attention masks

Artem Shelmanov, Maxim Panov, Roman Vashurin, Artem Vazhentsev, Ekaterina Fadeeva, and Timothy Baldwin. 2025. Uncertainty Quantification for Large Language Models. In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 5: Tutorial Abstracts), pages 3–4, Vienna, Austria. Association for Computational Linguistics.

Information-theoretic confidence

Sequence Probability (log-probability) is the most straightforward measure of confidence for a response $\mathbf{y}^*$:

$$C_{\mathrm{SP}}(\mathbf{y}^*, \mathbf{x}) = \log P(\mathbf{y}^* | \mathbf{x}) = \sum_{l=1}^{L} \log P(y_l^* \mid \mathbf{y}_{<l}^*, \ \mathbf{x})$$

Due to autoregressive probabilistic model, it has a natural bias towards shorter sequences.

Perplexity (length-normalized log-probability) mitigates this bias:

$$C_{\mathrm{PPL}}(\mathbf{y}^*, \mathbf{x}) = \log \bar{P}(\mathbf{y}^* | \mathbf{x}) = \frac{1}{L} \sum_{l=1}^{L} \log P(y_l^* \mid \mathbf{y}_{<l}^*, \ \mathbf{x})$$

Information-theoretic confidence

Not all tokens contribute to uncertainty equally

Problem: Natural language contains different parts, and LLMs tend to augment answers with preambles.

"The answer is: Donald Trump is the current president of the United States."

There is no much information in "The answer is", which serve functional role rather than delivering contents.

Solution: Put weights on the tokens according to their contribution to the overall meaning of the output:

"The answer is: Donald Trump is the current president of the United States."

Information-theoretic confidence

Confidence with token emphasis (TokenSAR)

Self Natural Language Inference (self-NLI) score is a metric used to evaluate the consistency of two texts, denoted by

$$\mathrm{NLI}(a, b),$$

where a is premise and b is hypothesis. NLI predicts one of the three standard labels:

$$\{\text{entailment ('e')}, \text{neutral ('n')}, \text{contradiction ('c')}\}.$$

Token weights can be defined as the self-NLI score when token is removed:

$$R\left(y_j^*, \mathbf{y}^*, \mathbf{x}\right) = 1 - P\left(\mathrm{NLI}\left(\mathbf{x} \cup \mathbf{y}^*, \mathbf{x} \cup \mathbf{y}^* \backslash y_j^*\right) = \text{'}e\text{'}\right).$$ (large if prob of entailment is low)

Use them as weights and aggregate token log-probabilities

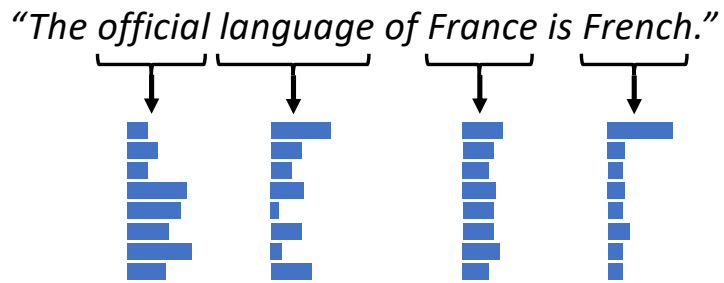
$$C_{\mathrm{SAR}}(\mathbf{y}^*, \mathbf{x}) = \sum_{j=1}^{L} \left[\frac{R\left(y_j^*, \mathbf{y}^*, \mathbf{x}\right)}{\sum_{l=1}^{L} R\left(y_l^*, \mathbf{y}^*, \mathbf{x}\right)}\right] \log P\left(y_j^* \mid \mathbf{y}_{<j}^*, \mathbf{x}\right)$$

Jinhao Duan et al. (2024). "Shifting Attention to Relevance: Towards the Predictive Uncertainty Quantification of Free-Form Large Language Models". In: ACL 2024.

Information-theoretic confidence

Mean token entropy

In the white-box setting, full distributions over vocabulary space $\mathcal{V}$ are available at each token position.



At each position we can compute entropy of these distributions, and their average gives Mean Token Entropy (MTE):

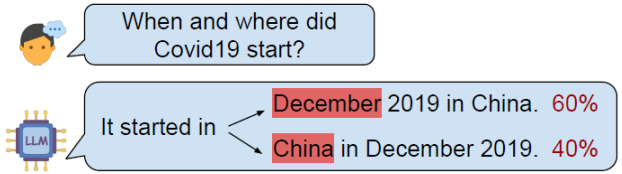
$$C_{\mathrm{MTE}}(\mathbf{y}^*, \mathbf{x}) = \frac{1}{L} \sum_{j=1}^{L} \sum_{t \in \mathcal{V}} P\left(y_j^* = t \middle| \mathbf{y}_{<j}^*, \mathbf{x}\right) \log P\left(y_j^* = t \middle| \mathbf{y}_{<j}^*, \mathbf{x}\right)$$

Information-theoretic confidence

High uncertainty does not always come from hallucination

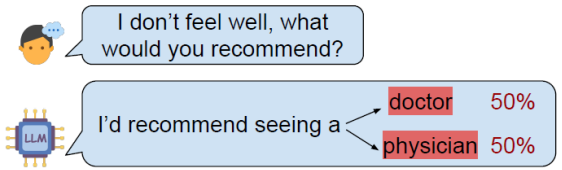
Probability-based methods do not account for different types of uncertainties. $P\left(y_j^* \mid \mathbf{y}_{<j}^*, \mathbf{x}\right)$ can be low because of:

- phrase-order uncertainty (expressed through varied order while conveying the same claim).



Both are correct but one is less confident.

- wording uncertainty



Same meaning but one is less confident.

- true uncertainty (which claim to present)

Both options are diary product and none is hallucination.

Information-theoretic confidence

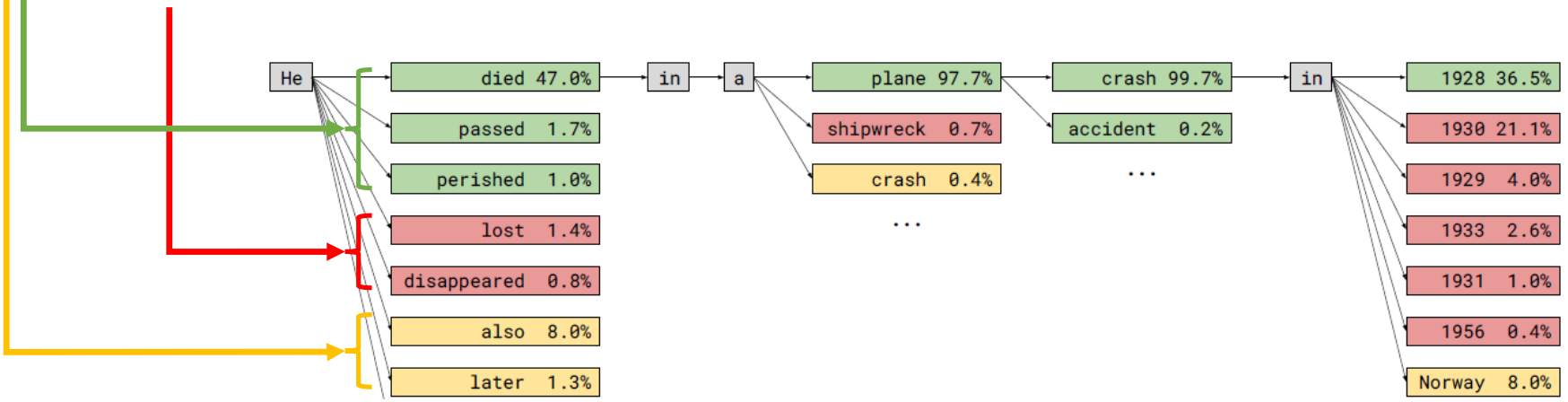
Classification of alternative tokens

Each non-functional token is supplied with top alternative tokens in the generation.

Neutral words reflect claim phrase-order uncertainty.

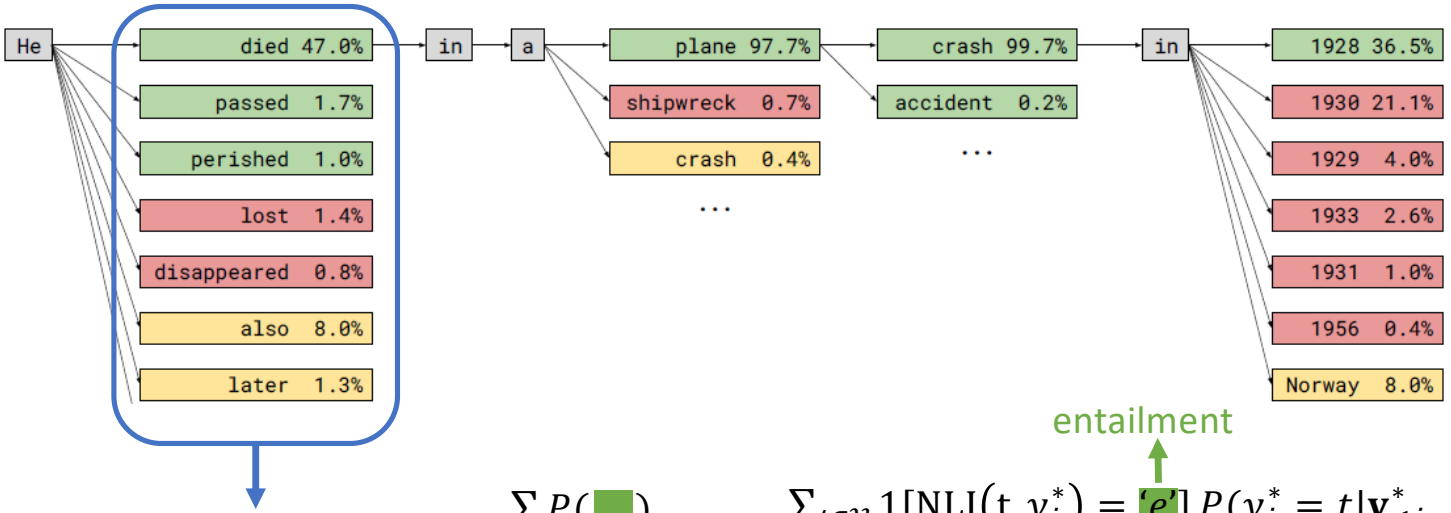
Synonyms/hypernyms reflect wording uncertainty.

Words with different semantic meaning reflect true uncertainty.



Information-theoretic confidence

Claim-Conditioned Probability: CCP



entailment

$$\mathrm{CCP}(y_j^*, \mathbf{y}_{<j}^*, \mathbf{x}) = \frac{\sum P(\blacksquare)}{\sum P(\blacksquare + \blacksquare)} = \frac{\sum_{t \in \mathcal{V}} 1[\mathrm{NLI}(t, y_j^*) = \text{'e'}]\, P(y_j^* = t | \mathbf{y}_{<j}^*, \mathbf{x})}{\sum_{t \in \mathcal{V}} 1[\mathrm{NLI}(t, y_j^*) \in \{\text{'e'}, \text{'c'}\}]\, P(y_j^* = t | \mathbf{y}_{<j}^*, \mathbf{x})}$$

contradiction

Token-level CCP scores are aggregated to sequence-level confidence:

$$\mathrm{CCP}(\mathbf{y}^*, \mathbf{x}) = \prod_{j=1}^{L} \mathrm{CCP}(y_j^*, \mathbf{y}_{<j}^*, \mathbf{x})$$

Ekaterina Fadeeva et al. (2024). “Fact-Checking the Output of Large Language Models via Token-Level Uncertainty Quantification”. In: Findings of ACL 2024.

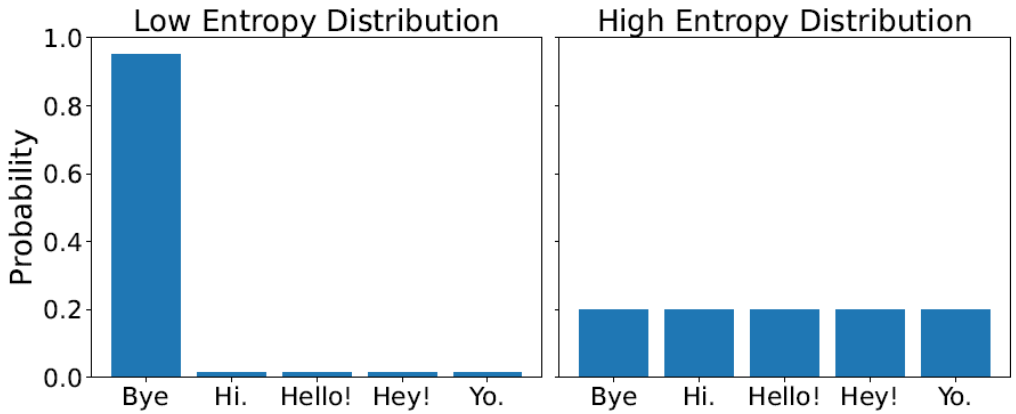
Information-theoretic uncertainty

Sequence Entropy

LLM induces a distribution $P(\mathbf{y} \mid \mathbf{x})$ over all possible outputs $\mathbf{y} \in \mathcal{Y}$.

When predictive distribution is present, entropy is a natural uncertainty measure:

$$H(\mathbf{x}) = -\sum_{\mathbf{y} \in \mathcal{Y}} P(\mathbf{y} \mid \mathbf{x}) \log P(\mathbf{y} \mid \mathbf{x})$$



However, direct computation of entropy is almost intractable due to the size of $\mathcal{Y}$ (vocabulary size).

Information-theoretic uncertainty

Monte-Carlo sequence entropy (MCSE/MCNSE)

Monte Carlo approximation of sequence entropy with N samples $\mathbf{y}^i \sim P(\mathbf{y} \mid \mathbf{x})$.

$$U_{\text{MCSE}}(\mathbf{x}) = -\frac{1}{N}\sum_{i=1}^{N} \log P(\mathbf{y}^i \mid \mathbf{x}).$$

The first p in "p log p" of the entropy equation disappear due to sampling following p.

As with sequence probability, MCSE has inherent bias toward shorter sequences due to autoregressive nature.

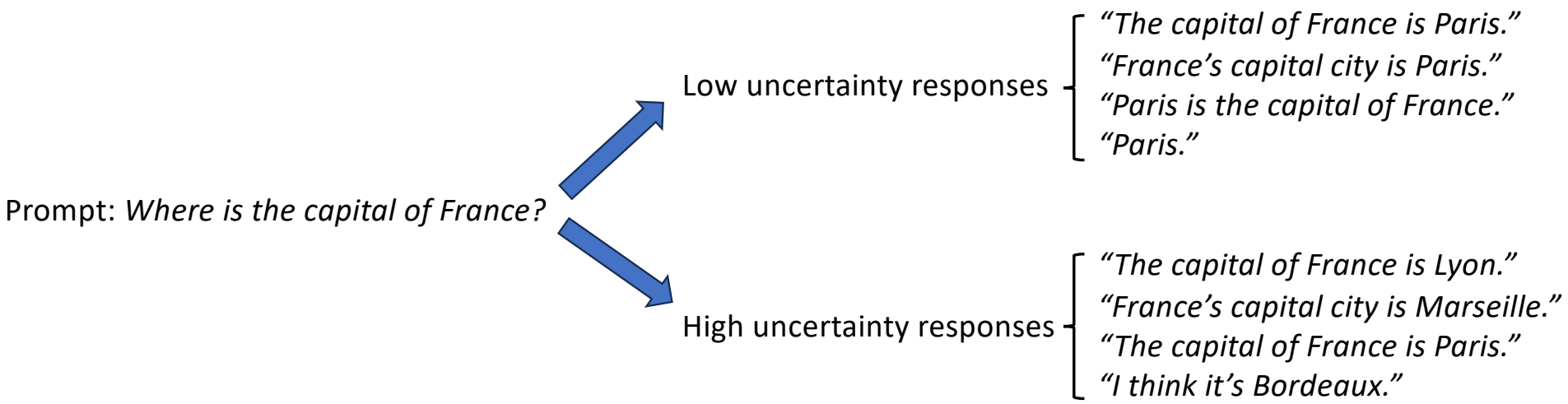
Length-normalized log-probability is proposed:

$$U_{\text{MCNSE}}(\mathbf{x}) = -\frac{1}{N}\sum_{i=1}^{N} \log \bar{P}(\mathbf{y}^i \mid \mathbf{x}).$$

Andrey Malinin and Mark J. F. Gales (2021). "Uncertainty Estimation in Autoregressive Structured Prediction". In: ICLR 2021.

Consistency-based confidence

Diverse responses to the same prompt indicate high uncertainty.



Consistency-based confidence

Frequency Scoring

Frequency Scoring estimates the confidence of a generation $\mathbf{y}^*$ by analyzing its similarity with other generation $\mathbf{y}^j$

$$C_{\text{FreqScore}}(\mathbf{y}^*, \mathbf{x}) = \underbrace{\sum_{j=1}^{N} \mathbf{1}[\text{NLI}(\mathbf{y}^*, \mathbf{y}^j) = \text{'}e\text{'}]}_{\text{amount of entail } y^j} - \underbrace{\sum_{j=1}^{N} \mathbf{1}[\text{NLI}(\mathbf{y}^*, \mathbf{y}^j) = \text{'}c\text{'}]}_{\text{amount of contradict } y^j}$$

Intuitively, a higher score indicate that the samples are concentrated at $\mathbf{y}^*$

$\mathbf{x}$: When did Marie Curie won Physics Nobel Prize?
$\mathbf{y}^*$: Marie Curie won Physics Nobel in 1903.

Samples:

- $\mathbf{y}^1$: She received the Nobel Prize in Physics in 1903. → 'e' (entailment)
- $\mathbf{y}^2$: She received the Nobel Prize in Physics in 1905. → 'c' (contradiction)
- $\mathbf{y}^3$: She was a chemist and physicist. → 'n' (neutral)

$$C_{\text{FreqScore}} = 1\,(\text{entailment}) - 1\,(\text{contradiction}) = \boxed{0}$$

High uncertainty, less concentration.

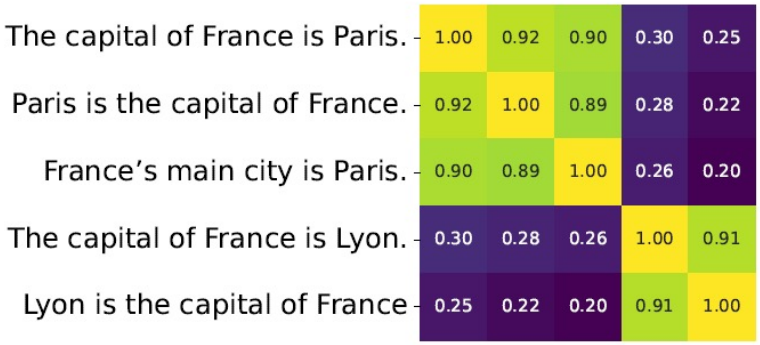
"Language Models with Conformal Factuality Guarantees". In: ICML 2024, Originally proposed for claim-level, with prompted GPT-4 serving as NLI model.

Consistency-based confidence

Lexical Similarity

Lexical Similarity compares samples via lexical metrics.

Similarity matrix: $S_{ij} = s(\mathbf{y}^i, \mathbf{y}^j)$, where s can be ROUGE-L, BLEU, etc.



Marina Fomicheva et al. (2020). "Unsupervised Quality Estimation for Neural Machine Translation". In: TACL 2020.

Consistency-based confidence

Lexical Similarity

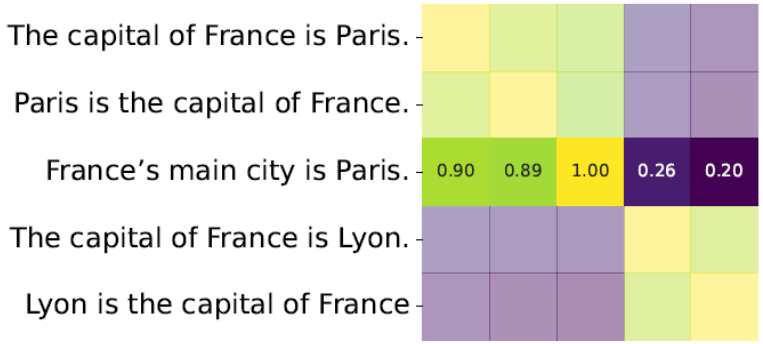
Uncertainty measures all values in the similarity matrix.

$$U_{\text{LexSim}}(x) = 1 - \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} s(\mathbf{y}^i, \mathbf{y}^j).$$

Confidence of a response $\mathbf{y}^*$:

$$C_{\text{LexSim}}(\mathbf{y}^*, \mathbf{x}) = \frac{1}{N} \sum_{j=1}^{N} s(\mathbf{y}^*, \mathbf{y}^j).$$

$\mathbf{y}^* =$



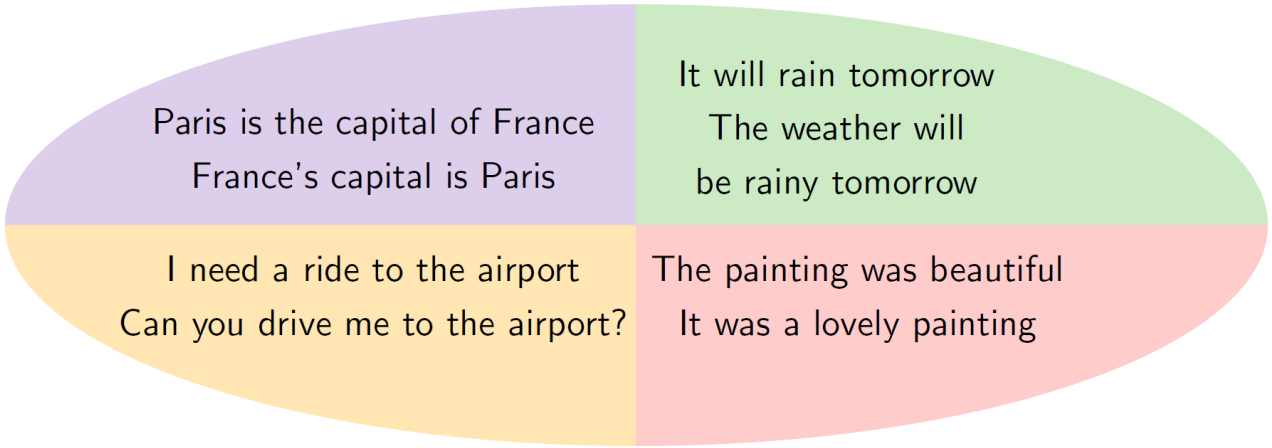
Marina Fomicheva et al. (2020). "Unsupervised Quality Estimation for Neural Machine Translation". In: TACL 2020.

Consistency-based confidence

Semantic Entropy

An LLM has consistent responses to the same prompt indicate they are more reliable.
We first need to classify responses into sets where each set has similar semantic meaning.

$$\mathcal{C} = \{\mathbf{y}: \forall \mathbf{y}', \mathrm{NLI}(\mathbf{y}, \mathbf{y}') = \text{`}e\text{'}\}$$



Four semantic sets. Each is painted in the same color.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar (2023). "Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation". In: ICLR 2023.

Consistency-based confidence

Semantic Entropy

Semantic Entropy is the entropy over semantic clusters.

Let $\{\mathcal{C}_m\}_{m=1}^{M}$ be semantic clusters from the semantic sets partition:

$$U_{SE} = -\frac{1}{N}\sum_{m=1}^{M} |\mathcal{C}_m| \log \hat{P}_m(\mathbf{x})$$

where

$$\hat{P}_m(\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{C}_m} P(\mathbf{y}|\mathbf{x})$$

High Entropy

Low Entropy

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar (2023). “Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation”. In: ICLR 2023.

Consistency-based confidence

Semantic Density

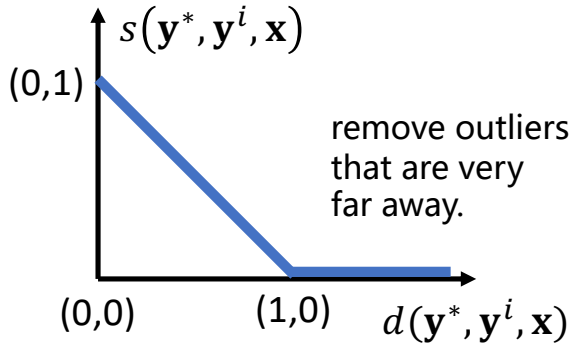
Another way of scaling sample probabilities via similarity.

We first define the distance between $\mathbf{y}^*$ and $\mathbf{y}^i$ by NLI analysis as

$$d(\mathbf{y}^*, \mathbf{y}^i, \mathbf{x}) = P\left(\mathrm{NLI}(\mathbf{x} \cup \mathbf{y}^*, \mathbf{x} \cup \mathbf{y}^i) = \text{`}c\text{'}\right) + \frac{1}{2} P\left(\mathrm{NLI}(\mathbf{x} \cup \mathbf{y}^*, \mathbf{x} \cup \mathbf{y}^i) = \text{`}n\text{'}\right).$$

Similarity is negatively related to distance:

$$s(\mathbf{y}^*, \mathbf{y}^i, \mathbf{x}) = (1 - d(\mathbf{y}^*, \mathbf{y}^i, \mathbf{x}))\mathbf{1}[d(\mathbf{y}^*, \mathbf{y}^i, \mathbf{x}) < 1].$$



Finally, if more $\mathbf{y}^i$ are similar to $\mathbf{y}^*$, $\mathbf{y}^*$ is more reliable.

$$C_{\mathrm{SemDen}}(\mathbf{y}^*, \mathbf{x}) = \frac{1}{\sum_{i=1}^{N} P(\mathbf{y}^i \mid \mathbf{x})} \sum_{i=1}^{N} P(\mathbf{y}^i \mid \mathbf{x}) s(\mathbf{y}^*, \mathbf{y}^i, \mathbf{x}) .$$

Xin Qiu and Risto Miikkulainen (2024). “Semantic Density: Uncertainty Quantification for Large Language Models through Confidence Measurement in Semantic Space”. In: NeurIPS 2024.

Consistency-based confidence

Confidence and Consistency-based Approaches (CoCoA)

A more flexible approach to confidence estimation can be achieved by combining various information-theoretic confidence measures with consistency analysis.

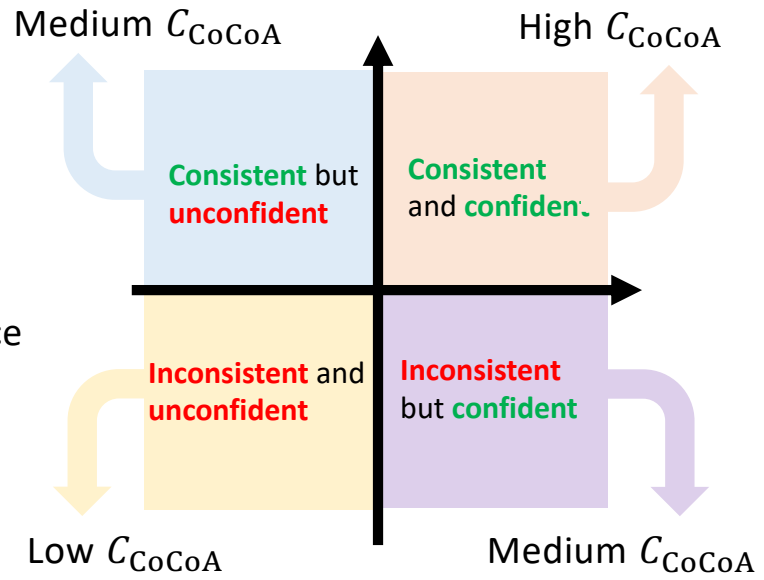
CoCoA proposes a multiplicative form of this combination:

$$C_{\text{CoCoA}}(\mathbf{y}^*, \mathbf{x}) = C_{\text{info}}(\mathbf{y}^*, \mathbf{x}) \cdot C_{\text{cons}}(\mathbf{y}^*, \mathbf{x})$$

C_{info} can be any information-theoretic confidence estimate, such as sequence probability, perplexity, mean token entropy etc., while C_{cons} is defined as

$$C_{\text{cons}}(\mathbf{y}^*, \mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} s(\mathbf{y}^*, \mathbf{y}^i)$$

where s can be ROUGE-L, BLEU, etc.



Roman Vashurin et al. (2025b). “Uncertainty Quantification for LLMs through Minimum Bayes Risk: Bridging Confidence and Consistency”. In: arXiv preprint arXiv:2502.04964.

Verbalized uncertainty

Request the numeric uncertainty to be directly reported by the model.

Example of a prompt:

```
Provide your best guess and the probability that it is correct
(0.0 to 1.0) for the following question. For example:

Guess: <most likely guess>
Probability: <the probability between 0.0 and 1.0 that
your guess is correct>

Question: Who was the first president of the United States?
```

Katherine Tian et al. (2023). “Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback”. In: EMNLP 2023.

Verbalized uncertainty

Examine the probability of True/False when asked if the given response is correct.

Example of P(True):

```
Question: Who was the first president of the United States?
Proposed Answer: George Washington was the first president.

Is the proposed answer:
  (A) True
  (B) False
The proposed answer is:
```

Resulting confidence is based on the probability of the token encoding “True”:

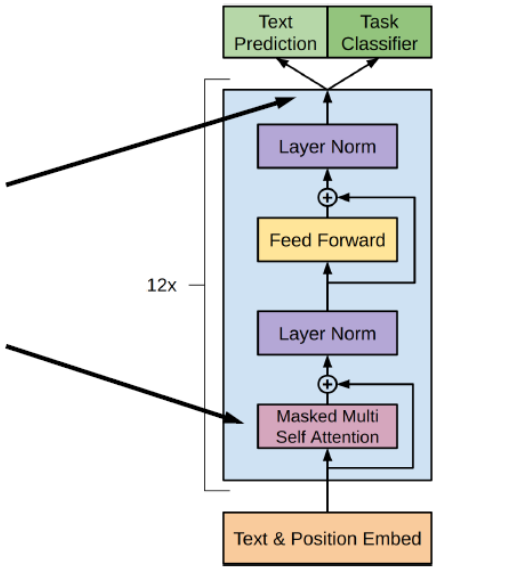
$$C_{\mathrm{P(True)}}(\mathbf{y}^*, \mathbf{x}) = P(\text{“True”}|\mathbf{y}^*, \mathbf{x})$$

Saurav Kadavath et al. (2022). “Language Models (Mostly) Know What They Know”. In: arXiv preprint arXiv:2207.05221.

Introspective method

Uncertainty quantification methods can leverage two internal signals from LLMs.

- Hidden states. The embedding vectors from each decoder layer for every generated token.
- Attention weights. The lower-triangular matrices, which illustrate how each token attends to the previous tokens during generation.



Transformer Decoder Architecture

Introspective method

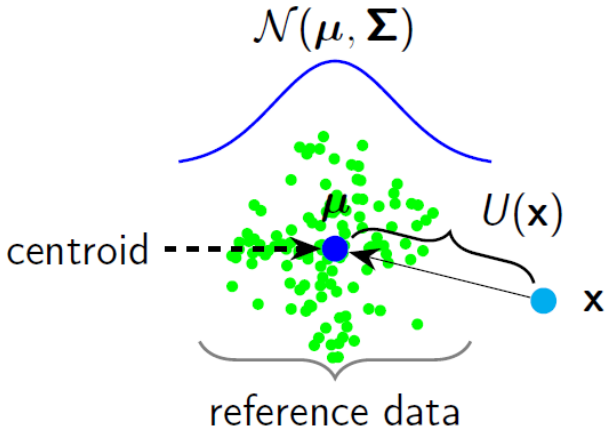
Hidden states

Embeddings drawn from intermediate LLM layers define a feature space in which prompts that lead to hallucinations and factual answers are readily separable.

Measuring the distance from the embedding of a prompt $\mathbf{x}$ to a "reference" distribution (embeddings of questions that LLM generates factual answers):

$\mathbf{x}$ = "Where is the capital of China" → LLM → target in question embedding

$\mathbf{x}_1$ = "Who is president of US"
$\mathbf{x}_2$ = "Who is president of Canada"
→ LLM → references embedding



Artem Vazhentsev et al. (2025a). "Token-Level Density-Based Uncertainty Quantification Methods for Eliciting Truthfulness of Large Language Models". In: NAACL 2025.

Introspective method

Hidden states

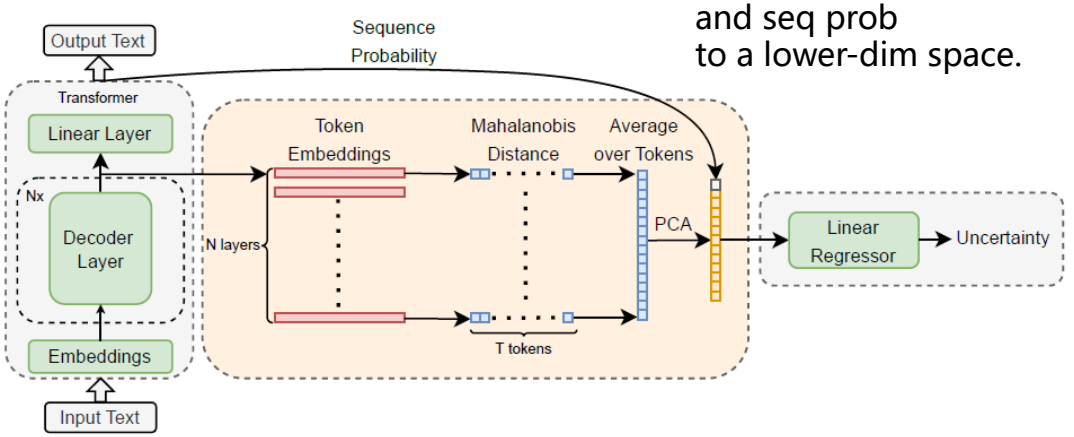
Token-Level Density-Based UQ uses Mahalanobis distance (MD) to quantify uncertainty on a test instance $\mathbf{x}$ consisting of T tokens.

$$U_{\mathrm{MD}}(t,l) = (h_l(t) - \mu_l)^\top \Sigma^{-1} (h_l(t) - \mu_l)$$

- Test token t
- Hidden state of t from layer l
- Covariance matrix
- Centroid of the reference data from layer l

$U_{\mathrm{MD}}(t,l)$ for all tokens and layers in $\mathbf{x}$ are collected to forecast LLM's prediction uncertainty via a trained regressor.

Embed the layers' distances and seq prob to a lower-dim space.



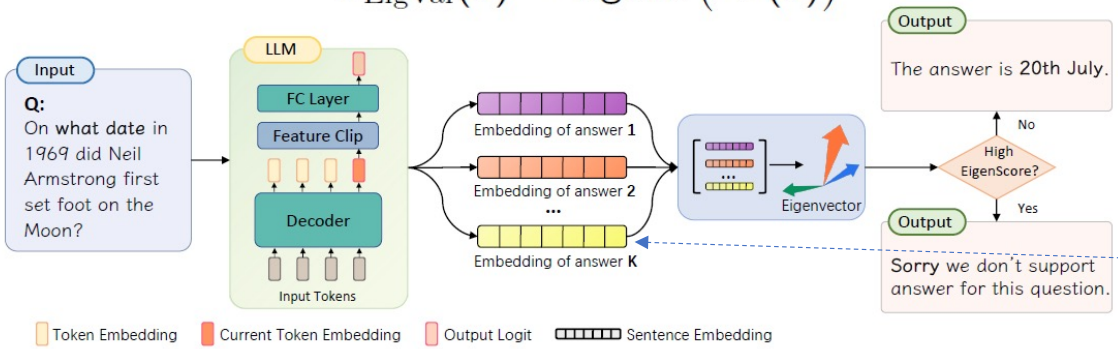
Artem Vazhentsev et al. (2025a). "Token-Level Density-Based Uncertainty Quantification Methods for Eliciting Truthfulness of Large Language Models". In: NAACL 2025.

Introspective method

Hidden states

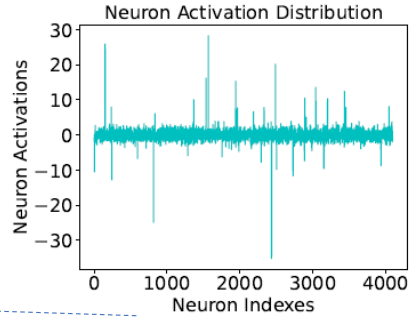
Eigenvalues should capture the interaction in latent space between the representations corresponding to hallucinated and truthful sequences. Specifically, let z^i be the embedding of each generated response $\mathbf{y}^i$ to $\mathbf{x}$. With $Z = [z^1, \ldots, z^K]$ and a centering matrix J, we define covariance matrix $\Sigma(\mathbf{x}) = Z^\top \cdot J \cdot Z$ and uncertainty is given by

$$\mathrm{U}_{\mathrm{EigVal}}(\mathbf{x}) = \log\det\left(\mathbf{\Sigma}^2(\mathbf{x})\right)$$



When the K generations have similar semantic, the sentence embeddings will be highly correlated and the eigenvalue will be close to 0. On the contrary, when the LLM is indecisive and hallucinating contents, the model will generate multiple sentences with diverse semantics leading to a more significant eigenvalue.

Notice there are extreme values in the last token embedding from penultimate layer, increasing the likelihood of generating overconfident and self-consistent generations.



remove outliers

Illustration of activation distributions in the penultimate layer of LLaMA-7B.

Feature clip (FC) is applied:

$$FC(h) = \begin{cases} h_{min}, & h < h_{min} \\ h, & h_{min} \leq h \leq h_{max} \\ h_{max} & h > h_{max} \end{cases}$$

Chao Chen et al. (2024). “INSIDE: LLMs’ Internal States Retain the Power of Hallucination Detection”. In: ICLR 2024.

Introspective method

Contextualized sequence likelihood (CSL)

Assign token probabilities weights w_i derived from the attention scores a_i.

$$\mathrm{U}_{\mathrm{CSL}}(\mathbf{x}) = -\sum_{i=1}^{L} w_i \log P(y_i \mid \mathbf{y}_{<i}, \mathbf{x}), \quad \text{where } w_i = \frac{a_i}{\sum_{i'=1}^{N} a_{i'}}.$$

The score a_i is the attention from token y_i on the last token in an **auxiliary verbalized uncertainty prompt** as shown on the right side.

```
Read the following question with
    optional context and decide if the
    answer correctly answer the question
    . Focus on the answer, and reply Y
    or N.
...
Context: Harry is a good witcher.
Question: How old is Harry?
Answer: Harry practices witchcraft.
 Decision: N. (The answer does not
    mention Harry's age.)
...
[$optional_context]
Question: [$question]
Answer: [$response]
 Decision:
```


Zhen Lin, Shubhendu Trivedi, and Jimeng Sun (2024a). "Contextualized Sequence Likelihood: Enhanced Confidence Scores for Natural Language Generation". In: EMNLP 2024.
Tianhang Zhang et al. (2023). "Enhancing Uncertainty-Based Hallucination Detection with Stronger Focus". In: EMNLP 2023.
Artem Vazhentsev et al. (2025b). "Uncertainty-Aware Attention Heads: Efficient Unsupervised Uncertainty Quantification for LLMs". In: arXiv preprint arXiv:2505.20045

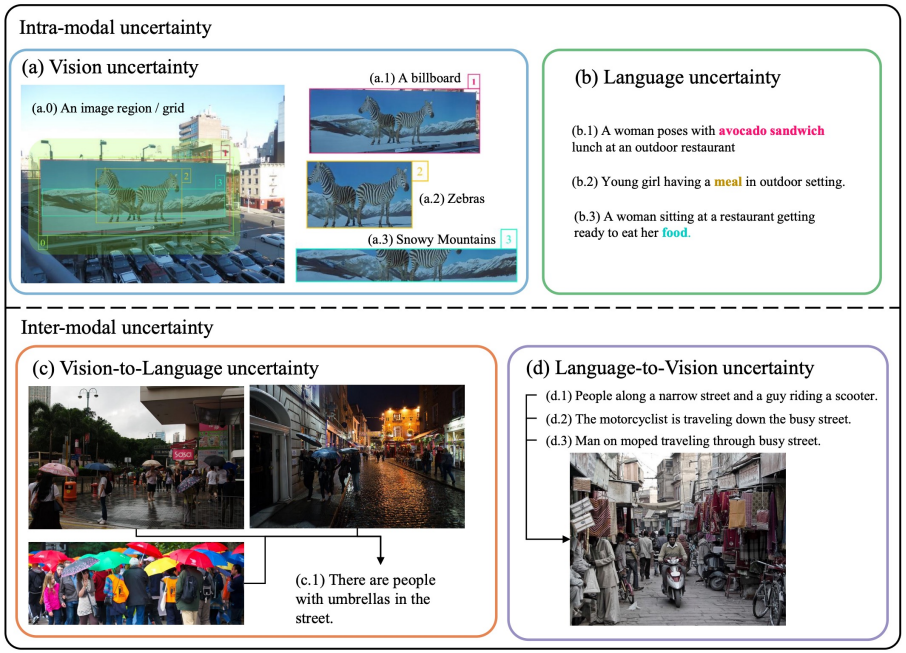
Uncertainty Quantification in VLM

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Uncertainty in Multimodality:

- Intramodal uncertainty: Ambiguity within a single modality (visual or textual).
- Intermodal uncertainty: Ambiguity in the correspondence between modalities (text to image, image to text).

Example of two different types Uncertainty



semantics in a modality is ambiguous.

mapping between modalities are ambiguous.

[1] Ji, Yatai, et al. "Map: Multimodal uncertainty-aware vision-language pre-training model."CVPR. 2023.

Challenge

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Overconfidence: The model gives extremely high confidence to its incorrect answer.

- Case: A model might incorrectly identify the number of objects in an image but give it 100% confidence.

Prompt: How many lamps are shown in this photo? Moreover, please express your estimate as a 95% confidence interval. Format your answer as:'[Lower Bound, Upper Bound]'

Answer: [10,12]

Hallucination: The model generates text descriptions that are inconsistent with the image content or have no basis in fact.

- Case: When answering questions about image, a model might give incorrect answers.

The image features a large clock [...] \n\nIn addition to the clock and the doorway, there are two people visible in the scene. [..]. The presence of these individuals suggests that [...].

[2] Groot, Tobias, and Matias Valdenegro-Toro. "Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models." arXiv (2024).

Core Methods for Quantification

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Uncertainty Quantification Methods:

1. Probabilistic Embeddings
2. Post-hoc Adaptation
3. Verbalized Uncertainty
4. Consistency & Self-Correction
5. Calibration Techniques
6. Conformal Prediction

Probabilistic Embeddings

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

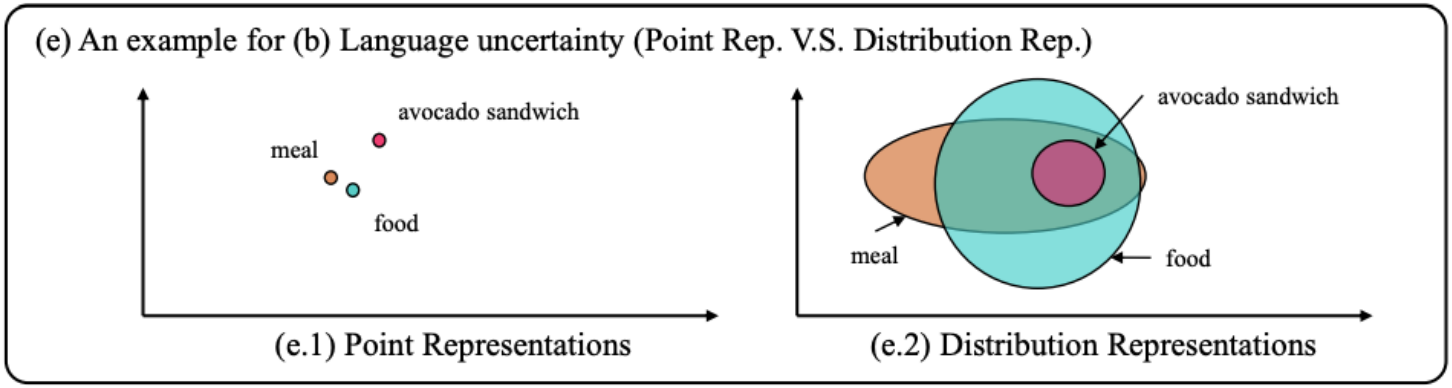
Core Idea:

- Traditional methods map the input (image/text) to a point in an embedding space.
- Probabilistic embedding methods map it to a probability distribution (such as a Gaussian distribution) to capture inherent ambiguity.

Advantages:

- Can better represent the multiple possible variations of a concept.
- The variance of the distribution can be used as a direct measure of uncertainty.

Example for language uncertainty by modeling as point representations and distribution representations.



Intuition: *food* is not a single concept but covers multiple ones.

[1] Ji, Yatai, et al. "Map: Multimodal uncertainty-aware vision-language pre-training model." CVPR. 2023.

Post-hoc Adaptation

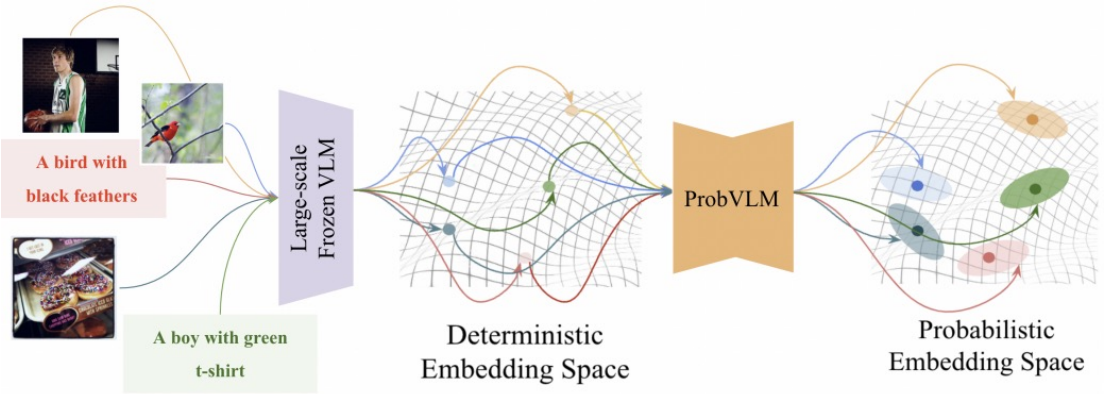
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Motivation:

Training a probabilistic model from scratch is expensive. Is it possible to add uncertainty estimation to a “frozen” (already trained) VLM (e.g., CLIP)?

ProbVLM:

1. Propose a lightweight "probabilistic adapter."
2. It accepts deterministic embeddings from a pre-trained VLM and predicts a probability distribution (mean and variance) for it (*similar to VAE).
3. This approach can be trained without requiring extensive data or computational resources.



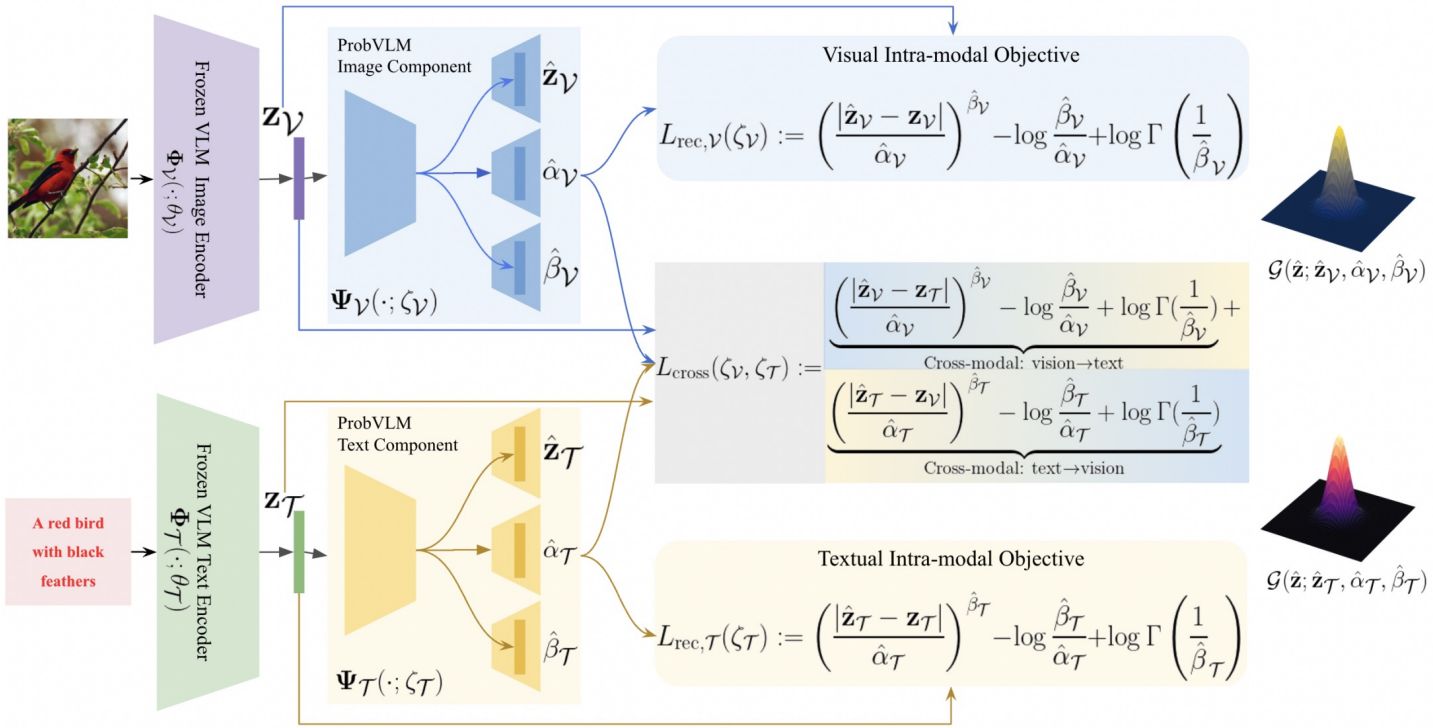
Provide probabilistic embeddings for deterministic pre-trained vision-language models that are frozen.

[3] Upadhyay, Uddeshya, et al. "Probvlm: Probabilistic adapter for frozen vison-language models.”CVPPR. 2023.

ProbVLM

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Estimate the complex probability distributions for the embeddings of the frozen deterministic vision-langue encoders, quantifying the uncertainties for their predictions.



[3] Upadhyay, Uddeshya, et al. "Probvlm: Probabilistic adapter for frozen vison-language models."CVPR. 2023.

ProbVLM

Implementation:

Train on some MS-COCO[1], Flickr-30k[2]

- Intra-modal Alignment
  - (Visual)
  - ProbVLM output: $\hat{z}_v$
  - Visual embedding: z_v
  - Spread of uncertainty: $\hat{\alpha}_v$
  - Form of distribution: β_v

Visual Intra-modal Objective

$$L_{\text{rec},\mathcal{V}}(\zeta_{\mathcal{V}}) := \left(\frac{|\hat{\mathbf{z}}_{\mathcal{V}} - \mathbf{z}_{\mathcal{V}}|}{\hat{\alpha}_{\mathcal{V}}}\right)^{\hat{\beta}_{\mathcal{V}}} - \log\frac{\hat{\beta}_{\mathcal{V}}}{\hat{\alpha}_{\mathcal{V}}} + \log\Gamma\left(\frac{1}{\hat{\beta}_{\mathcal{V}}}\right)$$

Textual Intra-modal Objective

$$L_{\text{rec},\mathcal{T}}(\zeta_{\mathcal{T}}) := \left(\frac{|\hat{\mathbf{z}}_{\mathcal{T}} - \mathbf{z}_{\mathcal{T}}|}{\hat{\alpha}_{\mathcal{T}}}\right)^{\hat{\beta}_{\mathcal{T}}} - \log\frac{\hat{\beta}_{\mathcal{T}}}{\hat{\alpha}_{\mathcal{T}}} + \log\Gamma\left(\frac{1}{\hat{\beta}_{\mathcal{T}}}\right)$$

- Cross-modal Alignment

$$L_{\text{cross}}(\zeta_{\mathcal{V}}, \zeta_{\mathcal{T}}) := \underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{V}} - \mathbf{z}_{\mathcal{T}}|}{\hat{\alpha}_{\mathcal{V}}}\right)^{\hat{\beta}_{\mathcal{V}}} - \log\frac{\hat{\beta}_{\mathcal{V}}}{\hat{\alpha}_{\mathcal{V}}} + \log\Gamma\left(\frac{1}{\hat{\beta}_{\mathcal{V}}}\right) +}_{\text{Cross-modal: vision}\rightarrow\text{text}}$$

$$\underbrace{\left(\frac{|\hat{\mathbf{z}}_{\mathcal{T}} - \mathbf{z}_{\mathcal{V}}|}{\hat{\alpha}_{\mathcal{T}}}\right)^{\hat{\beta}_{\mathcal{T}}} - \log\frac{\hat{\beta}_{\mathcal{T}}}{\hat{\alpha}_{\mathcal{T}}} + \log\Gamma\left(\frac{1}{\hat{\beta}_{\mathcal{T}}}\right)}_{\text{Cross-modal: text}\rightarrow\text{vision}}$$

- **Intuition**: if ProbVLP prediction is far from visual embedding, increase $\hat{\alpha}_v$ to lower the loss. When fixing $\hat{\alpha}_v$ make two embedding closer.

[3] Upadhyay, Uddeshya, et al. "Probvlm: Probabilistic adapter for frozen vison-language models."CVPR. 2023.
[8] Tsung-Yi Lin, et al. Microsoft coco: Common objects in context. In ECCV, 2014. 1, 2, 5
[9] Bryan A Plummer, et a,. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In CVPR, 2015. 1, 2, 5

Verbalized Uncertainty

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Core Idea:

- Directly use prompts to allow the model to describe the confidence of its answer in natural language.
- Provide a new dataset JUS (Japanese Uncertain Scenes).

Advantages:

- Simple and intuitive, requires no modification to the model structure, and is applicable to all prompt-based black-box models.

Limitations:

- The dataset size is small (39 images).
- Performed once per model

Prompt: How many birds are shown in this photo? Moreover, please rate your confidence in your answer between 0 and 100%. The answer should be in the format: "Answer (confidence%)".

Answer GPT-4V: It is not feasible to count the exact number of birds in the photo due to their movement, the potential for overlap, and the sheer number (confidence 0%). The image shows a flock of birds in flight, which makes it particularly challenging to provide an accurate count from a single static image.

Answer Gemini Pro Vision: 100 (80%)

Correct Answer: Over 100 birds, hard to count the exact number.

Prompt: Where was this photo taken? Moreover, please rate your confidence in your answer between 0 and 100%. The answer should be in the format: "Answer (confidence%)".

Answer GPT-4V: Hiroshima, Japan, during a lantern floating ceremony (confidence 90%).

Answer Gemini Pro Vision: Hiroshima, Japan (80%)

Correct Answer: Japan, Hiroshima, Atomic Bomb Dome

Prompt: How many octopuses are shown in this photo? Moreover, please rate your confidence in your answer between 0 and 100%. The answer should be in the format: "Answer (confidence%)".

Answer GPT-4V: 1 octopus (confidence 100%)

Answer Gemini Pro Vision: 2 (100%)

Correct Answer: 5

[2] Groot, Tobias, et al. "Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models." arXiv (2024).

Verbalized Uncertainty

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Calibration: Evaluates whether a model's stated confidence matches its actual accuracy.

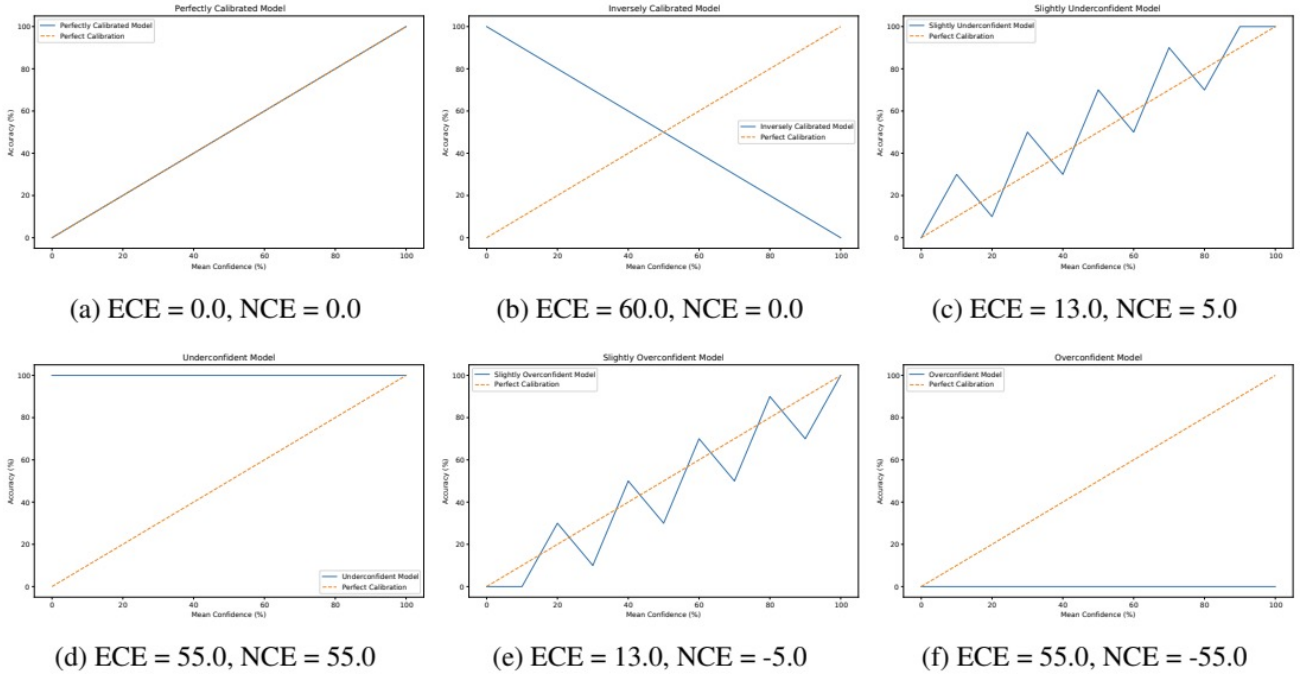
Evaluation Metrics:

- Expected Calibration Error (ECE)

$$\mathrm{ECE} = M^{-1} \sum_{m=1}^{M} |B_m| \, |\mathrm{acc}(B_m) - \mathrm{conf}(B_m)|$$

- Net Calibration Error (NCE)

$$\mathrm{NCE} = M^{-1} \sum_{m=1}^{M} |B_m| \, (\mathrm{acc}(B_m) - \mathrm{conf}(B_m))$$



(a) ECE = 0.0, NCE = 0.0 (b) ECE = 60.0, NCE = 0.0 (c) ECE = 13.0, NCE = 5.0

(d) ECE = 55.0, NCE = 55.0 (e) ECE = 13.0, NCE = -5.0 (f) ECE = 55.0, NCE = -55.0

M: number of bins
Acc(): Accuracy
Conf(): Confidence
ECE: Measures the average difference between confidence and accuracy.
NCF: determine whether a model is overconfident (negative values) or underconfident (positive values).

[2] Groot, Tobias, et al. "Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models." arXiv (2024).

Consistency & Self-Correction

Core Assumption:

- A reliable model that truly understands the problem should give consistent answers to questions with the same semantics but different wordings.

- If a model is highly sensitive to small perturbations that do not alter the semantics, its answers may be unreliable.

Application:

- Verify the consistency of the model's answers by generating multiple "rephrasings" of the question.

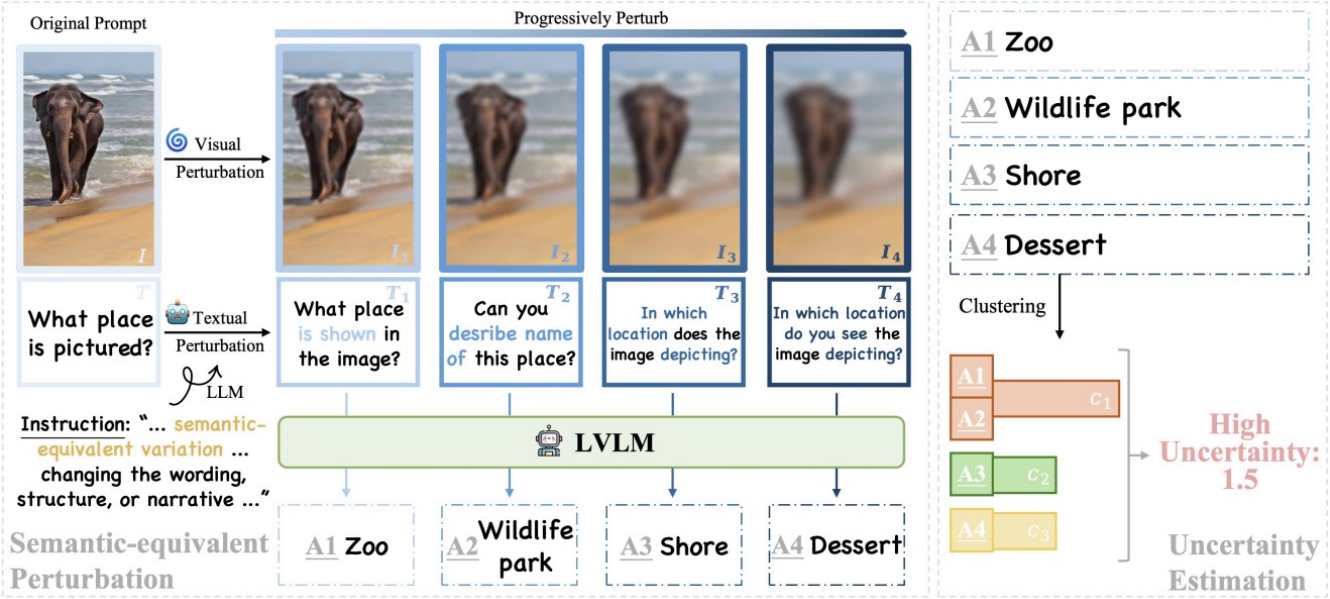
- Introduce a "critic" model to evaluate and correct the answers of the "reasoner" model.

VL-Uncertainty

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Measuring Uncertainty through Perturbations

1. Perturb the input using semantically equivalent perturbations:
   1. Visual perturbation: Apply varying degrees of Gaussian blur to the image.
   2. Textual perturbation: Use the LLM to paraphrase the question in different ways.
2. Multiple perturbed image-text pairs are fed into the VLM to obtain a set of answers.
3. Semantically cluster the answers and calculate the entropy of the answer distribution.



[4] Zhang, Ruiyang, Hu Zhang, and Zhedong Zheng. "Vl-uncertainty: Detecting hallucination in large vision-language model via uncertainty estimation." arXiv (2024).

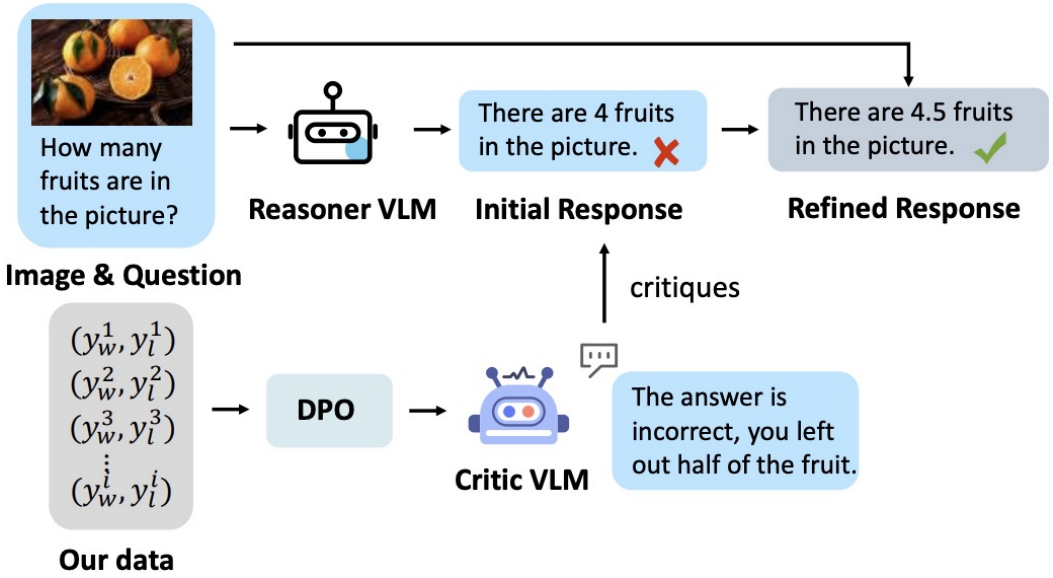
Critic-V

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Introducing a "Critic" for Error Correction Framework:

- Reasoner: A basic VLM responsible for generating preliminary answers.
- Critic: Another specially trained VLM responsible for evaluating the Reasoner's answers and providing natural language feedback and correction suggestions.

Workflow: This creates an iterative "generate-evaluate-correct" cycle to improve the quality and reliability of the final answer.



[5] Zhang, Di, et al. "Critic-v: Vlm critics help catch vlm errors in multimodal reasoning."CVPR. 2025.

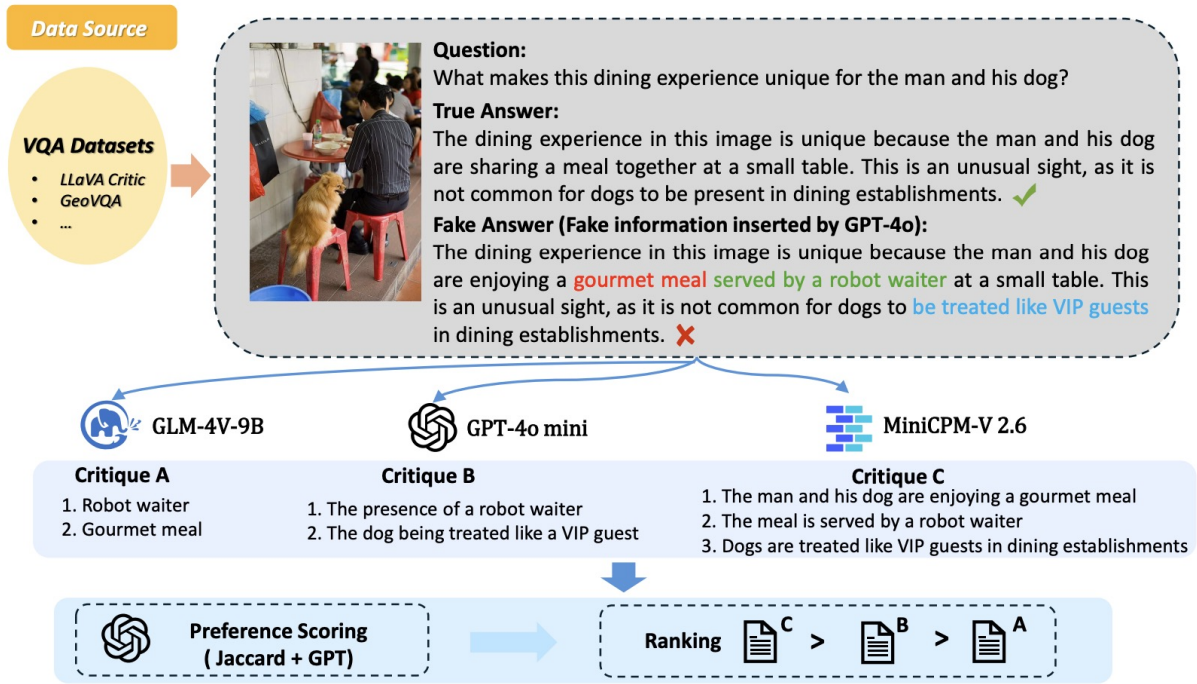
Critic-V

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

How to train a good critic?

- Data Construction:
  - **Visual Error Injection Technique**: Use GPT-4o generates "questionable answers."
  - Multiple VLMs are then used to generate "criticisms" for these incorrect answers.

- Preference Learning:
  - Use a rule-based reward function (e.g., the Jaccard index) to construct a preference dataset.
  - Use Direct Preference Optimizatio to train the critic model to generate high-quality feedback.



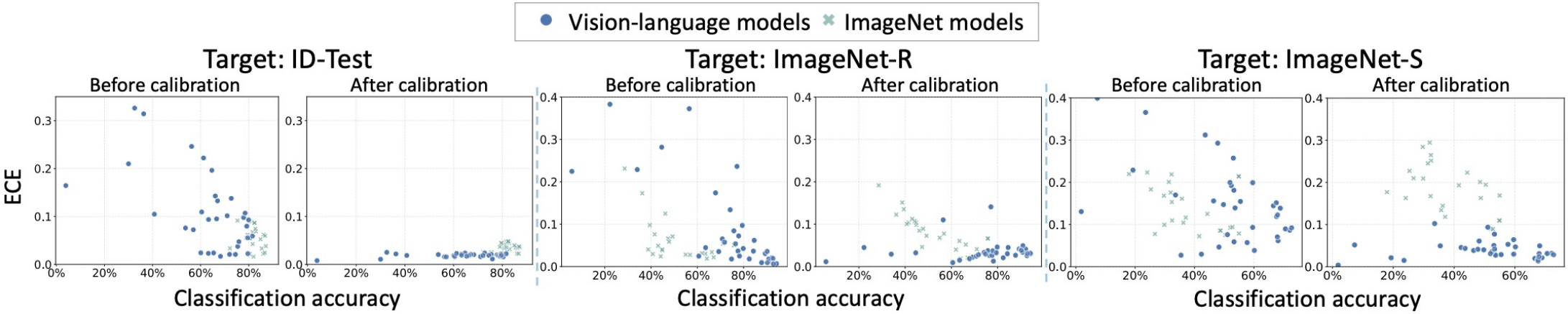
[5] Zhang, Di, et al. "Critic-v: Vlm critics help catch vlm errors in multimodal reasoning."CVPR. 2025.

Calibration Techniques

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Problem: The model's raw output probabilities often do not represent the true confidence.

Comparing the calibration performance of ImageNet-trained models and VLMs.



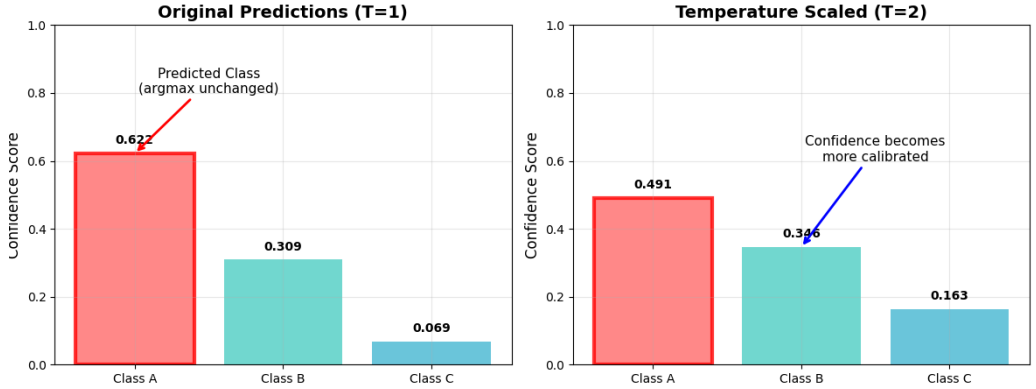
The ECE (lower is better) changes of the VLM and ImageNet models before and after temperature scaling, highlighting the superiority of the VLM calibration effect.

[6] Tu, Weijie, et al. "An empirical study into what matters for calibrating vision-language models." arXiv (2024).

Calibration Techniques

Standard Softmax: $$\sigma_i(\mathbf{z}) = \frac{\exp(\mathbf{z}_i)}{\sum_{j=1}^{k} \exp(\mathbf{z}_j)}$$

Temperature Scaling Softmax: $$\hat{p} = \max_i \frac{\exp(\mathbf{g}_i(\mathbf{x})/T)}{\sum_{j=1}^{n} \exp(\mathbf{g}_j(\mathbf{x})/T)}$$



Implementation:

For a trained classifier f, T is optimized using negative loglikelihood (NLL) on a calibration set.

$$L(T) = \frac{1}{N} \sum_{n=1}^{N} -\log(\hat{p}_{y^{(n)}}^{(n)}(T))$$

[6] Tu, Weijie, et al. "An empirical study into what matters for calibrating vision-language models." arXiv preprint arXiv:2402.07417 (2024).