Uncertainty Quantification for Generative Language Models

Maxim Panov

based on joint works with Artem Shelmanov, Roman Vashurin, Artem Vazhentsev, Ekaterina Fadeeva, Maiya Goloburda, Gleb Kuzmin, Preslav Nakov, Alexander Panchenko and many others

UncertaiNLP workshop at EMNLP 2025

November 9, 2025

Mohamed bin Zayed University of Artificial Intelligence

LLMs hallucinate I

A Tell me a bio of Dr. Artem Shelmanov

AI Dr. Artem Shelmanov is a prominent figure in the fields of **neuroscience**, **artificial intelligence (AI)**, and **cognitive science**. While specific details of his personal life and early years are not widely publicized, his professional contributions and impact are well-documented. Here's a concise biography based on available information:

Early Life and Education

Artem Shelmanov demonstrated a keen interest in science and technology from a young age. He pursued higher education in **neuroscience** and **computer science**, likely earning advanced degrees (Master's and Ph.D.) from prestigious institutions. His interdisciplinary background allowed him to bridge the gap between biological and artificial intelligence systems.

Career and Research

Dr. Shelmanov is best known for his groundbreaking work in **neuro-AI interfaces**, where he explores how the human brain interacts with AI systems. His research focuses on:

- **Brain-Computer Interfaces (BCIs):** Developing technologies that enable direct communication between the brain and external devices.

LLMs hallucinate I

A Tell me a bio of Dr. Artem Shelmanov

AI Dr. Artem Shelmanov is a prominent figure in the fields of **neuroscience**, **artificial intelligence (AI)**, and **cognitive science**. While specific details of his personal life and early years are not widely publicized, his professional contributions and impact are well-documented. Here's a concise biography based on available information:

Early Life and Education

Artem Shelmanov demonstrated a keen interest in science and technology from a young age. He pursued higher education in **neuroscience** and **computer science**, likely earning advanced degrees (Master's and Ph.D.) from prestigious institutions. His interdisciplinary background allowed him to bridge the gap between biological and artificial intelligence systems.

Career and Research

Dr. Shelmanov is best known for his groundbreaking work in **neuro-AI interfaces**, where he explores how the human brain interacts with AI systems. His research focuses on:

- **Brain-Computer Interfaces (BCIs):** Developing technologies that enable direct communication between the brain and external devices.

LLMs hallucinate II

User: How many letters in the word "nineteen"?

LLM: The word "nineteen" has 9 letters.

User: No there are 12 letters in the word "nineteen."

LLM: My apologies, you are correct! The word "nineteen" does indeed have 12 letters: N-I-N-E-T-E-E-N. Thank you for pointing out my mistake.

LLMs hallucinate II

User: How many letters in the word "nineteen"?

LLM: The word "nineteen" has 9 letters.

User: No there are 12 letters in the word "nineteen."

LLM: My apologies, you are correct! The word "nineteen" does indeed have 12 letters: N-I-N-E-T-E-E-N. Thank you for pointing out my mistake.

Uncertainty as a universal hallucination detector

Consider: $\mathbf{x}$ is an input sequence; $\mathbf{y}$ is an output sequence of length L.

Autoregressive LLM models:

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

In addition to $P(\mathbf{y} \mid \mathbf{x})$, we would like to obtain an uncertainty score:

$$\mathrm{U}(\mathbf{x}, \mathbf{y}).$$

We want $\mathrm{U}(\mathbf{x}, \mathbf{y})$ to correlate with LLM mistakes and help us to solve such tasks as **selective generation** and **hallucination detection**.

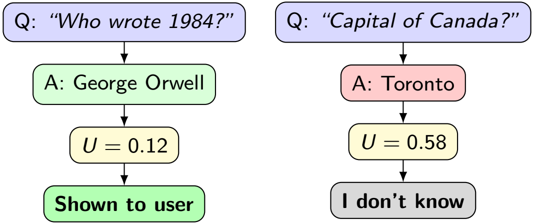
Selective generation

- **Goal:** Deliver answers only when the LLM is confident.

- **Selective rule:**

$$\text{output} = \begin{cases} \text{answer}, & \mathrm{U}(\mathbf{x}, \mathbf{y}) < \tau, \\ \text{“I don't know”}, & \mathrm{U}(\mathbf{x}, \mathbf{y}) \geq \tau. \end{cases}$$

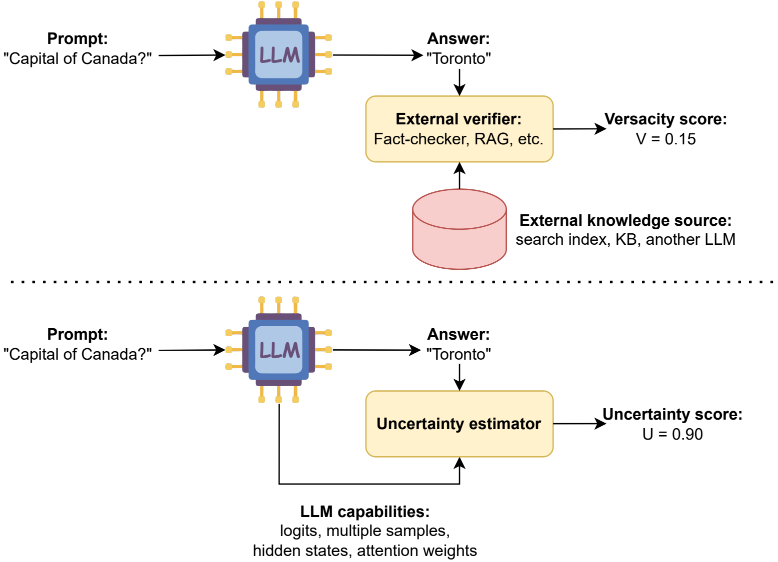
- By **rejecting** uncertain answers, we increase performance for remaining answers.



Example: with $\tau = 0.3$ we keep **70%** of answers while boosting accuracy from **80%** to **95%**

Important: not limited to question answering, also applies to machine translation, summarization, reasoning, ...

External verifier vs uncertainty quantification



Claim-level hallucination detection

Tell me a bio of Madonna.

Madonna is an American singer, songwriter, actress, and businesswoman. She was born on August 16, 1958, in Bay City, Michigan, and is best known for her provocative image and music. Madonna has sold over 300 million records worldwide, making her one of the best-selling music artists of all time. She has won numerous awards, including 20 Grammy Awards, a Golden Globe Award, and an Emmy Award. Madonna is also known for her philanthropic work and her activism on various social issues.

If $\mathrm{U}(\mathbf{x}, \mathbf{y}, c) \geq \tau$, highlight claim $c \in C(\mathbf{y})$ of LLM as potentially erroneous.

1. Theoretical Background

Uncertainty in classification

In binary classification, the value of predicted probability is a confidence in the positive class:

$$\widehat{p}(\mathbf{x}) = p\left(\mathbf{y} = 1 \mid \mathbf{x}\right).$$

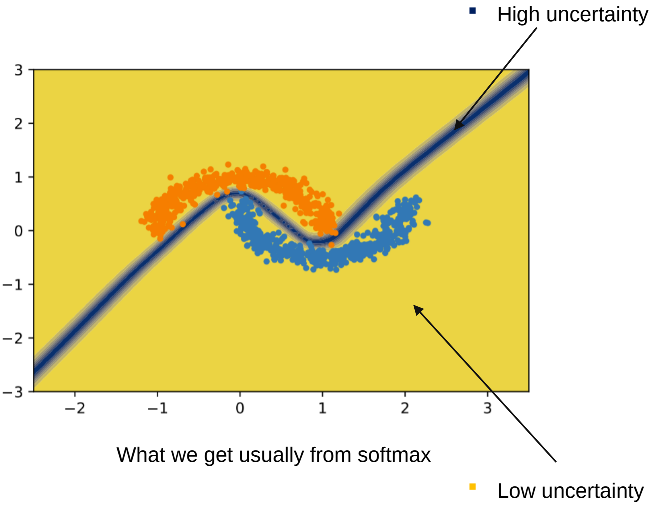
It generalizes to multiclass problems by using the value of the class with maximum probability:

$$\widehat{p}(\mathbf{x}) = \max_{c} p\left(\mathbf{y} = c \mid \mathbf{x}\right).$$

It leads to *Maximum Probability* uncertainty measure:

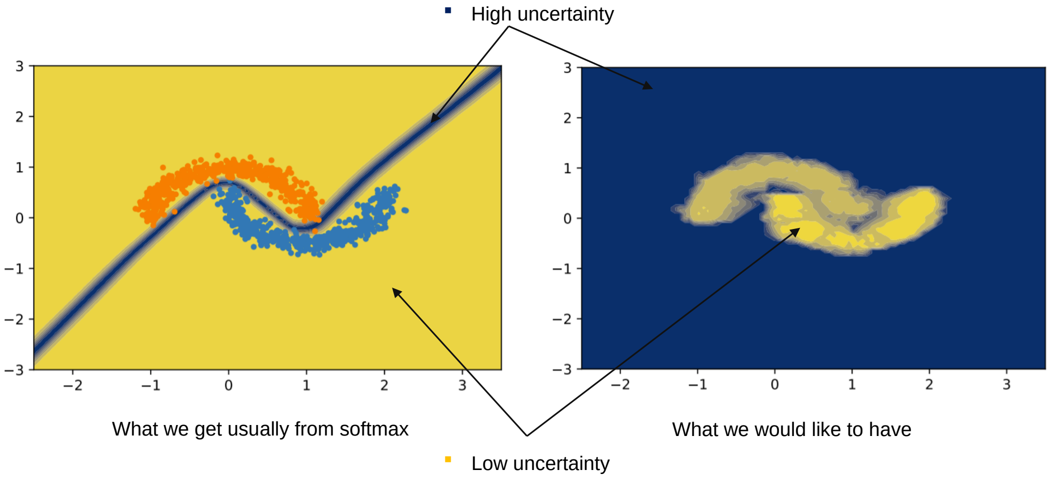
$$\mathrm{U}_{\mathrm{MP}}(\mathbf{x}) = 1 - \widehat{p}(\mathbf{x}) = 1 - \max_{c} p\left(\mathbf{y} = c \mid \mathbf{x}\right).$$

Why softmax probabilities are bad uncertainty estimates?



What we get usually from softmax

Why softmax probabilities are bad uncertainty estimates?



Question: what part of the uncertainty are we missing when using simple probabilities?

Epistemic vs aleatoric uncertainty[1]

The prediction uncertainty can be decomposed into **aleatoric** and **epistemic** terms.

- **Aleatoric** uncertainty reflects noise in data.
  - It could be due to noisy labels, class overlap, or data ambiguity.
  - **This part of the uncertainty <u>can not</u> be reduced when more data is given.**
- **Epistemic** uncertainty reflects lack of knowledge.
  - It is due to the total absence or just a few samples from a particular region.
  - **This part of the uncertainty <u>can</u> be reduced when more data is given.**

[1] Eyke Hüllermeier and Willem Waegeman (2021). "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods". In: *Machine learning*.

Peculiarities of UQ for LLMs

- When solving a classification problem for text generated by LLMs, we can quantify the uncertainty using the methods described above.

- However, there are more complex tasks sequence-to-sequence (Seq2Seq) in NLP:
  - machine translation;
  - question answering;
  - abstractive summarization.

- For these problems, we need to compute uncertainty not for a classification label but for the entire generated text.

Peculiarities of UQ for LLMs. Machine translation

Translate into French language: I want a small cup of coffee

Je veux une petite tasse de café.

Confidence: 100%

Peculiarities of UQ for LLMs. Question answering

Consider there is a human looking for help. How to cure him from pneumonia?

To cure a person suffering from pneumonia, it is important to seek medical attention promptly. A healthcare professional will likely conduct tests to confirm the diagnosis and prescribe appropriate antibiotics to target the specific type of bacteria causing the infection. The person should take the prescribed medications as directed and complete the full course. Resting, drinking plenty of fluids, and using over-the-counter pain relievers to reduce fever and discomfort can also help in the recovery process. Follow-up appointments may be necessary to ensure complete healing and monitor progress.

Confidence: 67%

Claim-level uncertainty quantification

Claim-level is more representative:

What Nobel prizes did Marie Curie win?

Marie Curie won the Nobel Prize in Chemistry in 1910 and Physics in 1903.

Extracted Claims:

1. Chemistry Nobel in 1910
   Confidence: 0.18
2. Physics Nobel in 1903
   Confidence: 0.82

Numerical confidence for each extracted claim.

Claim-level UQ pipeline:

- Extract claims from the generated text
- For each claim, get the tokens in the generation related to this claim
- Perform Uncertainty Quantification for each token
- Aggregate uncertainties of tokens into claim uncertainty

Types of uncertainty quantification methods

- Unsupervised methods
  - Typically are based on some intuitive statistic that reflects predictive uncertainty.
  - Example: sequence probability $P(\mathbf{y} \mid \mathbf{x})$.
- Supervised methods
  - Fit function that predicts erroneous generations based on labeled dataset.
  - Example: linear regression over embedding vectors of the model.

Types of unsupervised UQ methods

Black-box methods

- Verbalized uncertainty
  - Directly asking the model about its confidence in a generated answer

*There exist white-box variations of verbalized uncertainty methods.

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

*There exist white-box variations of verbalized uncertainty methods.

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

*There exist white-box variations of verbalized uncertainty methods.

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

*There exist white-box variations of verbalized uncertainty methods.

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

Hybrid methods

- Combinations of various methods above

*There exist white-box variations of verbalized uncertainty methods.

2. Unsupervised Uncertainty Quantification for LLMs

Information-theoretic uncertainty

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

$$\mathrm{U}_{\mathrm{SP}}(\mathbf{y}^*, \mathbf{x}) = -\log P\big(\mathbf{y}^* \mid \mathbf{x}\big) = -\sum_{j=1}^{L} \log P\big(y_j^* \mid \mathbf{y}_{<j}^*, \mathbf{x}\big).$$

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

[2] Roman Vashurin et al. (2025a). "UNCERTAINTY-LINE: Length-Invariant Estimation of Uncertainty for Large Language Models". In: *EMNLP 2025*.

Information-theoretic uncertainty

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

$$\mathrm{U}_{\mathrm{SP}}(\mathbf{y}^*, \mathbf{x}) = -\log P(\mathbf{y}^* \mid \mathbf{x}) = -\sum_{j=1}^{L} \log P(y_j^* \mid \mathbf{y}_{<j}^*, \mathbf{x}).$$

- Due to autoregressive probabilistic model, it has a **natural bias towards shorter sequences**.
- **Perplexity** (length-normalized log-probability) partially mitigates this bias[2]:

$$\mathrm{U}_{\mathrm{PPL}}(\mathbf{y}^*, \mathbf{x}) = -\log \bar{P}(\mathbf{y}^* \mid \mathbf{x}) = -\frac{1}{L}\sum_{j=1}^{L} \log P(y_j^* \mid \mathbf{y}_{<j}^*, \mathbf{x}).$$

[2] Roman Vashurin et al. (2025a). "UNCERTAINTY-LINE: Length-Invariant Estimation of Uncertainty for Large Language Models". In: *EMNLP 2025*.

Monte-Carlo sequence entropy (MCSE/MCNSE)

- **Key idea:** use multiple samples $\mathbf{y}^i \sim P(\mathbf{y} \mid \mathbf{x}), \quad i = 1, \dots, N.$
- **Monte Carlo approximation** of sequence entropy[3]:

$$\mathrm{U}_{\mathrm{MCSE}}(\mathbf{x}) = -\frac{1}{N} \sum_{i=1}^{N} \log P\big(\mathbf{y}^i \mid \mathbf{x}\big).$$

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

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

Monte-Carlo sequence entropy (MCSE/MCNSE)

- **Key idea:** use multiple samples $\mathbf{y}^i \sim P(\mathbf{y} \mid \mathbf{x}), \quad i = 1, \dots, N.$
- **Monte Carlo approximation** of sequence entropy[3]:

$$\mathrm{U}_{\mathrm{MCSE}}(\mathbf{x}) = -\frac{1}{N}\sum_{i=1}^{N} \log P\left(\mathbf{y}^i \mid \mathbf{x}\right).$$

- As with sequence probability, MCSE has inherent bias toward shorter sequences due to autoregressive nature.
- In the same fashion, to mitigate it, **length-normalized log-probability** $\bar{P}\left(\mathbf{y}^i \mid \mathbf{x}\right)$ can be used:

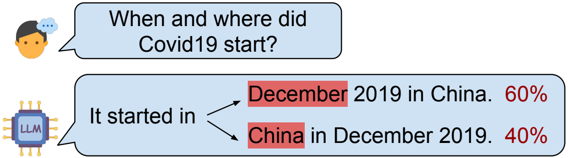
$$\mathrm{U}_{\mathrm{MCNSE}}(\mathbf{x}) = -\frac{1}{N}\sum_{i=1}^{N} \log \bar{P}\left(\mathbf{y}^i \mid \mathbf{x}\right).$$

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

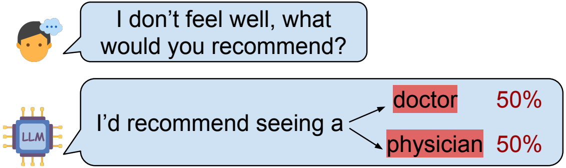
Variability is not always relevant for hallucination detection

Problem: probability-based methods do not account for different types of uncertainties. $P(y_j \mid \mathbf{y}_{<j}, \mathbf{x})$ can be low because model is not confident of:

which claim to mention at current token — *Claim type/order uncertainty*:



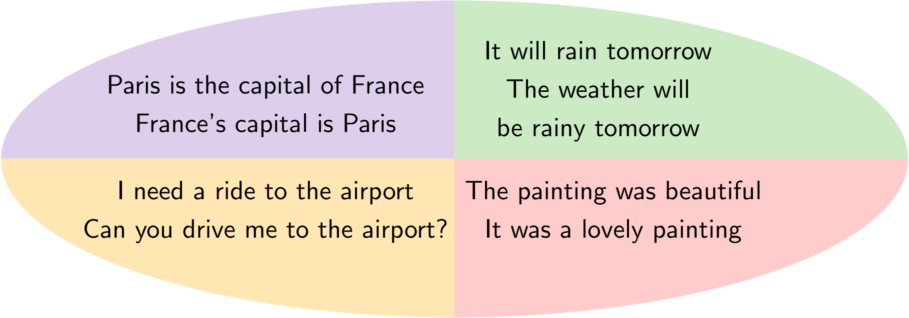
which words to use to mention the claim — *Surface form uncertainty*:



Adding semantics

- **Key idea:** different sequences can have the same semantic meaning.
- Consider a semantic set:

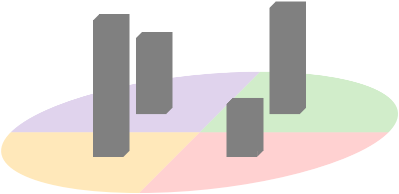
$$\mathcal{C} = \{\mathbf{y} \colon \forall \mathbf{y}' \in \mathcal{C}, \mathrm{NLI}(\mathbf{y}, \mathbf{y}') = \mathrm{NLI}(\mathbf{y}', \mathbf{y}) = \textit{entail}\}.$$



Semantic entropy

- Entropy over semantic clusters[4].
- Let $\{\mathcal{C}_m\}_{m=1}^{M}$ be semantic clusters, then

$$\mathrm{U}_{\mathrm{SE}} = -\frac{1}{N}\sum_{m=1}^{M} |\mathcal{C}_m| \log \hat{P}_m(\mathbf{x}), \quad \hat{P}_m(\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{C}_m} P(\mathbf{y} \mid \mathbf{x}).$$

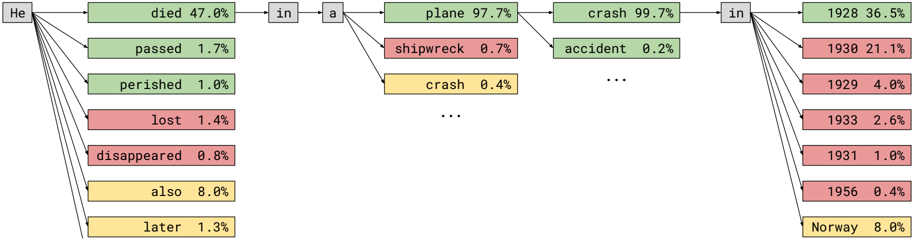


[4] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar (2023). "Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation". In: *ICLR 2023*.

Claim-Conditioned Probability: CCP

Key idea: look on semantics along the generation.

Solution: Claim-Conditioned Probability[5]



$$\mathrm{CCP}(y_j^*, \mathbf{y}_{<j}^*, \mathbf{x}) = \frac{\sum P(\;\text{green}\;)}{\sum P(\;\text{green} + \text{red}\;)}$$

[5] Ekaterina Fadeeva et al. (2024). "Fact-Checking the Output of Large Language Models via Token-Level Uncertainty Quantification". In: *Findings of ACL 2024*.

Claim-Conditioned Probability: CCP

Solution: Claim-Conditioned Probability[6]

$$\mathrm{CCP}(y_j^*, \mathbf{y}_{<j}^*, \mathbf{x}) = \frac{\sum_{t \in \mathcal{V}} \mathbb{1}\left[\mathtt{NLI}(t, y_j^*) = \text{`e'}\right] P(y_j^* = t \mid \mathbf{y}_{<j}^*, \mathbf{x})}{\sum_{t \in \mathcal{V}} \mathbb{1}\left[\mathtt{NLI}(t, y_j^*) \in \{\text{`e'}, \text{`c'}\}\right] P(y_j^* = t \mid \mathbf{y}_{<j}^*, \mathbf{x})}.$$

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

$$\mathrm{U}_{\mathrm{CCP}}(\mathbf{y}^*, \mathbf{x}) = -\sum_{j=1}^{L} \log \mathrm{CCP}(y_j^*, \mathbf{y}_{<j}^*, \mathbf{x}).$$

[6] Ekaterina Fadeeva et al. (2024). "Fact-Checking the Output of Large Language Models via Token-Level Uncertainty Quantification". In: *Findings of ACL 2024*.

Towards more grounded UQ via Minimum Bayes Risk

Minimum Bayes Risk (MBR) decoding:

$$\mathbf{y}^* = \arg\min_{\mathbf{y} \in \mathcal{Y}} R(\mathbf{y} \mid \mathbf{x}),$$

where $R(\mathbf{y} \mid \mathbf{x})$ is a risk function:

$$R(\mathbf{y} \mid \mathbf{x}) = \mathbb{E}_{\mathbf{y}' \sim p(\mathbf{y} \mid \mathbf{x})}\, r(\mathbf{y}, \mathbf{y}').$$

with $r(\mathbf{y}, \mathbf{y}')$ being a pairwise loss function.

Natural UQ measure under MBR:

$$\mathrm{U}_{\mathrm{MBR}}(\mathbf{y}^* \mid \mathbf{x}) = \min_{\mathbf{y} \in \mathcal{Y}} R(\mathbf{y} \mid \mathbf{x}) = R(\mathbf{y}^* \mid \mathbf{x}).$$

Towards more grounded UQ via Minimum Bayes Risk

Consider

$$r(\mathbf{y}, \mathbf{y}') = \mathbf{1}\{\mathbf{y}' \neq \mathbf{y}\}.$$

In this case, the Bayes risk corresponds to the expected zero-one loss of decoding:

$$R_{0/1}(\mathbf{y} \mid \mathbf{x}) = \mathbb{E}_{\mathbf{y}' \sim p(\mathbf{y} \mid \mathbf{x})} \mathbf{1}\{\mathbf{y}' \neq \mathbf{y}\} = 1 - p(\mathbf{y} \mid \mathbf{x}).$$

This leads to *Sequence Probability (SP)* UQ measure:

$$\mathrm{U}_{\mathrm{SP}}(\mathbf{y}^* \mid \mathbf{x}) = 1 - p\big(\mathbf{y}^* \mid \mathbf{x}\big).$$

Consistency-based uncertainty

- **Key idea:** diverse responses to the same prompt indicate high uncertainty.

Low uncertainty	**High uncertainty**
LLM	**LLM**
The capital of France is Paris.	The capital of France is Lyon.
France's capital city is Paris..	France's capital city is Marseille.
Paris is the capital of France.	The capital of France is Paris.
Paris.	I think it's Bordeaux.

Consistency-based uncertainty

- Consider a set of sampled outputs: $\mathbf{y}^i \sim p(\mathbf{y} \mid \mathbf{x}), \quad i = 1, \ldots, N.$
- Similarity matrix: $S_{ij} = s(\mathbf{y}^i, \mathbf{y}^j)$, where s can be ROUGE-L, BLEU, NLI, ...

The capital of France is Paris.	1.00	0.92	0.90	0.30	0.25
Paris is the capital of France.	0.92	1.00	0.89	0.28	0.22
France's main city is Paris.	0.90	0.89	1.00	0.26	0.20
The capital of France is Lyon.	0.30	0.28	0.26	1.00	0.91
Lyon is the capital of France	0.25	0.22	0.20	0.91	1.00

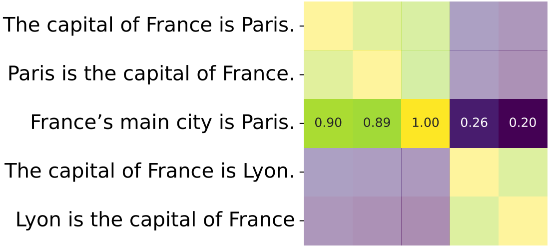
Consistency-based uncertainty

- Within MBR framework take $r(\mathbf{y}, \mathbf{y}') = 1 - s(\mathbf{y}, \mathbf{y}')$.
- Now MBR-based uncertainty becomes:

$$\mathrm{U}_{\mathsf{cons}}(\mathbf{y}_* \mid \mathbf{x}) = 1 - \mathbb{E}_{\mathbf{y}' \sim p(\mathbf{y} \mid \mathbf{x})} s(\mathbf{y}_*, \mathbf{y}').$$

- Consistency-based UQ measure given samples $\{\mathbf{y}^i\}$:

$$\mathrm{U}_{\mathrm{Cons}}(\mathbf{y}^*, \mathbf{x}) = 1 - \frac{1}{N} \sum_{i=1}^{N} s(\mathbf{y}^*, \mathbf{y}^i).$$



CoCoA

- **Key idea:** combine information theoretic and consistency-based methods via MBR:

$$r(\mathbf{y}, \mathbf{y}' \mid \mathbf{x}) = \mathrm{U}_{\mathrm{inf}}(\mathbf{y}, \mathbf{x}) \cdot \big(1 - s(\mathbf{y}, \mathbf{y}')\big),$$

- $\mathrm{U}_{\mathrm{inf}}$ can be any information-theoretic uncertainty estimate.
- The resulting CoCoA[7] UQ measure becomes:

$$\mathrm{U}_{\mathrm{CoCoA}}(\mathbf{y}^*, \mathbf{x}) = \mathrm{U}_{\mathrm{inf}}(\mathbf{y}^*, \mathbf{x}) \cdot \mathrm{U}_{\mathrm{cons}}(\mathbf{y}^*, \mathbf{x}).$$

- For example, one can consider:

$$\mathrm{U}_{\mathrm{CoCoA}}(\mathbf{y}^*, \mathbf{x}) = \Big[- \log P(\mathbf{y}^* \mid \mathbf{x})\Big] \cdot \frac{1}{N} \sum\nolimits_{i=1}^{N} \big(1 - s(\mathbf{y}^*, \mathbf{y}^i)\big).$$

[7] Roman Vashurin et al. (2025b). "Uncertainty Quantification for LLMs through Minimum Bayes Risk: Bridging Confidence and Consistency". In: *NeurIPS 2025*.

Results

UQ Method	Llama-3.1 8B			Gemma-3 12B			Mean
	QA	ATS	NMT	QA	ATS	NMT	
Sequence Probability	.353	.173	.395	.382	.317	.479	.350
Perplexity	<u>.392</u>	**.354**	.402	<u>.456</u>	.334	.548	<u>.414</u>
Mean Token Entropy	.335	<u>.336</u>	<u>.437</u>	.447	.311	<u>.576</u>	.407
MC SE	.330	.118	.382	.394	.060	.346	.272
MC NSE	.343	.180	.384	.369	.085	.428	.298
NumSemSets	.328	.136	.047	.356	.114	.038	.170
Semantic Entropy	.339	.119	.393	.392	.048	.353	.274
EigVal	.369	.133	.391	.399	.084	.295	.279
Semantic Density	.136	.316	.382	.139	<u>.353</u>	.405	.289
CoCoA	**.413**	.283	**.506**	**.464**	**.385**	**.626**	**.446**

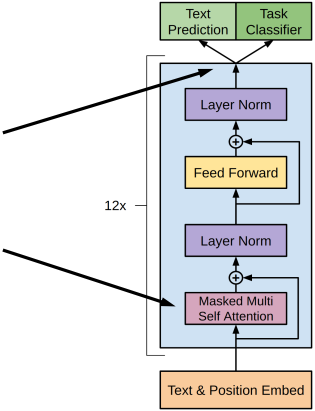
Table: Mean PRR↑ across tasks for the evaluated LLMs

- CoCoA shows the strongest overall performance, consistently ranking one of the best by average performance in a task.

Inside LLM

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.

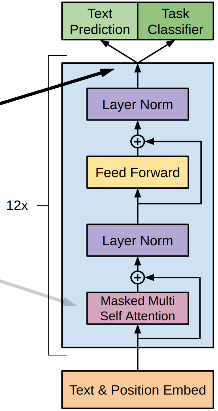


Transformer Decoder Architecture

Inside LLM: Hidden states

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.



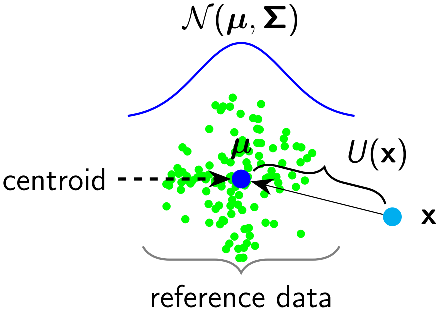
Transformer Decoder Architecture

Inside LLM: Hidden states

Key idea:

- Softmax always gives probabilities that sum to one.
- Unusual behaviour might be easier to observe before the softmax, i.e. looking at hidden states.

Density-based approach:



Mahalanobis distance

Mahalanobis distance (MD)[8,9] is proportional to the negative log-likelihood of a multivariate normal distribution, up to an additive constant:

$$\underbrace{\mathrm{U}_{\mathrm{MD}}(\mathbf{x}}_{\text{Tested instance}}, l) = (\underbrace{h_l}_{\text{Hidden representation of } \mathbf{x} \text{ from layer } l}(\mathbf{x}) - \boldsymbol{\mu})^T \underbrace{\boldsymbol{\Sigma}^{-1}}_{\text{Covariance matrix}}(h_l(\mathbf{x}) - \underbrace{\boldsymbol{\mu}}_{\text{Centroid of the reference data}}).$$

[8] Kimin Lee et al. (2018). "A simple unified framework for detecting out-of-distribution samples and adversarial attacks". In: *NeurIPS 2018*.

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

Eigenvalues analysis

Key idea: eigenvalues should capture the interaction in latent space between the different representations corresponding to hallucinated and truthful sequences.

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

Covariance matrices $\boldsymbol{\Sigma}(\mathbf{x})$ derived from:

- sequence-level embeddings of $\mathbf{y}^i$ aggregated across sampled generations[10];
- token-level embeddings of the concatenation $\mathbf{x} \oplus \mathbf{y}$[11].

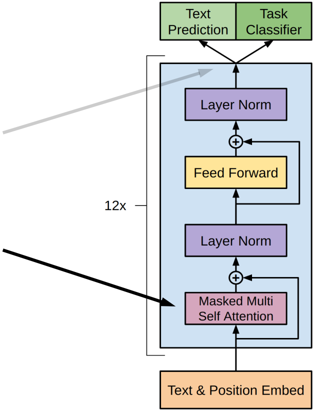
[10] Chao Chen et al. (2024). "INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection". In: *ICLR 2024*.

[11] Gaurang Sriramanan et al. (2024). "LLM-Check: Investigating Detection of Hallucinations in Large Language Models". In: *NeurIPS 2024*.

Inside LLM: Attention weights

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

Inside LLM: Attention weights

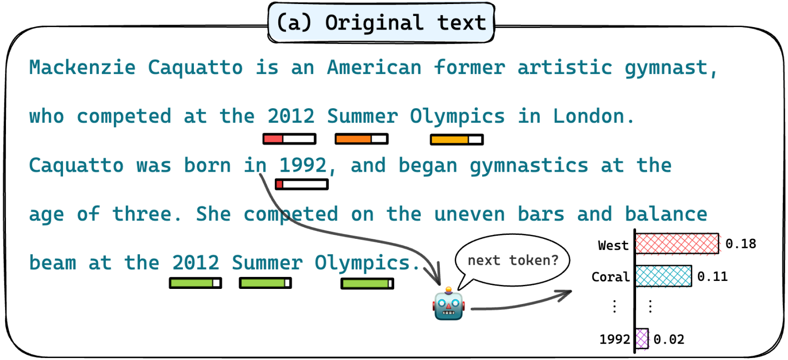
Intuition: the model attends differently when generating hallucinations versus truthful responses

- Analyze the eigenvalues of attention matrices[12]
- Reweight token probabilities based on attention weights
- Propagate uncertainty from preceding tokens using attention weights

[12] Gaurang Sriramanan et al. (2024). "LLM-Check: Investigating Detection of Hallucinations in Large Language Models". In: *NeurIPS 2024*.

Problem of conditional dependency of generation steps

Problem: conditional dependency on previous steps means the LLM assumes everything generated so far is correct, which may not always be the case.



Attention as a proxy for conditional dependence

- **Key idea:** if model attends to something uncertain for particular token, then we should decrease its confidence:

$$\mathbf{c}(y_i) = w_{ii}\, P(y_i \mid \mathbf{y}_{<i}, \mathbf{x}) + \sum_{j=1}^{i-1} w_{ij}\, \mathbf{c}(y_j),$$

  where w_{ij} is should be proportional to attention from token i to token j.

- This idea was coined in Focus[13] method.

- **Key difficulty:** different heads contribute unequally to hallucinations and aggregation over heads leads to unstable results.

[13] Tianhang Zhang et al. (2023). "Enhancing Uncertainty-Based Hallucination Detection with Stronger Focus". In: *EMNLP 2023*.

Recurrent attention-based uncertainty quantification: RAUQ

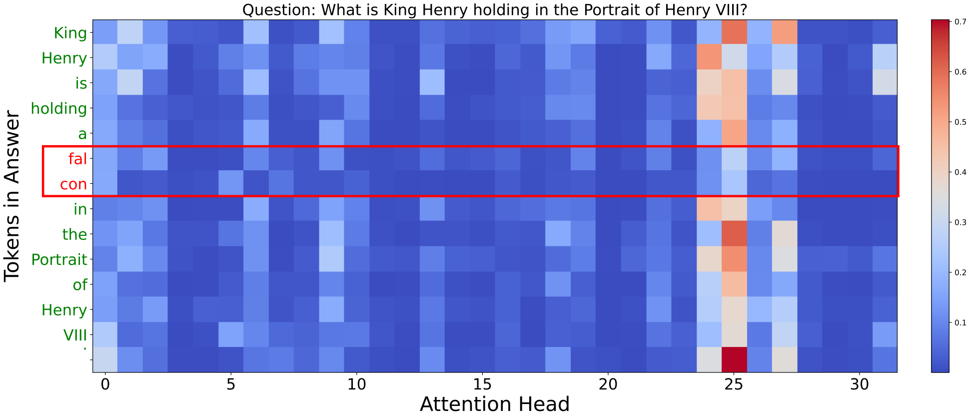
Key idea: identify patterns in attention maps that reveal hallucinations[14].

- *Question*: What is King Henry holding in the Portrait of Henry VII?
- *Correct Answer*: gloves and dagger.
- *LLM Answer (Llama-3.1 8b)*: King Henry is holding a **falcon** in the Portrait of Henry VII.

[14] Artem Vazhentsev et al. (2025b). "Uncertainty-Aware Attention Heads: Efficient Unsupervised Uncertainty Quantification for LLMs". In: *arXiv preprint arXiv:2505.20045*.

Recurrent attention-based uncertainty quantification: RAUQ

- Most attention heads show low weights.
- The 25th head: high attention for correct tokens, low for the hallucinated token.



Recurrent attention-based uncertainty quantification: RAUQ

1. Select the most informative attention head per layer:

$$\mathbf{h}_l(\mathbf{y}) = \underset{h=1 \dots H}{\arg\max} \; \frac{1}{L-1} \sum\nolimits_{i=2}^{L} a_{i,i-1}^{lh}.$$

Recurrent attention-based uncertainty quantification: RAUQ

1. Select the most informative attention head per layer:

$$\mathbf{h}_l(\mathbf{y}) = \operatorname*{arg\,max}_{h=1\ldots H} \frac{1}{L-1} \sum\nolimits_{i=2}^{L} a^{lh}_{i,i-1}.$$

2. Compute token-level layer-wise recurrent confidence score:

$$\mathbf{c}_l(y_i) = \begin{cases} P(y_i \mid \mathbf{x}), & \text{if } i = 1, \\ \alpha \cdot P(y_i \mid \mathbf{y}_{<i}, \mathbf{x}) + (1-\alpha) \cdot a^{l\,\mathbf{h}_l}_{i,i-1} \cdot \mathbf{c}_l(y_{i-1}), & \text{if } i > 1. \end{cases}$$

Recurrent attention-based uncertainty quantification: RAUQ

1. Select the most informative attention head per layer:

$$\mathbf{h}_l(\mathbf{y}) = \arg\max_{h=1\ldots H} \frac{1}{L-1} \sum_{i=2}^{L} a_{i,i-1}^{lh}.$$

2. Compute token-level layer-wise recurrent confidence score:

$$\mathbf{c}_l(y_i) = \begin{cases} P(y_i \mid \mathbf{x}), & \text{if } i = 1, \\ \alpha \cdot P(y_i \mid \mathbf{y}_{<i}, \mathbf{x}) + (1-\alpha) \cdot a_{i,i-1}^{l\,\mathbf{h}_l} \cdot \mathbf{c}_l(y_{i-1}), & \text{if } i > 1. \end{cases}$$

3. Aggregate the token-level layer-wise uncertainty scores to the final score:

$$\mathrm{U}_{\mathsf{RAUQ}}(\mathbf{y}) = \max_{l \in \mathcal{L}} \left[-\frac{1}{L} \sum_{i=1}^{L} \log \mathbf{c}_l(y_i) \right].$$

Introspective methods results

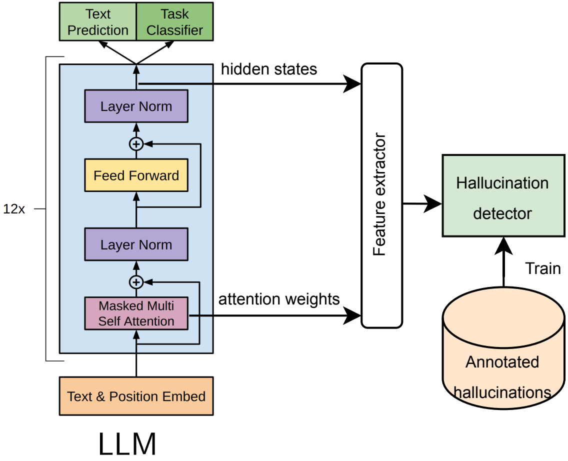
UQ Method	Llama-3.1 8B			Gemma-3 12B			Mean
	QA	ATS	NMT	QA	ATS	NMT	
MSP	.353	.173	.395	.382	.317	.479	.350
Mean Token Entropy	.335	**.336**	.437	.447	.311	.576	.407
CCP	.322	.195	.343	.392	.338	.428	.336
Attention Score	.075	.077	.172	.061	.002	.126	.085
Focus	.282	.259	.372	.384	.310	.499	.351
Eigen Score	.363	.129	.394	.367	.086	.346	.281
Semantic Entropy	.339	.119	.393	.392	.048	.353	.274
SAR	**.377**	.214	.437	.406	.096	.458	.331
Semantic Density	.136	.316	.382	.139	**.353**	.405	.289
RAUQ	.354	.335	**.494**	**.447**	.325	**.613**	**.428**

Table: Mean PRR↑ across tasks for the evaluated LLMs

- RAUQ shows the strongest overall performance, consistently ranking one of the best by average performance in a task.

3. Supervised UQ Methods

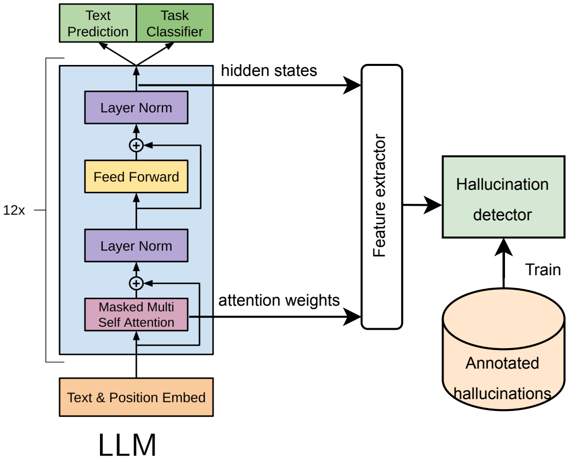
From manual engineering to data-driven methods



Intuition: Let's bypass all UQ challenges by learning from the data!

1. Create features from internal states and/or attention weights of the LLM.
2. Collect training data: hallucinations / non-hallucinations.
3. Train a supervised hallucination detector.

From manual engineering to data-driven methods

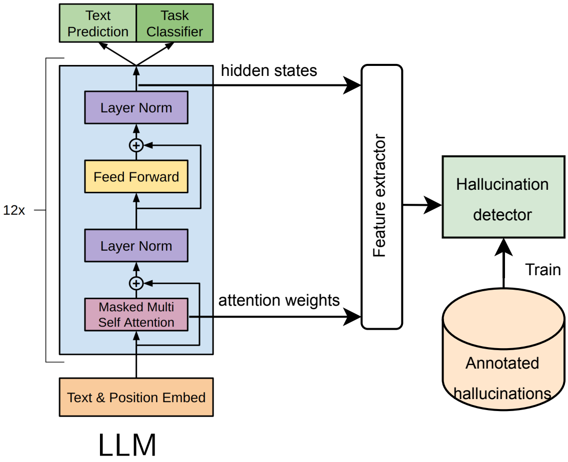


Intuition: Let's bypass all UQ challenges by learning from the data!

1. Create features from internal states and/or attention weights of the LLM.
2. Collect training data: hallucinations / non-hallucinations.
3. Train a supervised hallucination detector.

Question: How is it different with an external fact-checking tool?

From manual engineering to data-driven methods



Intuition: Let's bypass all UQ challenges by learning from the data!

1. Create features from internal states and/or attention weights of the LLM.
2. Collect training data: hallucinations / non-hallucinations.
3. Train a supervised hallucination detector.

Question: How is it different with an external fact-checking tool?

Answer: External fact-checker uses LLM outputs, supervised uncertainty quantification uses LLM internal states.

Supervised UQ methods

- UQ methods based on hidden states:
  - SAPLMA (Azaria and Mitchell 2023)
  - Factoscope (He et al. 2024)
  - Span-level hallucination detection (Ma 2025)
  - MIND (Su et al. 2024)
  - "The Curious Case of Hallucinatory (Un)answerability" (Slobodkin et al. 2023)
- **Key problem:** All methods based on hidden states suffer from overfitting to a particular domain
- **Key idea:** usage of only attention weights allows to partially mitigate the problem:
  - Lookback lens (Chuang et al. 2024)
  - TAD (Vazhentsev et al. 2025c)
  - UHead (Shelmanov et al. 2025)

Supervised methods and OOD generalization I

	Method \ Domain	Biographies (in domain)	Cities	Movies	Inventions	Books	Artworks	Landmarks	Events
Unsup.	MSP	.412	.310	.205	.319	.145	.317	.135	.141
	Max Token Entropy	.416	.289	.241	.381	.171	.321	.141	.161
	CCP	.496	.368	.267	.380	.167	.382	.196	.171
Supervised	SAPLMA	.536	.435	.269	.350	.292	.534	**.350**	.235
	Lookback lens	.557	.449	.254	.391	.259	.464	.257	.295
	UHead	**.660**	**.487**	**.466**	**.485**	**.395**	**.561**	.340	**.369**

Table: PR-AUC of claim-level hallucination detection. The supervised method is trained only on English *biographies* data.

- Performance gains for supervised UQ methods over unsupervised might be huge: up to 20 percentage points of PR-AUC.
- Supervised hallucination detectors might achieve good generalization to other domains: you can train on one domain and generalize to another.

Supervised methods and OOD generalization II

	Language / Method	English (in domain)	Russian	Chinese	German
Unsup.	MSP	.180	.433	.307	.203
	Max Token Entropy	.202	.437	.444	.217
	CCP	.307	.493	.439	.306
Supervised	SAPLMA	.342	.514	.331	<u>.391</u>
	Lookback lens	<u>.359</u>	<u>.576</u>	<u>.479</u>	.390
	UHead	**.457**	**.581**	**.556**	**.455**

Table: PR-AUC of claim-level hallucination detection. Supervised detectors were trained only on English *biographies* data.

- Supervised hallucination detectors achieve good generalization across languages: e.g. you can train on English and detect hallucinations in German generations.

Supervised methods and OOD generalization III

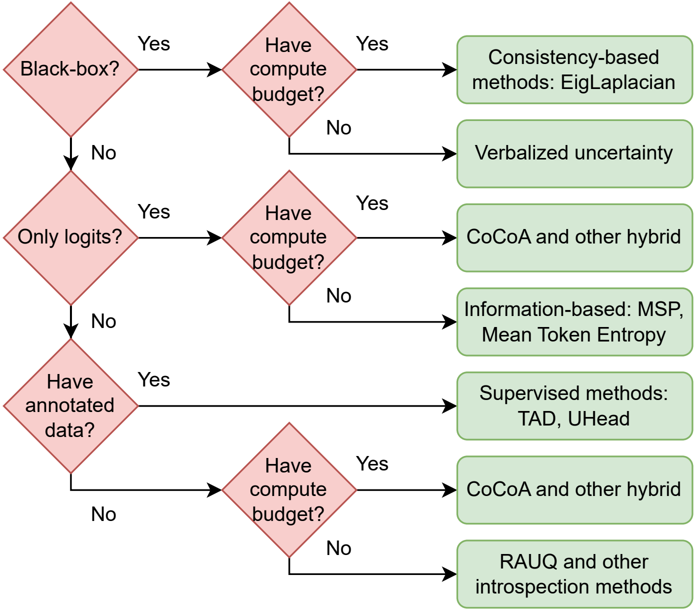
	Task / Method	SamSum	CNN	WMT19	Mean PRR
	MSP (unsupervised)	**.298**	**.157**	**.569**	**.342**
Supervised	Factoscope	.077	.023	.131	.077
	SAPLMA	.045	.021	-.250	-.061
	MIND	.077	.048	.174	.099
	Sheeps	.104	-.021	.157	.080
	Lookback Lens	-.026	-.032	.018	-.013
	TAD	.035	.003	.234	.091

Table: PRR for sequence-level selective generation. Supervised methods are trained on the QA task.

- Supervised methods generalize poorly across different tasks: you cannot train on QA and expect generalization to MT.

4. Conclusion & Future Work

Summary


Black-box?
Yes
Have compute budget?
Yes
Consistency-based methods: EigLaplacian
No
Verbalized uncertainty
No
Only logits?
Yes
Have compute budget?
Yes
CoCoA and other hybrid
No
Information-based: MSP, Mean Token Entropy
No
Have annotated data?
Yes
Supervised methods: TAD, UHead
No
Have compute budget?
Yes
CoCoA and other hybrid
No
RAUQ and other introspection methods

Emerging trends

- Improving performance of reasoning LLMs
  - Best-of-n reasoning trajectory selection
  - Truncate reasoning once sufficient confidence in the answer is achieved
  - Uncertainty-aware adaptive guidance
- Mitigating hallucinations in LLMs by intervening in the decoding process
- Teaching LLMs to express uncertainty
- Reinforcement learning from uncertainty-based feedback
- Epistemic and aleatoric uncertainty decomposition for LLMs

Final takeaways

- ML models are always imperfect due to limited amount of training data or ambiguity of the task.
- UQ is a crucial component of ML systems including LLMs.
- It is important to distill key ideas and perform well grounded research.
- UQ is a fascinating research direction that might be fruitful not only in terms of safety, but also improve reasoning LLMs and AI agents.

5. LM-Polygraph

LM-Polygraph: Uncertainty quantification framework for LLMs

- Python library: **40+ SOTA UQ methods** with unified program API
- Built-in routines for **normalization / calibration**
- Works with both **textual & visual** LLMs
- Integrates with
  - Open-weight LLMs from **HuggingFace**: Llama, Gemma, Qwen, DeepSeek, etc.
  - Models deployed via **vLLM**
  - **LLMs as a service**: ChatGPT, Gemini, OpenRouter, etc.
- **Extendable benchmark** covering MT, QA, summarization, language understanding, fact-checking, reasoning

GitHub:

License:

LM-Polygraph: High-level code example

```
from lm_polygraph import estimate_uncertainty
from lm_polygraph.model_adapters import WhiteboxModel
from lm_polygraph.estimators import *

model = WhiteboxModel.from_pretrained(
    "meta-llama/Llama-3.1-8B-Instruct",
    device_map="cuda:0"
)

ue_method  = MeanTokenEntropy()
input_text = "Who is George Bush?"
estimate_uncertainty(model, ue_method, input_text)
```

Thank you!

ACL 2025 Tutorial:

LM-Polygraph Github:

TACL paper:

References I

- Azaria, Amos and Tom Mitchell (2023). "The Internal State of an LLM Knows When It's Lying". In: *Findings of EMNLP 2023*.
- Chen, Chao et al. (2024). "INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection". In: *ICLR 2024*.
- Chuang, Yung-Sung et al. (2024). "Lookback Lens: Detecting and Mitigating Contextual Hallucinations in Large Language Models Using Only Attention Maps". In: *EMNLP 2024*.
- Fadeeva, Ekaterina et al. (2024). "Fact-Checking the Output of Large Language Models via Token-Level Uncertainty Quantification". In: *Findings of ACL 2024*.
- He, Jinwen et al. (2024). "Llm factoscope: Uncovering llms' factual discernment through measuring inner states". In: *Findings of the Association for Computational Linguistics ACL 2024*.

References II

- Hüllermeier, Eyke and Willem Waegeman (2021). "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods". In: *Machine learning*.
- Kuhn, Lorenz, Yarin Gal, and Sebastian Farquhar (2023). "Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation". In: *ICLR 2023*.
- Lee, Kimin et al. (2018). "A simple unified framework for detecting out-of-distribution samples and adversarial attacks". In: *NeurIPS 2018*.
- Ma, Yiran Rex (2025). "Do Androids Question Electric Sheep? A Multi-Agent Cognitive Simulation of Philosophical Reflection on Hybrid Table Reasoning". In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*.
- Malinin, Andrey and Mark J. F. Gales (2021). "Uncertainty Estimation in Autoregressive Structured Prediction". In: *ICLR 2021*.

References III

- Shelmanov, Artem et al. (2025). “A Head to Predict and a Head to Question: Pre-trained Uncertainty Quantification Heads for Hallucination Detection in LLM Outputs”. In: *arXiv preprint arXiv:2505.08200*.
- Slobodkin, Aviv et al. (Dec. 2023). “The Curious Case of Hallucinatory (Un)answerability: Finding Truths in the Hidden States of Over-Confident Large Language Models”. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.
- Sriramanan, Gaurang et al. (2024). “LLM-Check: Investigating Detection of Hallucinations in Large Language Models”. In: *NeurIPS 2024*.
- Su, Weihang et al. (Aug. 2024). “Unsupervised Real-Time Hallucination Detection based on the Internal States of Large Language Models”. In: *Findings of the Association for Computational Linguistics: ACL 2024*.
- Vashurin, Roman et al. (2025a). “UNCERTAINTY-LINE: Length-Invariant Estimation of Uncertainty for Large Language Models”. In: *EMNLP 2025*.

References IV

- Vashurin, Roman et al. (2025b). “Uncertainty Quantification for LLMs through Minimum Bayes Risk: Bridging Confidence and Consistency”. In: *NeurIPS 2025*.
- Vazhentsev, Artem et al. (2025a). “Token-Level Density-Based Uncertainty Quantification Methods for Eliciting Truthfulness of Large Language Models”. In: *NAACL 2025*.
- Vazhentsev, Artem et al. (2025b). “Uncertainty-Aware Attention Heads: Efficient Unsupervised Uncertainty Quantification for LLMs”. In: *arXiv preprint arXiv:2505.20045*.
- Vazhentsev, Artem et al. (2025c). “Unconditional Truthfulness: Learning Conditional Dependency for Uncertainty Quantification of Large Language Models”. In: *EMNLP 2025*.
- Zhang, Tianhang et al. (2023). “Enhancing Uncertainty-Based Hallucination Detection with Stronger Focus”. In: *EMNLP 2023*.