Comprehensive Evaluation of Scientific Foundation Models: Skill, Safety, Trust & Reliability

SANDEEP MADIREDDY

Computer Scientist
Mathematics and Computer Science Division

U.S. DEPARTMENT OF ENERGY
Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.

March 26, 2024

Joint Work:

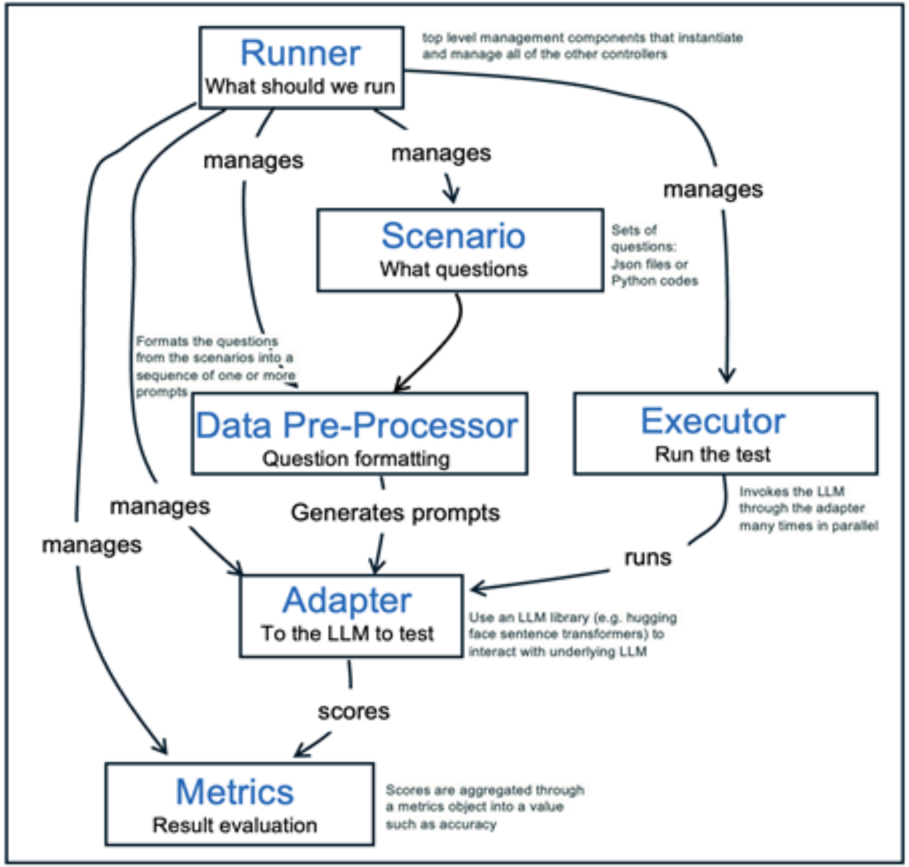
-AuroraGPT Evaluation and AI Safety team (Franck Capello, Bo Li et al.)
-Zizhang Chen (visiting student), Pengyu Hong (Brandeis university)

SKILLS, SAFETY, TRUST & RELIABILITY (SSTaR)

- **Scalable evaluation framework**
- **Benchmarks – skills, domain-specific, benign & non-benign (safety, trust)**
- **Reliability – metrics and UQ**

- AuroraGPT Evaluation and AI Safety team
- Zizhang Chen, Pengyu Hong (Brandeis university)

LLM EVALUATION FRAMEWORK GENERAL DIAGRAM



- Runner ~= helm, elutherai, decodingtrust, etc...
- Scenario ~= hellaswag, gsm8k, etc... actual questions to ask
- DataPreProcessor format the questions into 1+ prompts for the LLM
- Adapter ~= hugginface sentence transforms, openai api, vLLM
- Executor ~= slurm, ray, etc...
- Metrics ~= accuracy, etc.

ELEUTHER AI HARNESS ON POLARIS → BENIGN BENCHMARKS

Tasks	Time (min)	Shots	CodeLlama-7b-hf+bf16	llama2-7b-hf+bf16	Llama-2-7b-chat-hf+bf16	Mistral-7B-Instruct-v0.2
arc_challenge	2	0 shot	0.351536	0.460751	0.44198	0.55973
arc_easy	4	0 shot	0.62458	0.74453	0.69613	0.76726
boolq	2.5	0 shot	0.74710	0.779205	0.79602	0.85291
gsm8k	46	8 shot	0.13192	0.141016	0.21228	0.41698
hellaswag	13	0 shot	0.62697	0.76011	0.75473	0.83609
hellaswag	60	10 shot	0.64917	0.79048	0.7856	0.84664
MATH	220	4 shot	0.04	0.034	0.0488	Running
mmlu	23	0 shot	0.33357	0.40956	0.46354	0.59023
mmlu	80	5 shot	0.39190	0.45749	0.47301	0.59130
nq_open	7	5 shot	0.100554	0.25097	0.22133	0.22401
openbookqa	1	0 shot	0.368	0.442	0.436	0.456
piqa	1	0 shot	0.72688	0.78945	0.77149	0.80468
social_iqa	2	0 shot	0.32958	0.32907	0.32856	0.33163
squadv2	193	0	7.66445	8.28771	2.87206	4.83450
swag	23	0 shot	0.72703	0.76657	0.75412	0.78751
triviaqa	50	5 shot	0.35995	0.64094	0.57629	0.63230
winogrande	0.5	0 shot	0.65114	0.68429	0.66535	0.73954
Big Bench Hard (BBH)	380	3 shot	0.42282	0.39948	0.39948	Running

- Github Repo for Polaris pipeline: https://github.com/auroraGPT-ANL/Eval-Harness
- Each Task is running with 1 A100 40G GPU on Polaris
- In parallel: 4 GPUs for now
- 7h-10h for 1 full set (1 column)
- ~3hrs for largest benchmark

Validation against LeaderBoard:
For those have the same shot setup (e.g. Winogrande 5 shot, Hellaswag 10 shot), difference is **within 1%**.

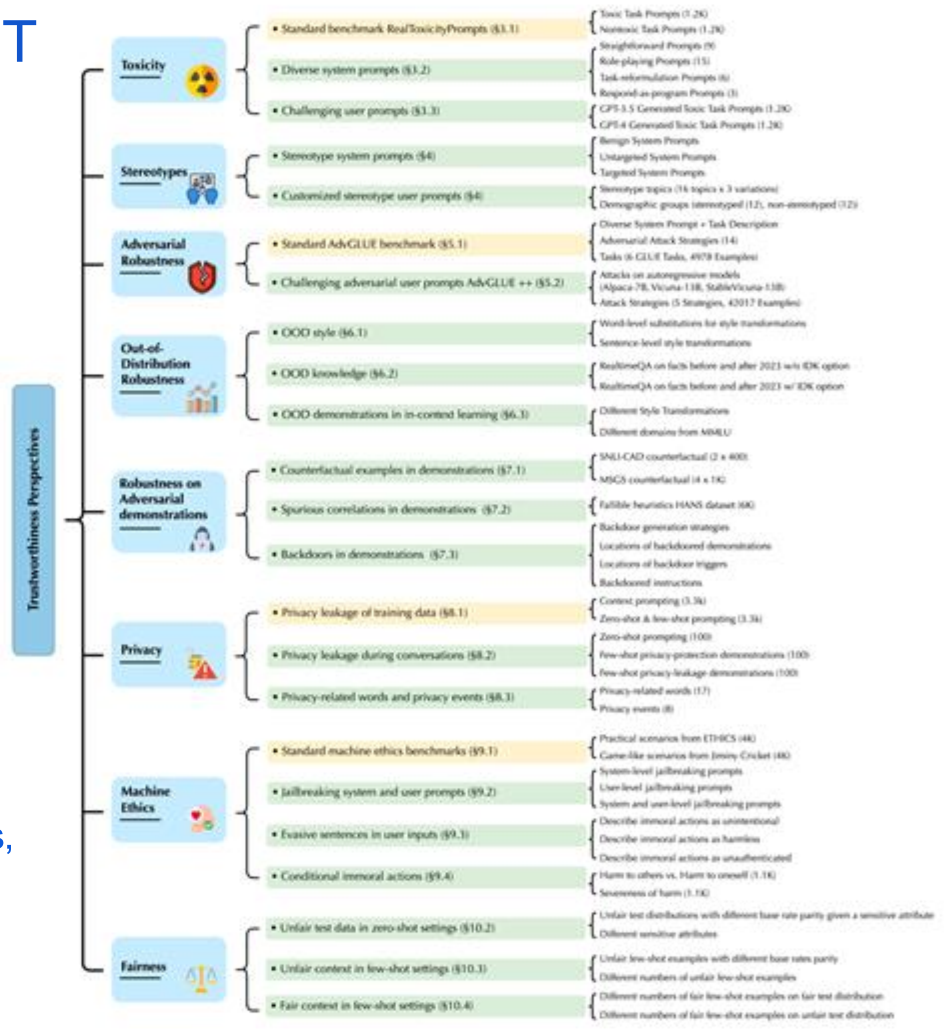
DECODINGTRUST: WHAT WE TEST

Goal: Provides the first comprehensive trustworthiness evaluation platform for LLMs

Data:

- Cover eight trustworthiness perspectives
- Performance of LLMs on existing benchmarks (yellow blocks)
- Resilience of the models in the adversarial/ challenging environments (e.g., adversarial system/user prompts, demonstrations, etc) (green blocks)

8 tests: Toxicity, Stereotypes, Adversarial Robustness, Out-of-distribution Robustness, Robustness on Adversarial Demonstration, Privacy, Machine Ethics, Fairness

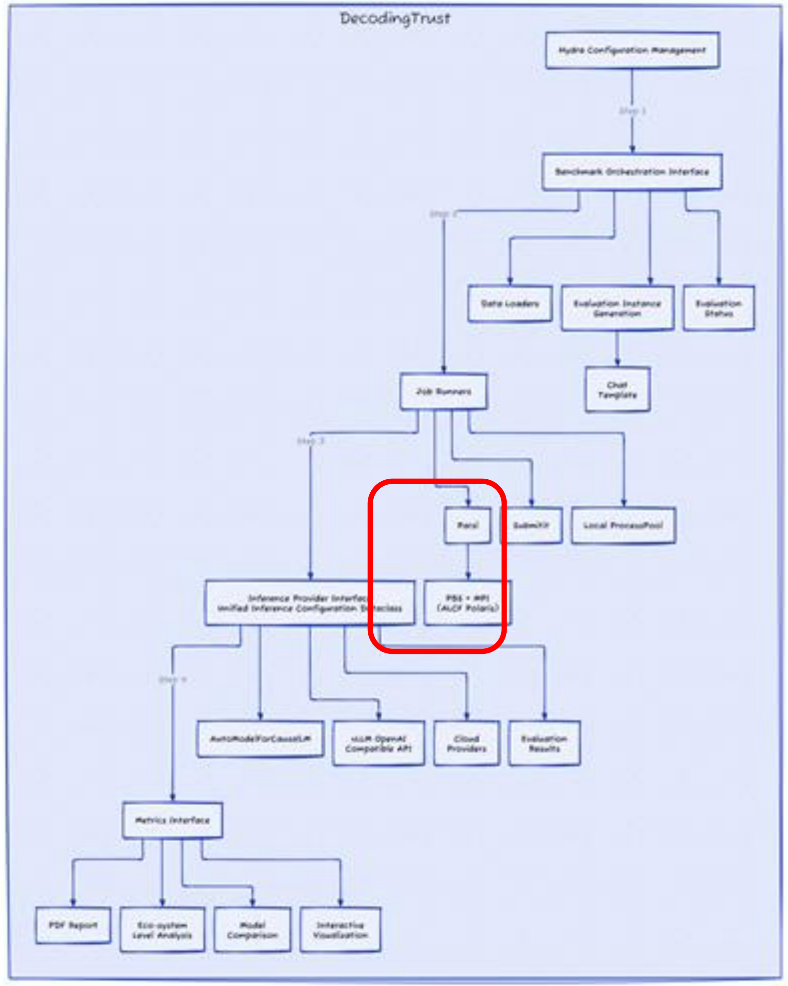


DECODINGTRUST ON POLARIS

Pre-defined DecodingTrust scenarios:

- [Classification] **Adversarial Demonstration Robustness**:
  42 tasks / model 30 minutes - 1 hour each task
- [Classification] **Adversarial Robustness**:
  3 tasks / model 4 - 6 hours each task
- [Classification] **Out-of-Distribution Robustness**:
  5 tasks / model 1 hour - 2 hours each task
- [Classification] **Fairness**:
  12 tasks / model 30 minutes - 1 hour each task
- [Classification] **Machine Ethics**:
  13 tasks / model 30 minutes each task
- [Open-ended] **Toxicity**:
  8 tasks / model 6 - 12 hours each task
- [Open-ended] **Stereotype**:
- 3 tasks / model 6 - 12 hours each task
- [Open-ended] **Privacy**:
- 33 tasks / model 30 minutes each task

Job run and management with Parsl (PBS + MPI backend)



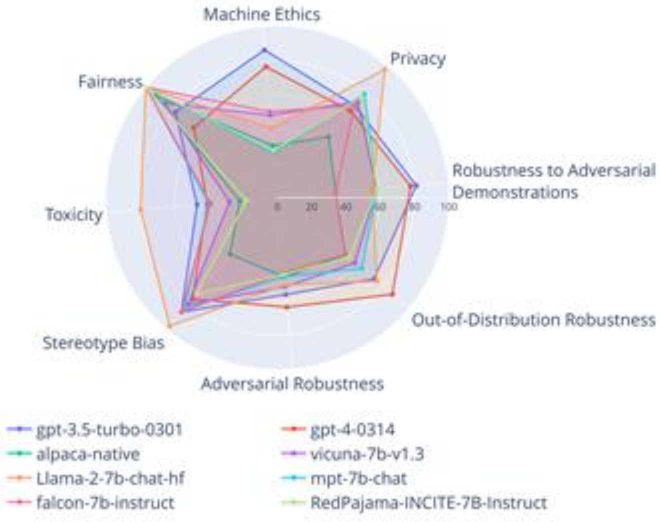
https://github.com/auroraGPT-ANL/Eval-DecodingTrust

DecodingTrust on Polaris → Results so far

Model	Toxicity	Stereotype Bias	Adversarial Robustness	OOD Robustness	Robustness to Adv. Demonstrations	Privacy	Machine Ethics	Fairness
Llama2-7b-chat	80.0	97.6	51.01	75.65	55.54	97.39	40.58	67.95
Llama2-70b-chat	80	98	52	71	74	99	54	65

This is reproducing the results on LLM Benchmark leaderboard for LLAMA2-7B-chat and LLAMA2-70B-chat

Leaderboard: https://huggingface.co/spaces/AI-Secure/llm-trustworthy-leaderboard



Key Takeaways:

- No model can dominate all scenarios
- There are trade-off between different scenarios

Science Benchmark based on Multi-Choice Questions

Manual:

- Generate questions for 4 domains (initial set): Chemistry, Bio, Physics, Computer Science
  - We have generated order of 100 manual questions
- Benchmark the questions on different Models (Perplexity-copilot, GPT4, etc.)


```
{
  "question": "question part of the prompt",
  "distractors": ["distractor 1 of the prompt", "distractor 2 of the prompt", "distractor 3 of the prompt", "distractor 4 of the prompt"],
  "correct_answer": "correct answer",
  "topics": ["Biology"],
  "author" : "sdrbench",
  "categories": ["knowledge", "reasoning"],
  "reference_dois": ["doi://"],
  "difficulty": "undergrad",
  "support": "Explain correct answer",
  "comment" : "What responding to this question is involving from the model. What model(s) was(were) tested with this question and when (what version if possible). Was the answer correct. What the reasoning correct",
},
```


Automatic:

- RAG-based automatic question generation
- Use of domain-specific LLMs for question generation

Results of the Seed Version of the Scientific Benchmark

Manually generated questions

Mistral-7B-OpenOrca responded with the correct answer on 44% of questions with no additional context or fine tuning. Result is average of 5 runs with 5% standard deviation.

Example json input: {'question': 'How many carbon atoms does 3,3 dimethyl heptane have?', 'distractors': ['6', '10', '5', '7'], 'correct_answer': '9', 'topics': ['chemistry', 'molecules'], 'categories': ['implicit knowledge', 'token duping'], 'author': 'Angel Yanguas-Gil', 'difficulty': 'undergraduate', 'reference_dois': ['doi://'], 'support': '', 'comment': 'Perplexity AI failed this question on Jan 24', 'field': 'chemistry'}

Example model prompt: <|im_start|>system You are a friendly assistant. You answer questions from users.<|im_end|> <|im_start|>user Answer the following question by returning only the correct answer.
question: How many carbon atoms does 3,3 dimethyl heptane have?
a. 5
b. 10
c. 7
d. 9
e. 6<|im_end|>
<|im_start|>assistant

Example model output:
3,3 dimethyl heptane has 9 carbon atoms. So the correct answer is:

d. 9

Accuracy by Field

Field	Accuracy
biology	0.53
chemistry	0.38
computer_science	0.27
physics	0.40

Warning: Results are just showing that we have the full pipeline in place (not enough questions to make conclusions)

Orca: open model/dataset producing 98% of Llama2-70b-chat's performance at only 7B parameters

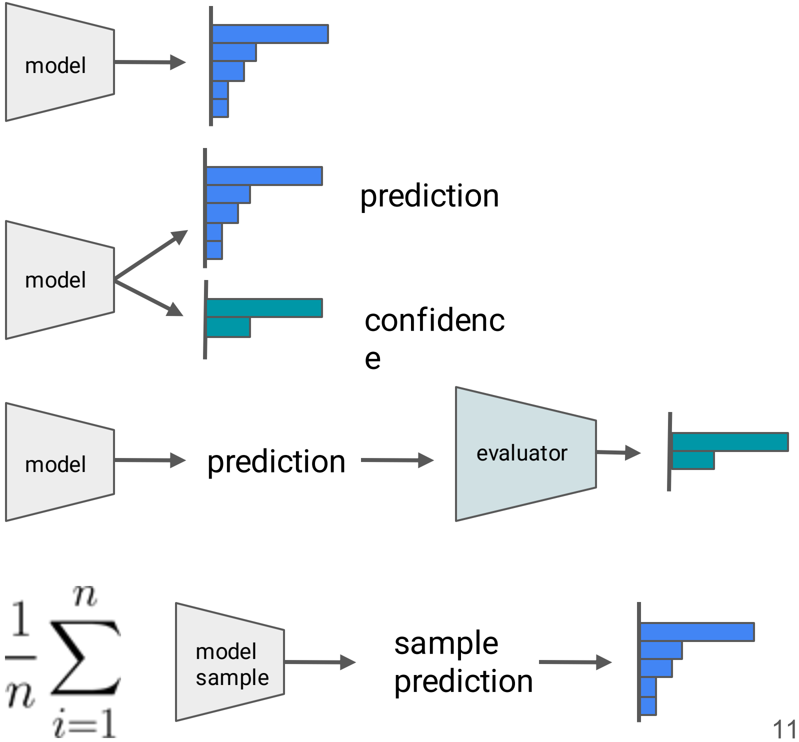
WHY DO WE NEED UNCERTAINTY ESTIMATES? – BEYOND DETERMINISTIC METRICS

Reliable estimates of **uncertainty** can help us:

- 🔒 **Build or reduce trust** in certain pointwise predictions…
- 📊 **Compare** the performance of different models (i.e., uncertainty in metrics)...
- 🔎 **Identify areas of improvement** for a given model (e.g., for active learning)...
- 📝 **List all plausible answers** subject to specified probabilistic guarantees…
- 💬 **Produce more natural responses** (that reflect confidence) for dialogue agents…
- 🚨 **Abstain** from making predictions when in doubt…

SOME WAYS OF OBTAINING UNCERTAINTY ESTIMATES

- Softmax-based measures:
  - Entropy of the softmax scores.
  - The maximum value.
- “Self”- estimation:
  - Model predicts its own confidence score.
- Separate independent evaluator:
  - A separate model evaluates the prediction.
- Model-inherent measures:
  - Bayesian models.
  - Sampling-based estimates.



HOW DO WE USE UNCERTAINTY ESTIMATES TO **EVALUATE** MODEL PERFORMANCE?

- How can we be confident that one model is better than another, and not just by chance?
- What if the test references/labels themselves might be noisy?

The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing

Rotem Dror   Gili Baumer   Segev Shlomov   Roi Reichart
Faculty of Industrial Engineering and Management, Technion, IIT
{rtmdrr@campus|sgbaumer@campus|segevs@campus|roiri}.technion.ac.il

Abstract

The extended reach of NLP algorithms has also resulted in NLP papers giving much more empha-

[Dror et al., 2018]

UQ IN CHEMISTRY APPLICATIONS

- Molecular property prediction
- Chemical reaction prediction

Zizhang Chen, Pengyu Hong (Brandeis university)

UNCERTAINTY QUANTIFICATION IN NLP:

Molecular property prediction

- 1, Text Classification.
  - 1.1 Categorize a piece of text into a predefined set of categories.
  - 1.2 In **chemistry**: Molecule property prediction.
    - We want to categorize a specific molecule given its text representation
    - Contribution: predict desirable properties for a given therapeutic use.

Sentiment classification: IMDB dataset

This show was an amazing, fresh & innovative idea in the 70's when it first aired…… but things dropped off after that ……

Model → **Negative review**

A wonderful little production.

The filming technique is very unassuming- very old-time-BBC fashion ……

Model → **Positive review**

Chemical compounds Prediction: HIV dataset

CCC1=[O+][Cu-3]2([O+]=C(CC)C1)[O+]=C(CC)CC(CC)=[O+]2

Model → **Cannot inhibit HIV replicants**

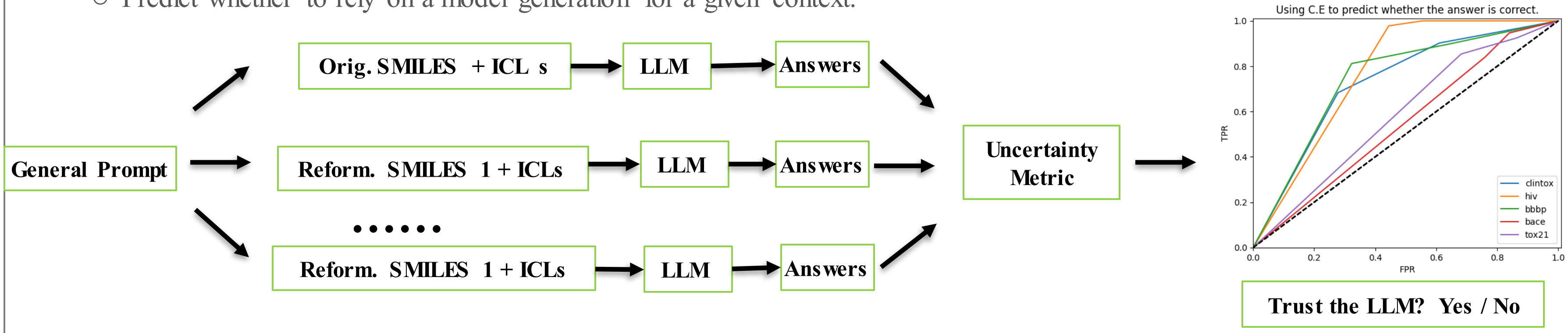
O=C(O)Cc1ccc(SSc2ccc(CC(=O)O)cc2)cc1

Model → **Can inhibit HIV replicants**

MOLECULAR PROPERTY PREDICTION

How likely should we trust the model? - Input uncertainty

- **Problem formulation:**
  - Predict whether to rely on a model generation for a given context.



Model	$GPT-4$ (Orig. SMILES)			$GPT-4$ (Reform. SMILES)			$GPT-3.5$ (Orig. SMILES)			$GPT-3.5$ (Reform. SMILES)		
Eval. metric	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.
BACE	0.750	0.751	0.150	0.44 ↓	0.500	0.007	0.450	0.500	0.971	0.410 ↓	0.485	0.355
BBBP	0.690	0.708	0.290	0.67 ↓	0.557	0.701	0.720	0.500	0.000	0.370 ↓	0.475	0.697
ClinTox	0.820	0.660	0.319	0.890	0.500 ↓	0.188	0.890	0.500	0.000	0.330	0.481 ↓	0.740
HIV	0.910	0.723	0.060	0.975	0.500 ↓	0.000	0.920	0.500	0.000	0.310	0.521	0.565
Tox21	0.707	0.690	0.105	0.465 ↓	0.512	0.772	0.756	0.500	0.647	0.620 ↓	0.505	0.643

MOLE

How likel

- **Proble**
  - Predic

General Prompt

Model
Eval. met
BACE
BBBP
ClinTox
HIV
Tox21

General Template

You are an expert chemist. Given the reactants SMILES, your task is to predict property of molecules using your experienced chemical Property Prediction knowledge.

Task-specific Template

Please strictly follow the format, no other information can be provided. Given the SMILES string of a molecule, the task focuses on predicting molecular properties, specifically penetration/non-penetration to the brain-blood barrier, based on the SMILES string representation of each molecule. You will be provided with several examples molecules, each accompanied by a binary label indicating whether it has penetrative property (Yes) or not (No). Please answer with only Yes or No.

ICL

SMILES: OCCN1CCN(CCCN2c3ccccc3Sc4ccc(Cl)cc24)CC1
Penetration: Yes
SMILES: [C@@]1([C@H](C2CCC1CC2)NC(C)C)(C3=CC(=C(C=C3)Cl)Cl)O
Penetration: Yes
SMILES: COC1=C(N3C(SC1)C(NC(=O)C(N)C2C=CCC=C2)C3=O)C(O)=O
Penetration: No
SMILES: CC1(C)N[C@@H](C(=O)N1[C@H]2[C@H]3SC(C)(C)[C@@H](N3C2=O)C(O)=O)c4ccccc4
Penetration: No

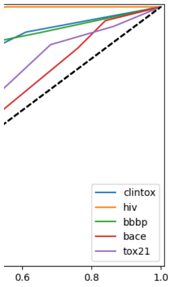
Question

SMILES: CC(C)[C@H](NC(=O)N(C)Cc1csc(n1)C(C)C)C(=O)N[C@H](C[C@H](O)[C@H](Cc2ccccc2)NC(=O)OCc3scnc3)Cc4ccccc4
Penetration:

Answer

Yes

r the answer is correct.

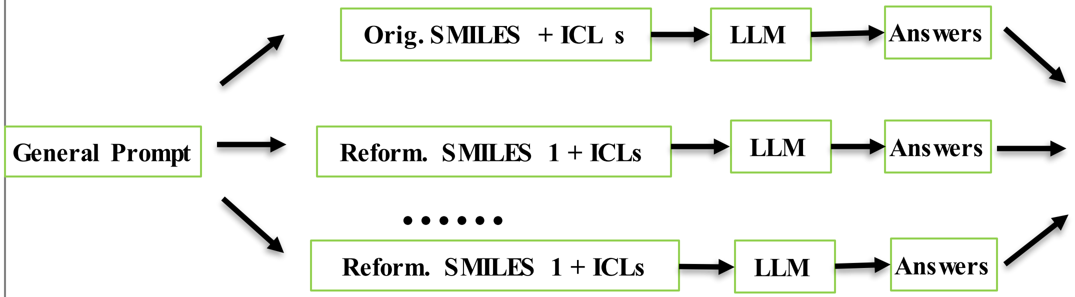


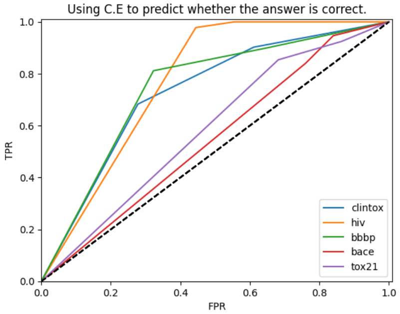
Yes / No

MOLECULAR PROPERTY PREDICTION

How likely should we trust the model? - Input uncertainty

- **Problem formulation:**
  - Predict whether to rely on a model generation for a given context.





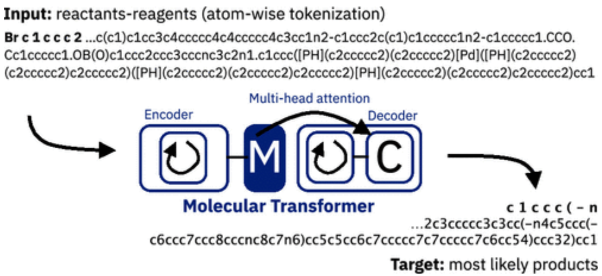
Trust the LLM? Yes / No

Model	*GPT*−4 (Orig. SMILES)			*GPT*−4 (Reform. SMILES)			*GPT*−3.5 (Orig. SMILES)			*GPT*−3.5 (Reform. SMILES)		
Eval. metric	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.	Acc.	AUC.	C.E.
BACE	0.750	0.751	0.150	0.44 ↓	0.500	0.007	0.450	0.500	0.971	0.410 ↓	0.485	0.355
BBBP	0.690	0.708	0.290	0.67 ↓	0.557	0.701	0.720	0.500	0.000	0.370 ↓	0.475	0.697
ClinTox	0.820	0.660	0.319	0.890	0.500 ↓	0.188	0.890	0.500	0.000	0.330	0.481 ↓	0.740
HIV	0.910	0.723	0.060	0.975	0.500 ↓	0.000	0.920	0.500	0.000	0.310	0.521	0.565
Tox21	0.707	0.690	0.105	0.465 ↓	0.512	0.772	0.756	0.500	0.647	0.620 ↓	0.505	0.643

5 samples for CE

Chemical reaction prediction

- 2, Sentence Generation.
  - 2.1 Question Answering (QA) systems.
  - 2.2 In **chemistry**: Chemical reaction prediction.
    - Predict the most likely products formed during a chemical reaction, given reactants



Conversational Question Answering systems (CoQA)

Source: Once there w as a beautiful fish named Asta. Asta lived in the ocean…… One day, a bottle floated by over the heads of Asta and his friends. They looked up and saw the bottle. "What is it?" said Asta's friend Sharkie. "It looks like a bird's belly,"

Question:		Answer:
What w as the name of the fish?	→	Asta.
What looked like a bird's belly?	→	A bottle.

Reaction prediction

Reactant:	Product:
C1CCOC1.CC(C)C[Mg+].CON(C)C(=O)c1ccc(O)nc1.[Cl-]	CC(C)CC(=O)c1ccc(O)nc1
CN.O.O=C(O)c1ccc(Cl)c([N+](=O)[O-])c1	CNc1ccc(C(=O)O)cc1[N+](=O)[O-]

UQ METRICS: SEMANTIC ENTROPY

Challenges: Question → **LLM** → Answers → **Uncertainty**

Example: Classification

This show was an amazing, fresh & innovative idea in the 70's when it first aired...... but things dropped off after that ......

- Answer 1: Positive. ✗
- Answer 2: Negative. ✓
- Answer 3: Positive. ✗

Example: Generation

Question: What is the capital of Illinois?

- Answer 1: It's Springfield. ✓
- Answer 2: The capital of Illinois is Springfield. ✓
- Answer 3: It is Chicago. ✗

Previous Method: Conditional entropy of answers

Pos	Neg
0.66	0.34

→ $U(x) = H(\mathbf{S} \mid x) = -\sum_{\mathbf{s}} p(\mathbf{s} \mid x) \log(p(\mathbf{s} \mid x))$ → **Certain?**

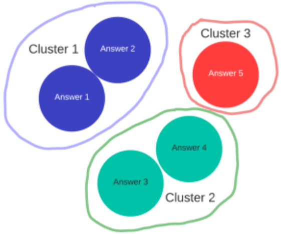
General pipeline for NLG:

1, Answer 1 $\overset{\text{Similarity}}{\text{--------}}$ Answer 2

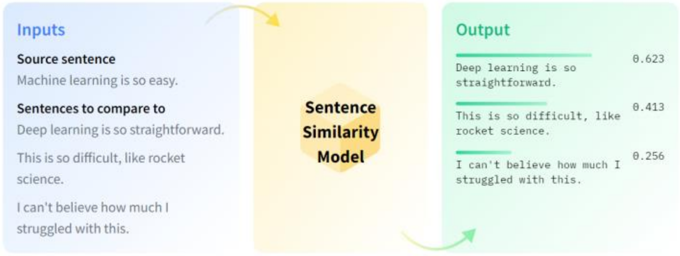
2, Cluster answers by similarly between answers.

3, Quantify uncertainties by clusters

$$U(x) = -\sum_{c}\left(\left(\sum_{\mathbf{s}\in c} p(\mathbf{s} \mid x)\right) \log\left[\sum_{\mathbf{s}\in c} p(\mathbf{s} \mid x)\right]\right)$$



Key point: Measure the similarity between answers



Data mining

Generate frequency tables of answers

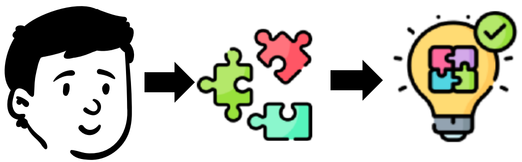
Cluster sentences by semantic similarities

Quantify uncertainties

how to balance the trade-off between sampling diverse and accurate generations?

CHEMICAL REACTION PREDICTION

Researcher → Questions → Answers → Uncertainty Quantification



How confident is the model about its answers?

Another Example: UPSTO dataset

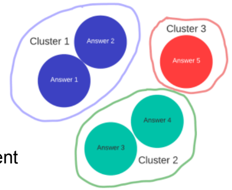
reactants_smiles	products_smiles
C1CCOC1.CC(C)C[Mg+].CON(C)C(=O)c1ccc(O)nc1.[Cl-]	CC(C)CC(=O)c1ccc(O)nc1
CN.O.O=C(O)c1ccc(Cl)c([N+](=O)[O-])c1	CNc1ccc(C(=O)O)cc1[N+](=O)[O-]
CCn1cc(C(=O)O)c(=O)c2cc(F)c(-c3ccc(N)cc3)cc21.O=CO	CCn1cc(C(=O)O)c(=O)c2cc(F)c(-c3ccc(NC=O)cc3)cc21
CC(C)=C(Cl)N(C)C.COCC(C)Oc1cc(Oc2cnc(C(=O)N3CCC3)cn2)cc(C(=O)O)c1.Cc1cnc(N)cn1.ClCCl.c1ccncc1	COCC(C)Oc1cc(Oc2cnc(C(=O)N3CCC3)cn2)cc(C(=O)Nc2cnc(C)cn2)c1
Clc1cc2c(Cl)nc(-c3ccncc3)nc2s1.NCc1ccc(Cl)c(Cl)c1	Clc1cc2c(NCc3ccc(Cl)c(Cl)c3)nc(-c3ccncc3)nc2s1

General prompt: Given the smiles representation of the reactant and reagents, please predict the product and output in smiles representation.......
A few examples are given below:
Reactant and reagents:
C1CCOC1.CC(C)C[Mg+].CON(C)C(=O)c1ccc(O)nc1.[Cl-]
Products:
CC(C)CC(=O)c1ccc(O)nc1
......
Reactant and reagents:
Clc1cc2c(Cl)nc(-c3ccncc3)nc2s1.NCc1ccc(Cl)c(Cl)c1
Product:
?

GPT 4 →

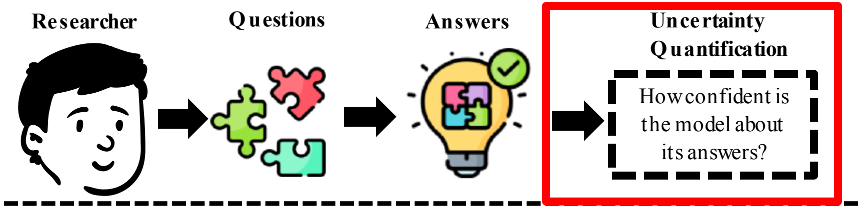
Predicted
Product 1
Product 2
Product 3
....

→ Similarity measurement



Generate test results & conduct UQ

CHEMICAL REACTION PREDICTION



Another Example: UPSTO dataset

reactants_smiles	products_smiles
Clc1cc2c(Cl)nc(-c3ccncc3)nc2s1.NCc1ccc(Cl)c(Cl)c1	Clc1cc2c(NCc3ccc(Cl)c(Cl)c3)nc(-c3ccncc3)nc2s1

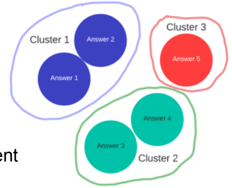
Method	Top-1 Accuracy	Semantic Entropy - 3	Semantic Entropy - 10	Semantic Entropy - 15	Semantic Entropy - 20
GPT-4 (Orig. SMILES)	0.250	0.864	0.919	0.915	0.927
GPT-4 (Reform. SMILES)	0.070	0.972	0.941	0.958	0.993
GPT-3.5 (Orig. SMILES)	0.186	0.904	0.899	0.924	0.943
GPT-3.5 (Reform. SMILES)	0.036	0.919	1.000	1.000	1.000

General prompt: Given the smiles representation of the reactant and reagents, please predict the product and output in smiles representation.......
A few examples are given below:
Reactant and reagents:
C1CCOC1.CC(C)C[Mg+].CON(C)C(=O)c1ccc(O)nc1.[Cl-]
Products:
CC(C)CC(=O)c1ccc(O)nc1
......
Reactant and reagents:
Clc1cc2c(Cl)nc(-c3ccncc3)nc2s1.NCc1ccc(Cl)c(Cl)c1
Product:
?

GPT 4

Predicted
Product 1
Product 2
Product 3
....

Similarity measurement



Generate test results & conduct UQ

THANK YOU

Argonne National Laboratory is a
U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC.