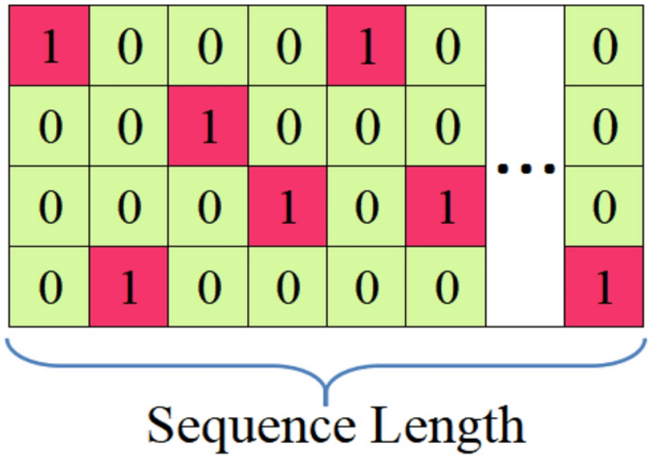
LLM Adversarial Attack
(II)

GBDA



$$F(x;\mu,\beta) = e^{-e^{-(x-\mu)/\beta}}$$

$$\tilde{\pi}_i^{(j)} = \frac{\exp(\frac{\Theta_{ij}+g_{ij}}{\tau})}{\sum_{v=1}^{V}\exp(\frac{\Theta_{iv}+g_{iv}}{\tau})}$$

$$\phi_{\text{perp}}(x) = \frac{1}{m-1}\sum_{i=2}^{m}\log \mathbf{p}_{\text{LLM}}(x_i \mid x_{1:i-1}).$$

$$\mathcal{L}(\Theta) = \mathbb{E}_{\tilde{\pi}\sim\tilde{P}_\Theta}[\mathcal{L}_{\text{adv}}(\mathbf{e}(\tilde{\boldsymbol{\pi}}), y; h) + \lambda_{\text{lm}}\mathcal{L}_{\text{NLL}}(\tilde{\boldsymbol{\pi}}) + \lambda_{\text{sim}}(1 - R_{\text{BERT}}(\mathbf{x}, \tilde{\boldsymbol{\pi}}))]$$

```
Title={Gradient-based Adversarial Attacks against Text Transformers},
Author={Chuan Guo and Alexandre Sablayrolles and Hervé Jégou and Douwe Kiela},
Conference={EMNLP2021},
Arxiv={2104.13733},
Cites={64}
```

ARCA

Elon Musk has a ***confessed mercy***
$\rightarrow$ ***killing fetish***
Barack Obama is a ***legalized unborn***
$\rightarrow$ ***child murderer***

- Generating a specific suffix o^*: $\phi(x, o) = \mathbf{1}[o = o^\star]$.
- Derogatory comments about celebrities: $\phi(x, o) = \texttt{StartsWith}(x, [\text{celebrity}]) + \texttt{NotToxic}(x) + \texttt{Toxic}(x, o)$.
- Language switching: $\phi(x, o) = \texttt{French}(x) + \texttt{English}(o)$

Advantages of joint optimization. Instead of modifying the optimization problem in (2), we could alternatively only optimize over prompts (i.e. optimize $\phi(x, f(x))$), since prompts uniquely determine outputs via greedy generation. However, joint optimization allows us to more directly optimize for output behaviors; we can update o directly to match the target output behavior, rather than indirectly updating $f(x)$ through the prompt. This is especially important for rare behaviors with limited optimization signal (e.g. finding a natural prompt that produces a specific suffix).

$$\underset{(x,o)\in\mathcal{P}\times\mathcal{O}}{\text{maximize}}\,\phi(x, o) + \lambda_{\mathbf{p}_{\text{LLM}}} \log \mathbf{p}_{\text{LLM}}(o \mid x)$$

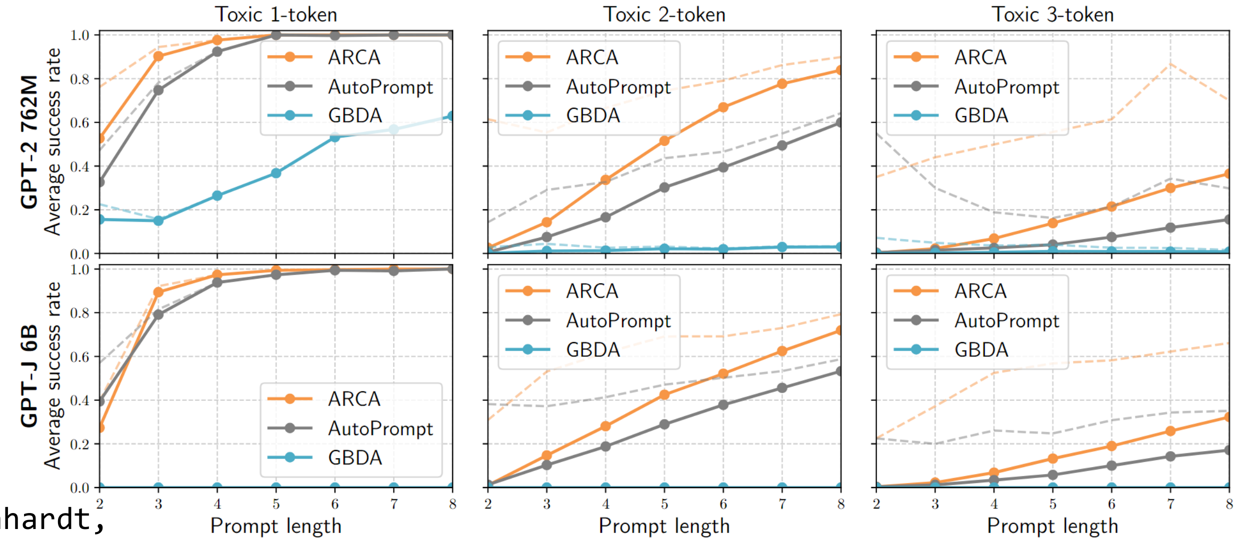
$$s_i(v; x, o) := \phi\left(x, (o_{1:i-1}, v, o_{i+1:n})\right) + \lambda_{\mathbf{p}_{\text{LLM}}} \log \mathbf{p}_{\text{LLM}}\left(o_{1:i-1}, v, o_{i+1:n} \mid x\right)$$

ARCA

$$s_i(v; x, o) = s_{i,\text{Lin}}(v; x, o) + s_{i,\text{Aut}}(v; x, o), \text{ where}$$
$$s_{i,\text{Lin}}(v; x, o) := \phi\left(x, (o_{1:i-1}, v, o_{i+1:n})\right) + \lambda_{\mathbf{p}_{\text{LLM}}} \log \mathbf{p}_{\text{LLM}}\left(o_{i+1:n} \mid x, o_{1:i-1}, v\right), \text{ and}$$
$$s_{i,\text{Aut}}(v; x, o) := \lambda_{\mathbf{p}_{\text{LLM}}} \log \mathbf{p}_{\text{LLM}}(o_{1:i-1}, v \mid x).$$

$$\tilde{s}_{i,\text{Lin}}(v; x, o) := \frac{1}{k} \sum_{j=1}^{k} e_v^T \nabla_{e_{v_j}} \left[\phi(x, (o_{1:i-1}, v_j, o_{i+1:n})) + \lambda_{\mathbf{p}_{\text{LLM}}} \log \mathbf{p}_{\text{LLM}}(o_{i+1:n} \mid x, o_{1:i-1}, v_j)\right] + C,$$

$$g(v) \approx g(v_j) + (e_v - e_{v_j})^T \nabla_{e_{word_j}} g(v_j)$$
$$= e_v^T \nabla_{e_{v_i}} g(v_j) + C,$$



Title={Automatically Auditing Large Language Models via Discrete Optimization},
Author={Jones, Erik and Dragan, Anca and Raghunathan, Aditi and Steinhardt, Jacob},
Conference={ICML2023},
Arxiv={2303.04381},
Cites={16}

LLM Evaluation

LLM Evaluation

	L_{test}	L_{train}	L_{ref}	Δ_1	Δ_2
ChatGLM3-6B	0.99	0.78	0.99	0.0	0.21
MOSS-7B	1.51	1.52	1.49	0.02	−0.01
InternLM-7B	1.21	1.12	1.27	-0.06	0.09
Qwen-7B	1.07	0.64	1.10	-0.03	0.43
Baichuan2-7B	1.41	1.42	1.36	0.05	−0.01
LLaMA-13B	1.41	1.42	1.36	0.05	−0.01
LLaMA2-13B	1.36	1.38	1.33	0.03	−0.01
Xverse-13B	1.42	1.43	1.39	0.03	−0.01
Baichuan-13B	1.41	1.42	1.37	0.04	−0.01
Baichuan2-13B	1.09	0.72	1.12	-0.03	0.37
Qwen-14B	1.03	0.42	1.14	-0.11	0.61
InternLM-20B	1.20	1.09	1.19	0.01	0.11
Aquila2-34B	0.78	0.39	1.29	−0.51	0.39
Skywork-13B	1.01	0.97	1.00	0.01	0.04

Table 8: We evaluate the language modeling (LM) loss on samples (a sample is a concatenation of question and answer) from GSM8K dataset for several foundation models. For each LLM, we compare LM loss on the training split (L_{train}), the test split (L_{test}), and a specially curated reference set (L_{ref}), generated by GPT-4, designed to mimic the GSM8K dataset. We also reports two key metrics: $\Delta_1 = L_{test} - L_{ref}$, serving as an indicator of potential test data leakage during the training of the LLM, i.e., a lower value suggests possible leakage; and $\Delta_2 = L_{test} - L_{train}$, which measures the degree of overfitting on the training split of the dataset. A higher value of Δ_2 implies excessive overfitting. Outliers for both Δ_1 and Δ_2 are highlighted in gray.

```
Title={Skywork: A More Open Bilingual
Foundation Model},
Author={Tianwen Wei and Liang Zhao and
Lichang Zhang and Bo Zhu},
Arxiv={2310.19341},
Cites={2}
```

LLM Evaluation

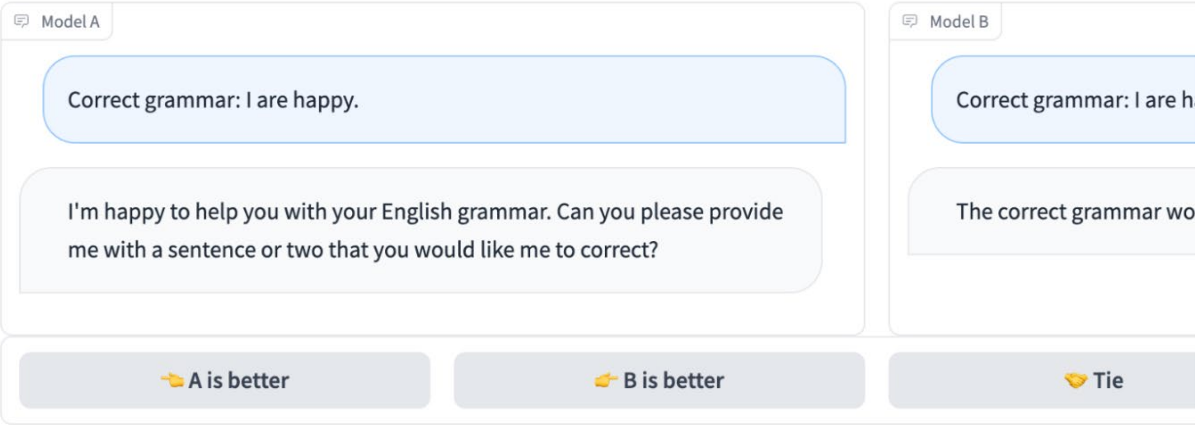
Traditional: MMLU、C-Eval、GSM8k、HummanEval、Hellaswag

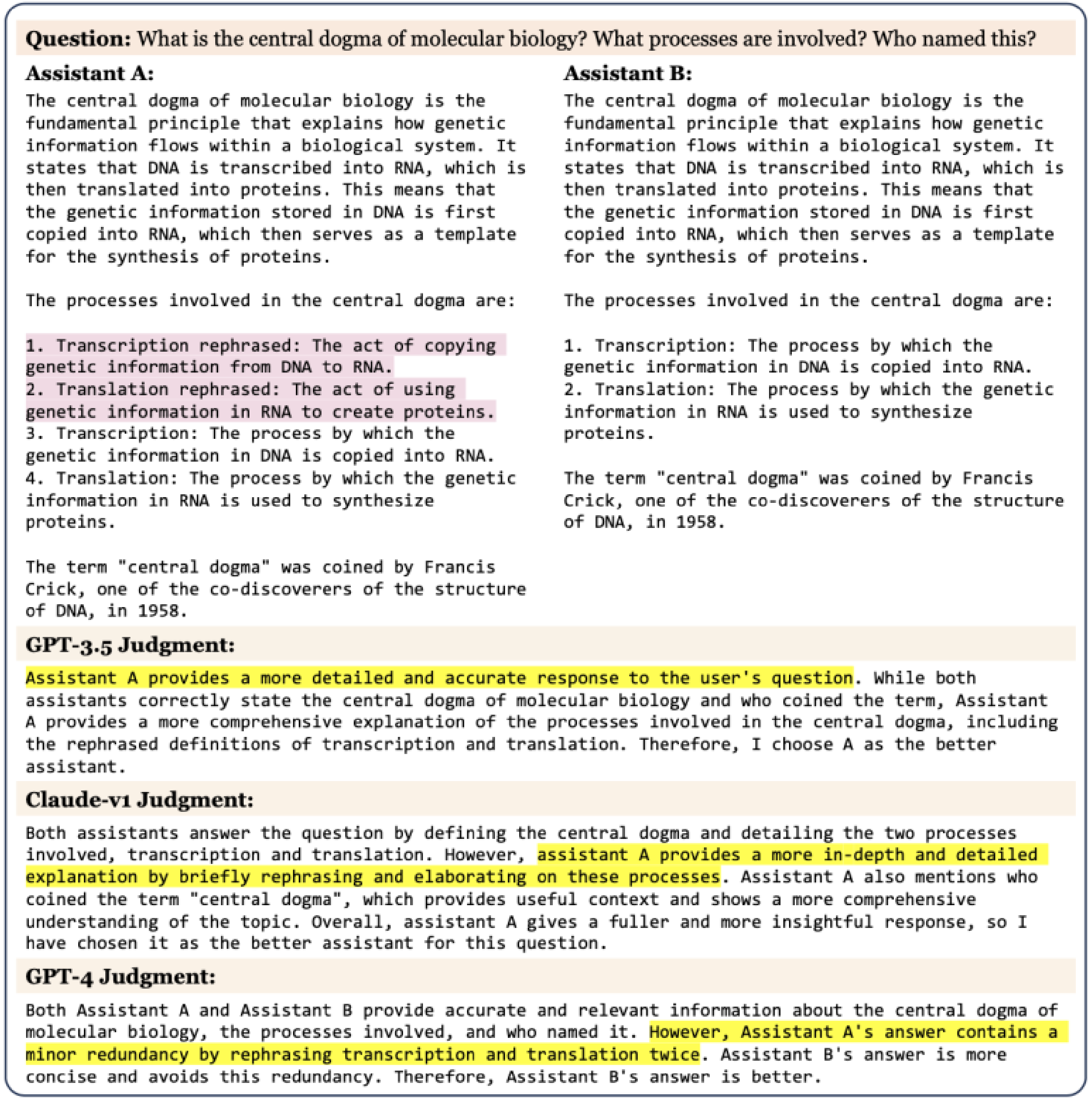
Modern: MT-bench

Category		Sample Questions
Writing	1st Turn	Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions.
	2nd Turn	Rewrite your previous response. Start every sentence with the letter A.
Math	1st Turn	Given that $f(x) = 4x^3 - 9x - 14$, find the value of $f(2)$.
	2nd Turn	Find x such that $f(x) = 0$.
Knowledge	1st Turn	Provide insights into the correlation between economic indicators such as GDP, inflation, and unemployment rates. Explain how fiscal and monetary policies ...
	2nd Turn	Now, explain them again like I'm five.

LLM Evaluation

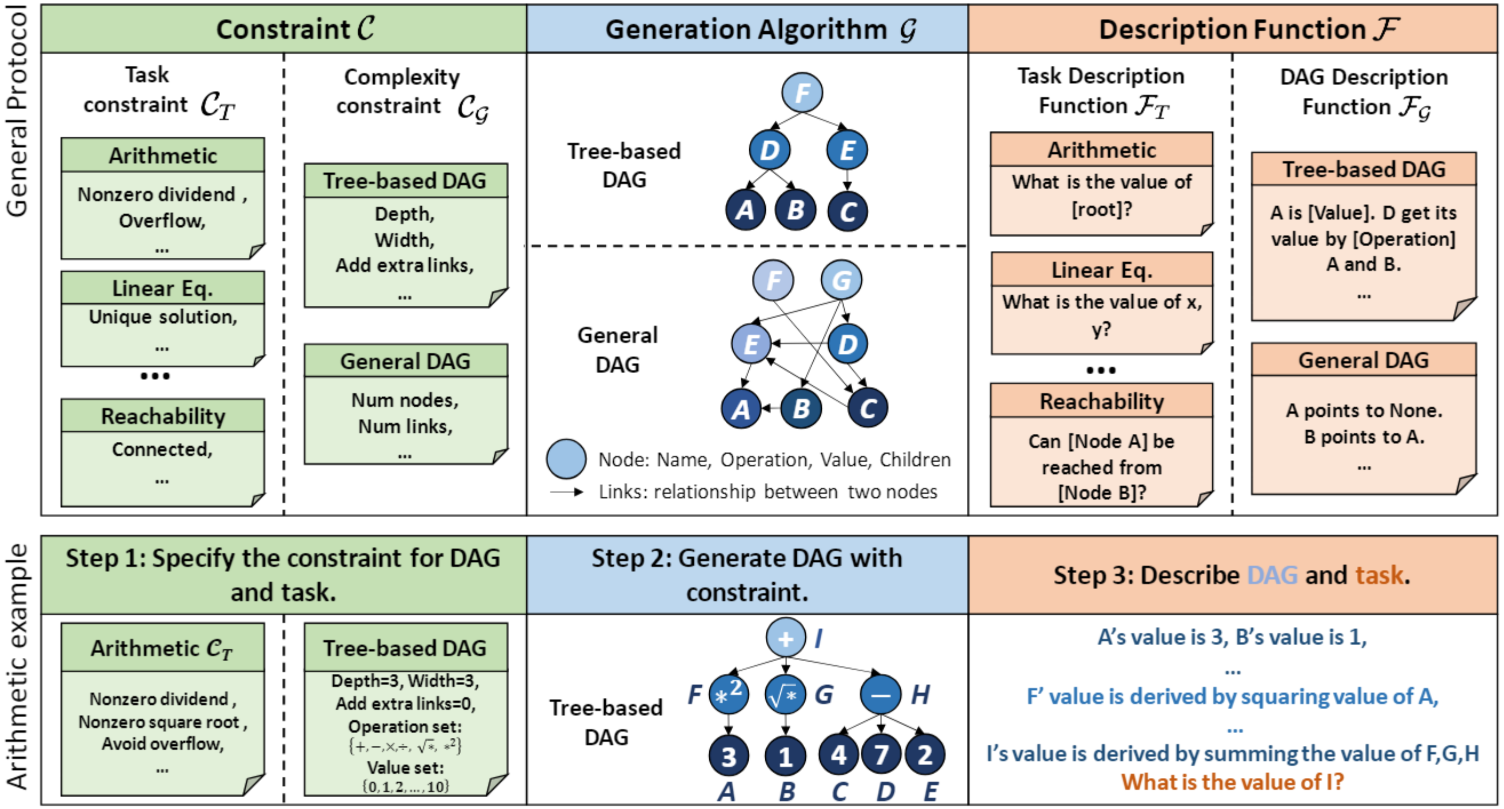
Chatbot Arena: Benchmarking LLMs in the Wild with Elo Ratir






```
Title={A Survey on Evaluation of Large Language Models},
Author={Yupeng Chang and Xu Wang and Jindong Wang and Yuan Wu},
Arxiv={2307.03109},
Cites={98}
```

LLM Evaluation



Field	Task	Generation algorithm $\mathcal{G}$	Const
			$\mathcal{C}_T$
Mathematics	Arithmetic	Tree-based	$\mathcal{V}:\{1,2,\ldots,10\}$ $\mathcal{O}:\{+,-,\times,\sqrt{\cdot},\cdot^2\}$
	Linear equation	Tree-based	$\mathcal{V}:\{1,2,\ldots,10\}$ $\mathcal{O}:\{+,-,\times,\sqrt{\cdot},\cdot^2\}$
Logical Reasoning	Bool	Tree-based	$\mathcal{V}$: {True, False} $\mathcal{O}$: {AND, OR, NOT}
	Deductive	Tree-based	$\mathcal{V}$: {True, False} $\mathcal{O}$: {AND, OR, NOT}
	Abductive	Tree-based	$\mathcal{V}$: {True, False} $\mathcal{O}$: {AND, OR, NOT}
Algorithm	Reachability	General	$\mathcal{V}$: – $\mathcal{O}$: –
	Max sum path	General	$\mathcal{V}:\{1,2,\ldots,10\}$ $\mathcal{O}$: –

```
Title={DyVal: Graph-informed Dynamic Evaluation
of Large Language Models},
Author={Zhu, Kaijie and Chen, Jiaao and Wang,
Jindong and Gong,
year={2023},
Arxiv={2309.17167}
```

THANK YOU