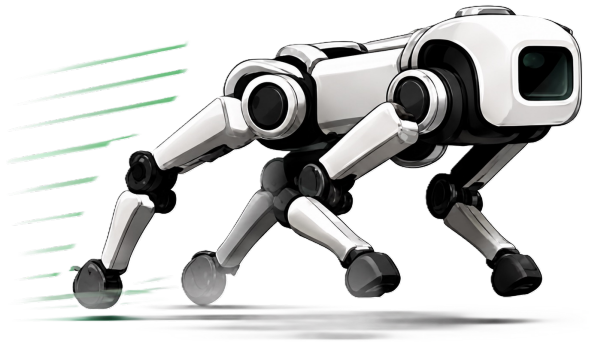
D i B O

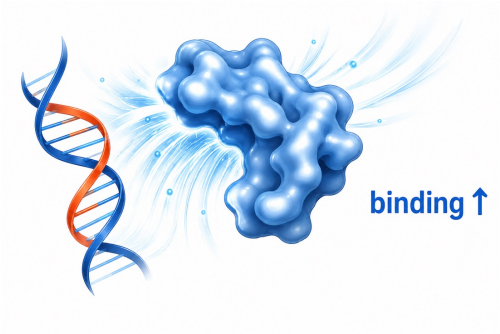
Training Diffusion Language Models for Black-Box Optimization

Zipeng Sun*, Can (Sam) Chen*, Ye Yuan, Haolun Wu, Jiayao Gu, Chris Pal, Xue Liu

* Equal contribution

ICML 2026 Spotlight (Top 2.2%)





Outline

1. What black-box optimization is
2. What a diffusion language model is
3. DiBO: putting the two together
4. Experiments: results and ablations

A small game: which coffee would you make next?

Recipe	Powder	Coffee	Water	Time	Rating
A	Fine	15 g	92 °C	2 min	7.1
B	Fine	15 g	96 °C	4 min	3.8
C	Medium	18 g	92 °C	3 min	6.9
D	Medium	15 g	96 °C	3 min	7.6
E	**Coarse**	**18 g**	**96 °C**	**5 min**	**8.0**
F	Coarse	18 g	88 °C	2 min	2.9

- Every tasting costs beans and time
- Today you are out of beans: no more tasting
- Suggest one new recipe that beats 8.0

That was black-box optimization

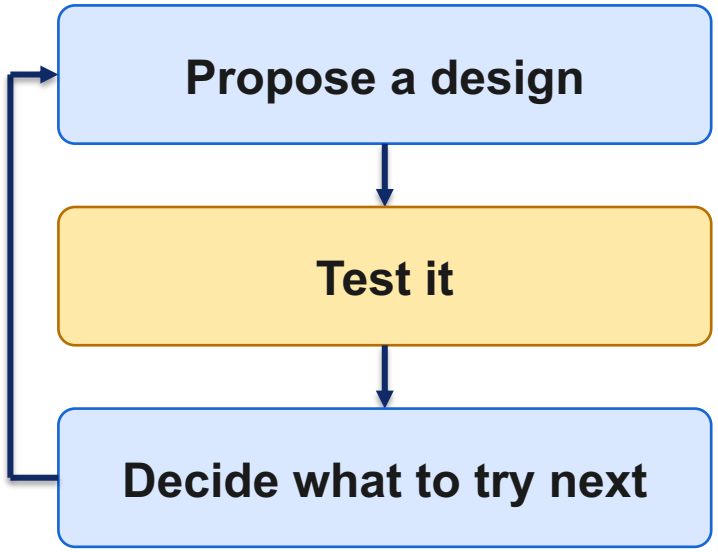


Design: what you choose. **Label:** how good it turns out.

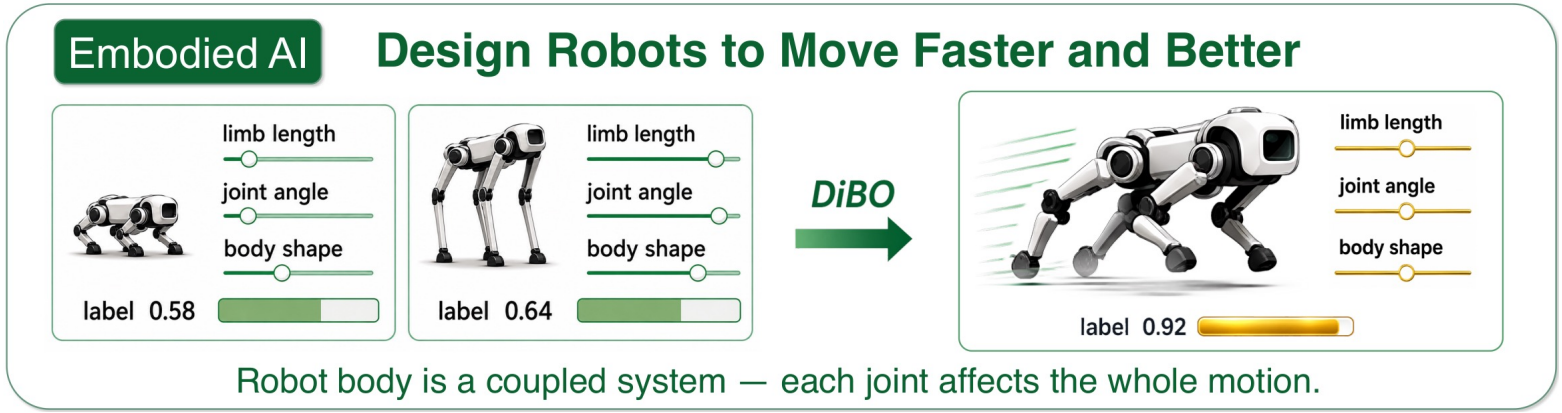
Find a design with a higher label. No formula, no gradient, only past examples.

	Design	Label
Coffee	recipe	rating
Robot	leg sizes, joint ranges	how well it moves
DNA	sequence	binding strength
Neural network	architecture (layers, widths)	accuracy

In science, every test is slow and expensive



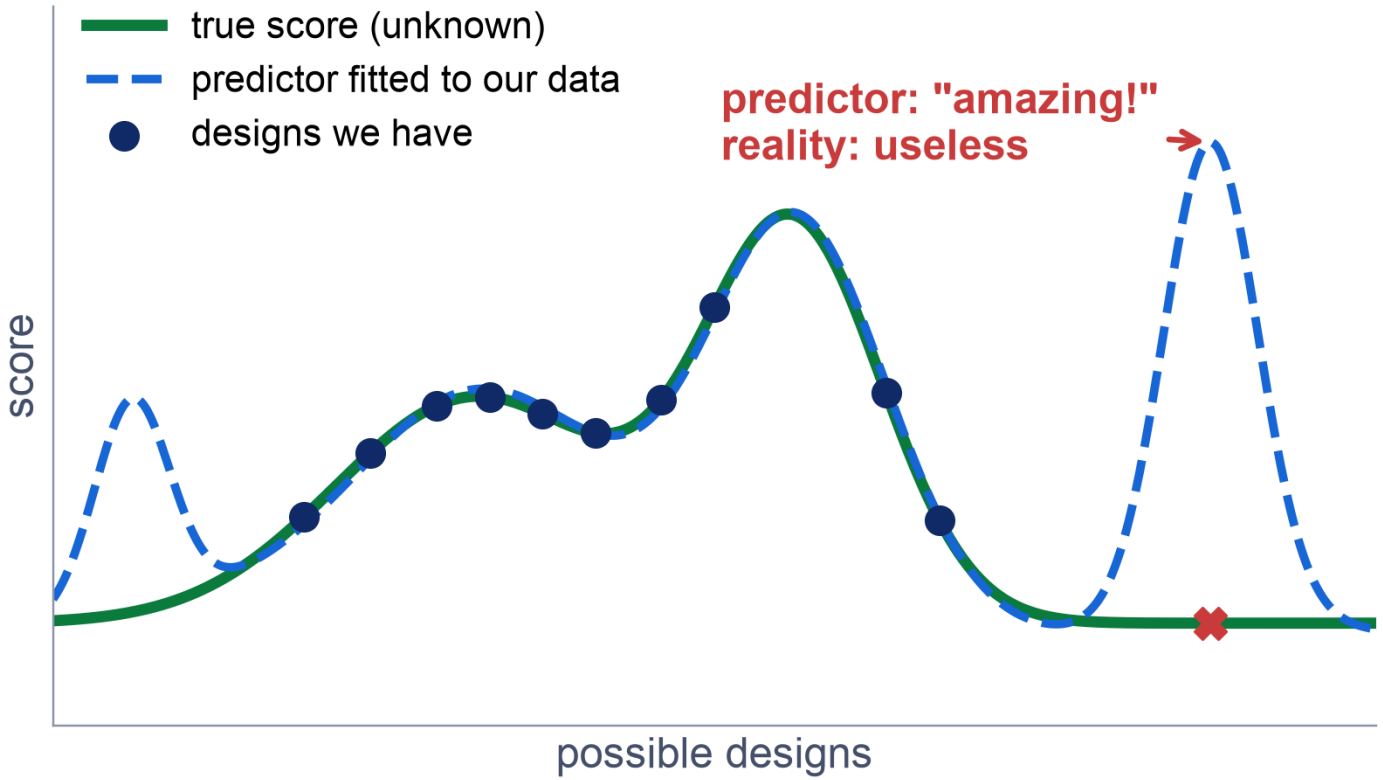
Slow part: the test. A long simulation, or a real lab experiment.





So real data is scarce. Can a model make better guesses from a few tests?

The usual approach: fit a predictor, then optimize it

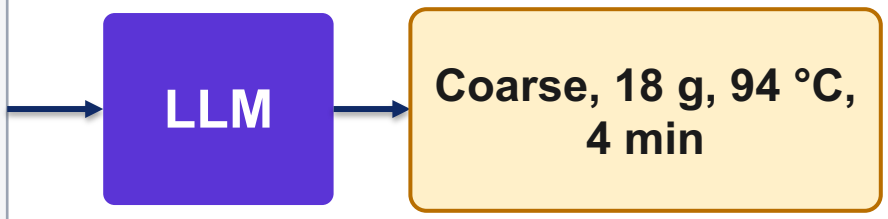


- Train a model to predict the score
- Search for the best predicted score
- The search goes where the predictor is wrong

The search ends up chasing the predictor's mistakes.

A newer approach: ask a language model

```
Suggest a coffee recipe with a higher rating.
Coarse, 18 g, 96 °C, 5 min → 8.0
Medium, 15 g, 96 °C, 3 min → 7.6
Fine,   15 g, 92 °C, 2 min → 7.1
Medium, 18 g, 92 °C, 3 min → 6.9
Propose a new recipe with a higher rating.
```



- Any task you can describe in text uses the same interface
- Most of this work uses left-to-right LLMs

Next: how the model writes its answer

A language model can propose designs. But the way it writes matters when the parts of a design depend on each other.

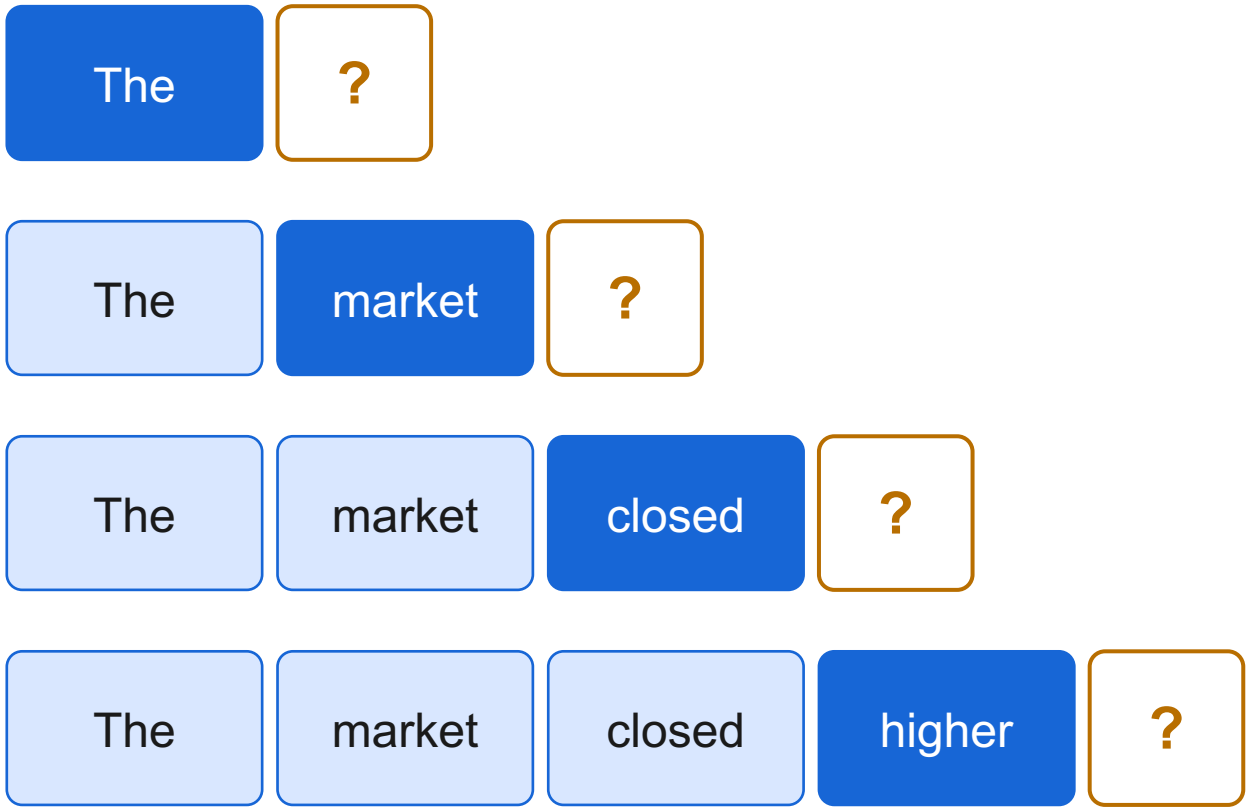
Left to right



Fill in the blanks

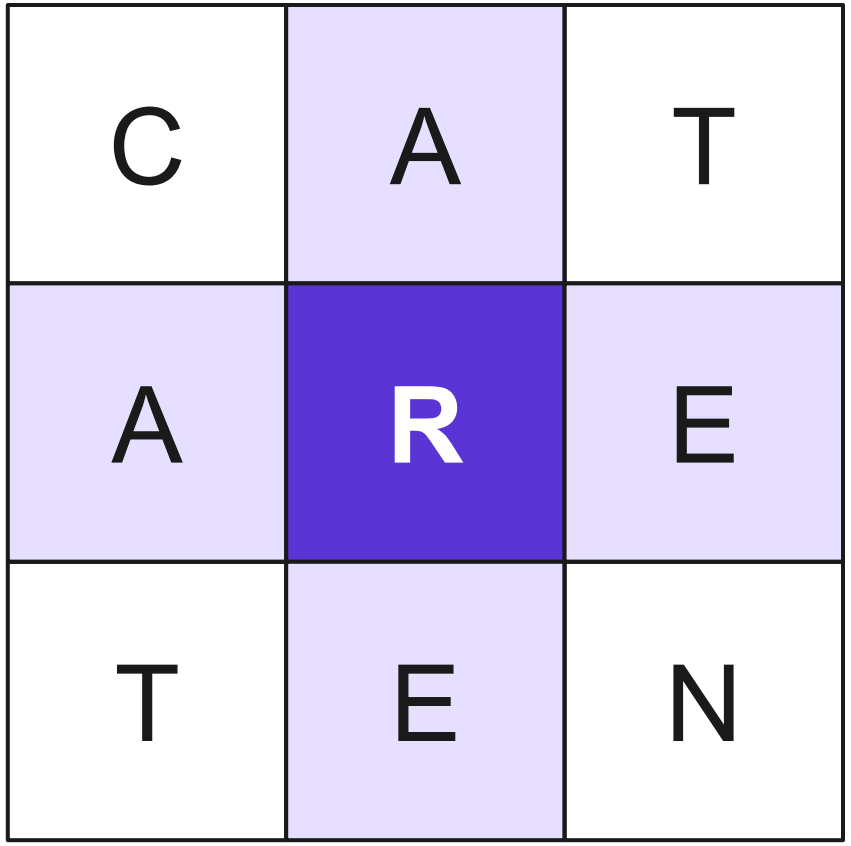


How most LLMs write



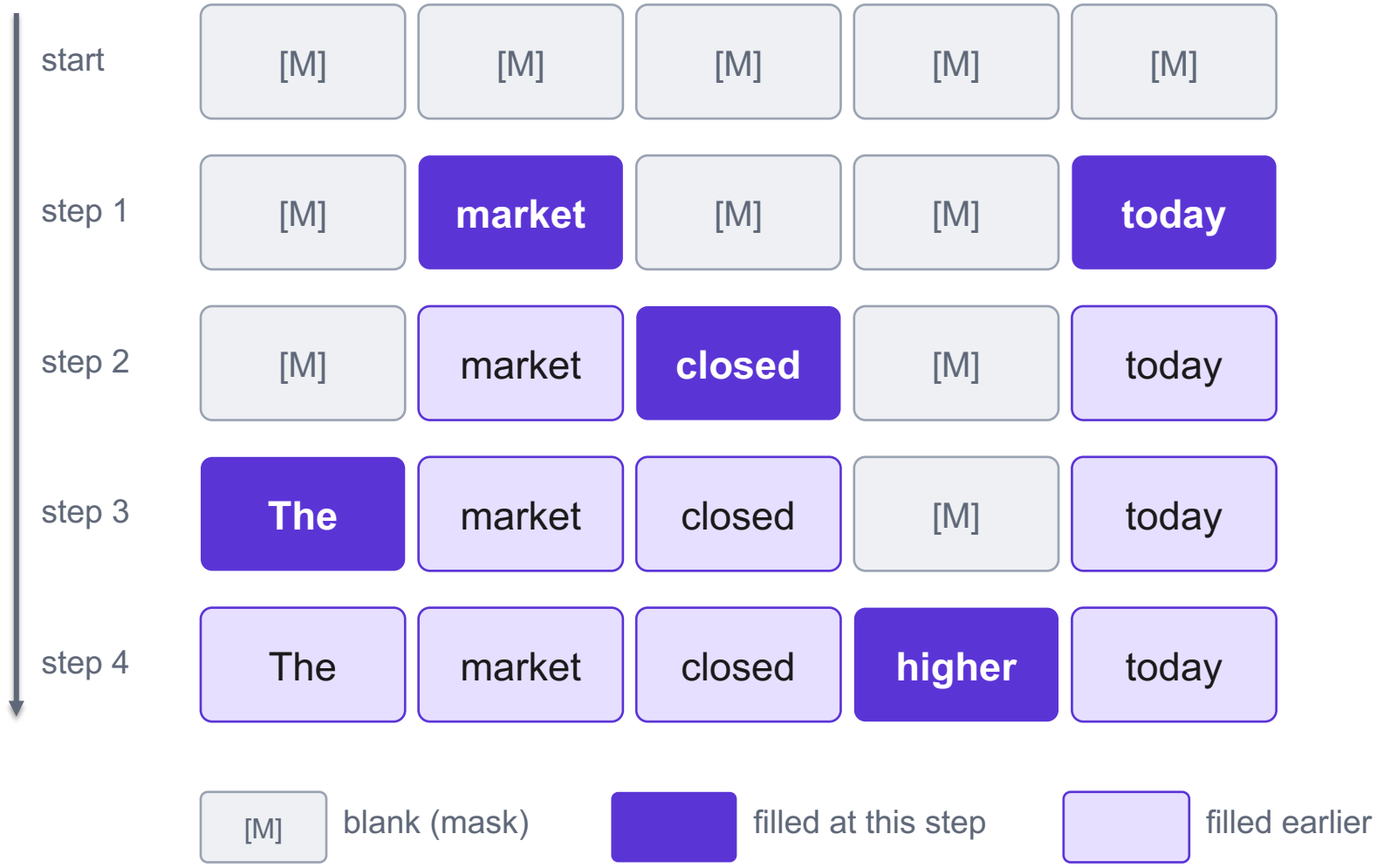
- One token at a time, left to right
- Each token sees only what came before
- Once written, a token stays

Left to right struggles when parts interlock



- Every letter has to fit its row and its column
- The R is constrained from all four sides
- Writing strictly left to right means committing to early letters before seeing later constraints

A diffusion language model starts from blanks

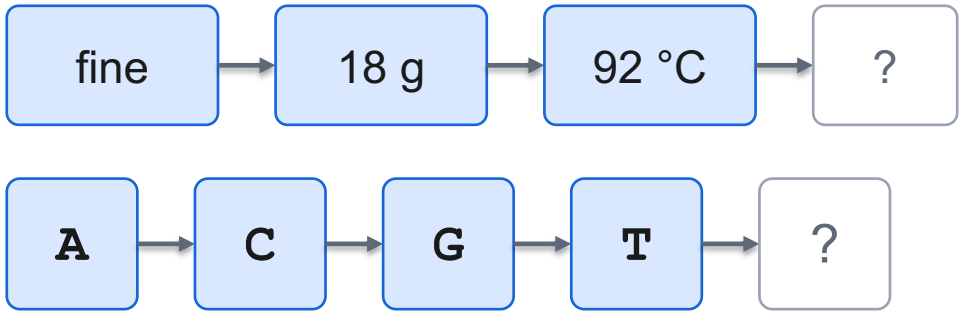


- Training: hide random tokens, predict them
- Generation: start fully masked, refine over a few steps
- Each prediction uses both sides
- We build on LLaDA-8B-Instruct

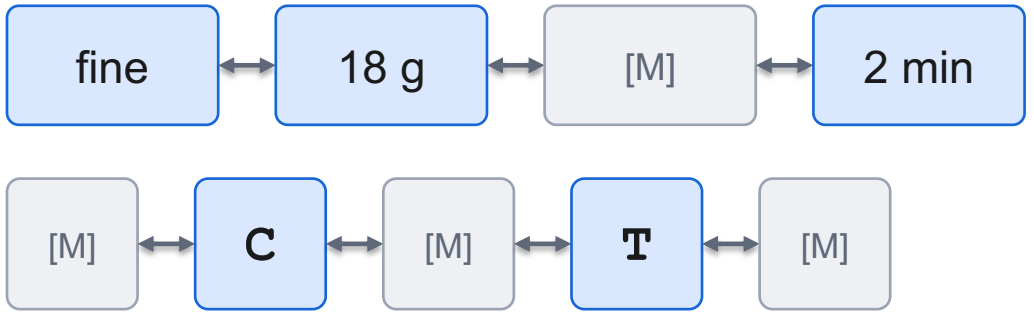
LLaDA: Nie et al., Large Language Diffusion Models, 2025.

Why this fits design problems

Autoregressive: left context only



Diffusion: both sides



- Coffee: the right brewing time depends on the powder
- DNA: the effect of a base depends on its neighbours on both sides
- Robots: change one joint and the whole motion changes

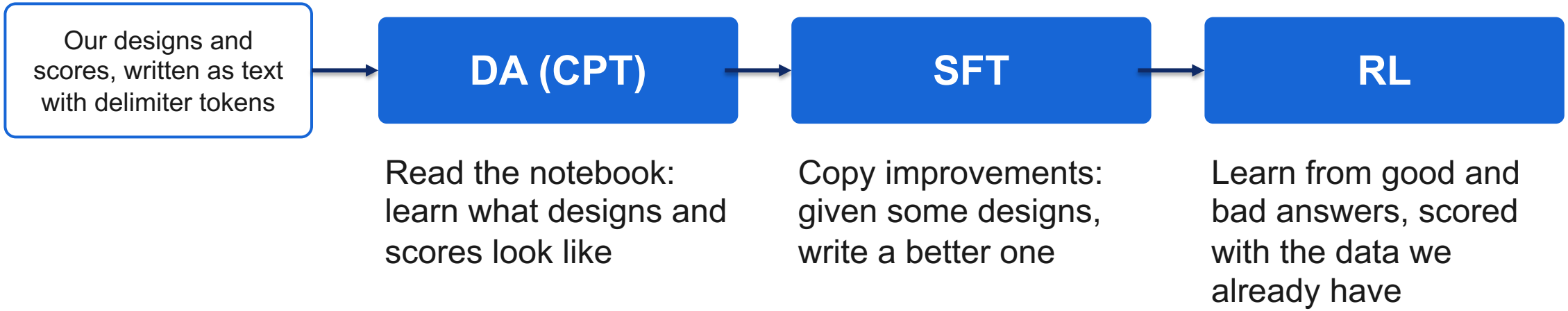
So: diffusion offers two things designs need: **it sees both sides**, and it can **refine a draft over several steps** instead of committing token by token.

Hypothesis: if the parts of a design depend on each other, model them jointly.

DiBO: teach the model your data in three stages

To a model that has read the web but not your lab notebook, these are just odd-looking text:

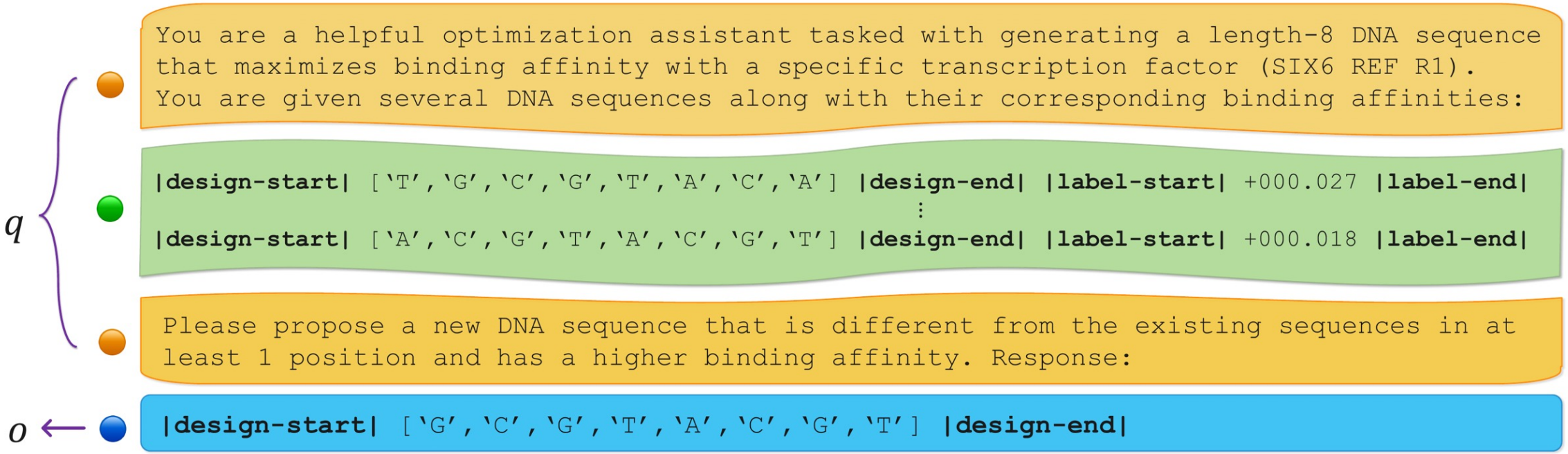
```
['T','G','C','G','T','A','C','A']  score: +000.027
```



Domain Adaptation via Continued Pretraining — DA (CPT)

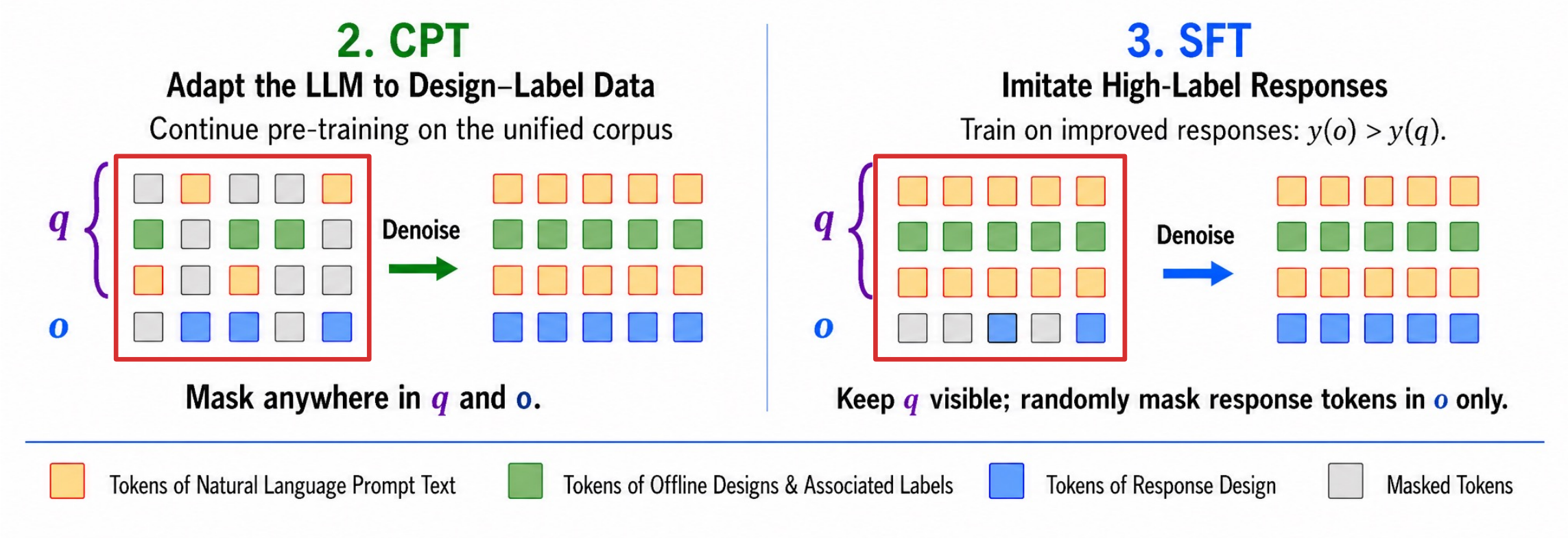
SFT: supervised fine-tuning. RL: reinforcement learning.

One text format for task, designs and labels



- Delimiter tokens mark where each design and label starts and ends
- Prompt examples are similar to the response, and the response scores higher

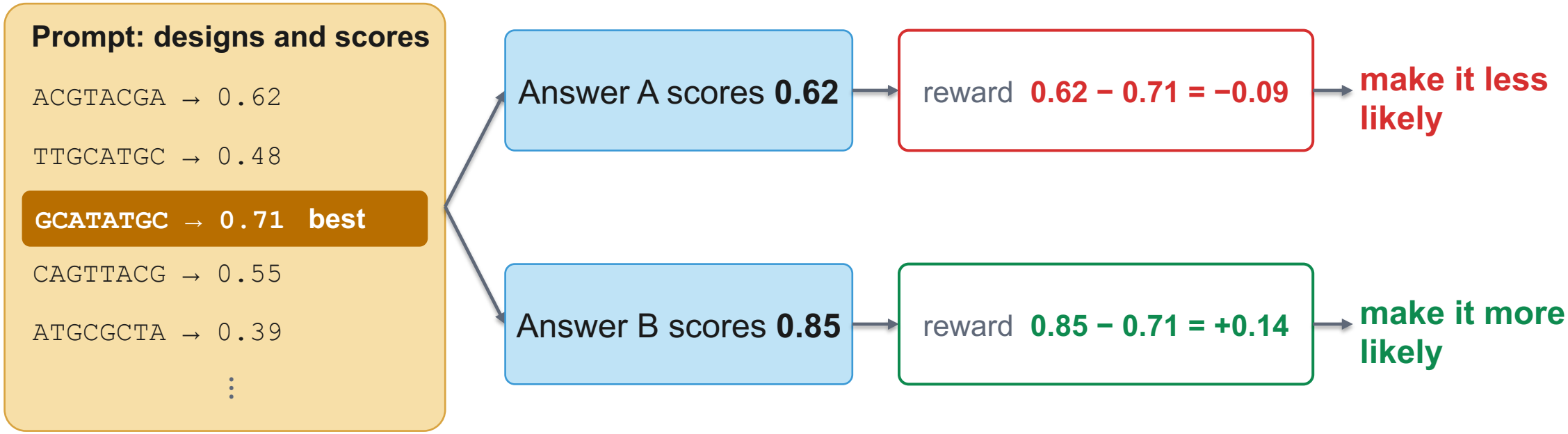
Stages 1 and 2: learn the format, imitate better designs



DA (CPT). Goal: understand the data. Hide pieces anywhere, in prompt and answer; fill them in.

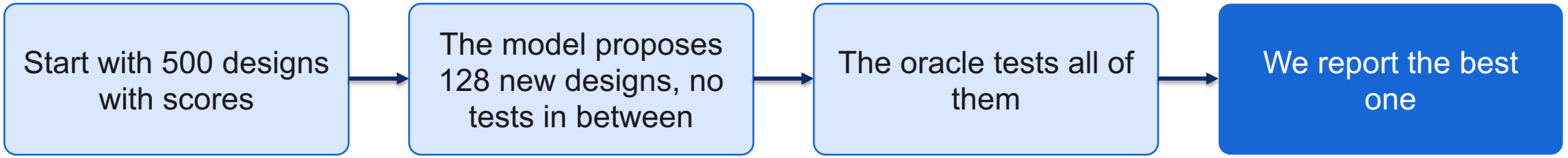
SFT. Goal: learn to improve. Show the prompt, hide only the answer. The answer always scores higher.

Stage 3: learn from good and bad answers



- Reward = answer's score minus the best score in the prompt. Bigger gap, bigger push
- SFT only saw better answers. RL also learns from worse ones
- All scores come from the labelled designs we already have: no new tests

How we tested it



Real data is scarce, so each task gives only 500 labelled designs and 128 tries.

Like getting 128 new coffee recipes, and only then tasting them all.

- Four Design-Bench tasks: DNA (TF Bind 8, TF Bind 10) and robots (Ant, D'Kitty)
- Plus three RNA design tasks in the paper's appendix
- 15 baselines, 8 random seeds
- Everything is scored by the benchmark: no real robots or wet-lab experiments

Main results

Method	Ant	D'Kitty	TF Bind 8	TF Bind 10	Mean	Rank mean	Rank median
Best in dataset	0.565	0.884	0.439	0.511			
Grad-mean	0.709 ± 0.002	0.920 ± 0.008	0.843 ± 0.082	0.736 ± 0.016	0.802 ± 0.027	4.25	3.5
Grad-EI	0.655 ± 0.002	0.923 ± 0.010	0.864 ± 0.091	0.727 ± 0.024	0.792 ± 0.032	4.25	4.0
UniSO-T	0.636 ± 0.045	0.939 ± 0.007	0.836 ± 0.027	0.522 ± 0.017	0.733 ± 0.024	6.75	6.5
ExPT	0.929 ± 0.049	**0.950 ± 0.041**	0.810 ± 0.044	0.703 ± 0.022	0.848 ± 0.039	4.0	4.0
DiBO (ours)	**0.932 ± 0.022**	0.912 ± 0.017	**0.946 ± 0.043**	**0.741 ± 0.027**	**0.883 ± 0.027**	**3.5**	**1.0**

Best overall of 16 methods, and first on three of four tasks. ExPT is better on D'Kitty.

Does diffusion actually help?

Same data, prompts, delimiters and training stages. Only the backbone changes.

After the full pipeline	TF Bind 8	TF Bind 10	Ant	D'Kitty
Left to right (LLaMA-3.1-8B)	0.915 ± 0.008	0.682 ± 0.053	0.930 ± 0.000	0.912 ± 0.011
Diffusion (LLaDA-8B)	0.946 ± 0.043	0.741 ± 0.027	0.932 ± 0.022	0.912 ± 0.017

Gain from diffusion after each stage

Stage	TF Bind 8	TF Bind 10	Ant	D'Kitty
DA (CPT)	**+0.080**	**+0.021**	+0.018	+0.015
SFT	**+0.064**	**+0.071**	+0.035	+0.003
RL	**+0.031**	**+0.059**	+0.002	0.000

- DNA: clear gains at every stage
- Robots: gains during training, then both end near the same score
- Possible reason: long lists of numbers are harder to capture at 8B than short DNA; a larger model may help

Every training stage matters

DA (CPT)	SFT	RL	TF Bind 8	Change	Ant	Change
✓	✓	✓	0.946 ± 0.043		0.932 ± 0.022	
✓	✓		0.939 ± 0.031	−0.007	0.929 ± 0.017	−0.003
	✓	✓	0.941 ± 0.030	−0.005	0.913 ± 0.024	−0.019
✓		✓	0.928 ± 0.052	−0.018	0.858 ± 0.032	−0.074
✓			0.883 ± 0.032	−0.063	0.875 ± 0.026	−0.057
	✓		0.938 ± 0.034	−0.008	0.898 ± 0.027	−0.034
		✓	0.437 ± 0.001	−0.509	no valid output	

- Remove any stage and the score drops
- RL works best after SFT
- RL alone fails: the model hasn't learned the format, so it writes almost no valid designs, and RL has nothing good to learn from

Delimiter tokens mark each design and score

Without

```
Designs: ['T','G','C','G','T','A','C','A']  Labels: +000.027
```

With (ours)

```
|design-start| ['T','G','C','G','T','A','C','A'] |design-end| |label-start| +000.027
|label-end|
```

Without markers, the model has to guess where a design ends and its score begins.

With them, every field is explicit, so it learns the format faster.

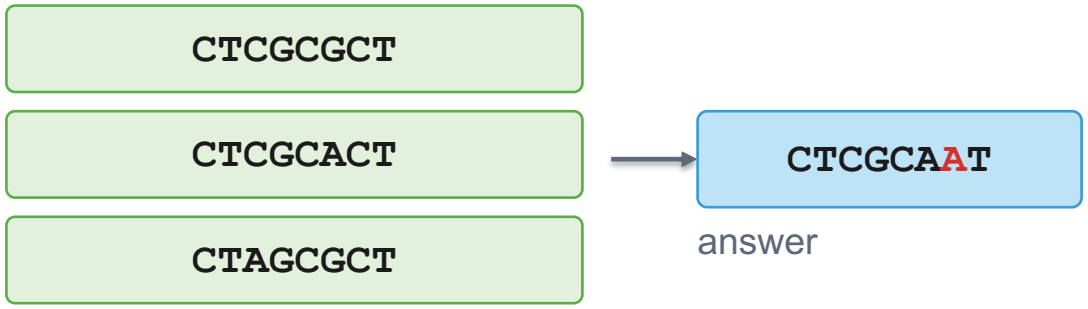
Ant, score after each stage

	DA (CPT)	SFT	RL
Without	0.693 ± 0.016	0.903 ± 0.022	0.923 ± 0.019
With (ours)	0.875 ± 0.026	0.929 ± 0.017	0.932 ± 0.022

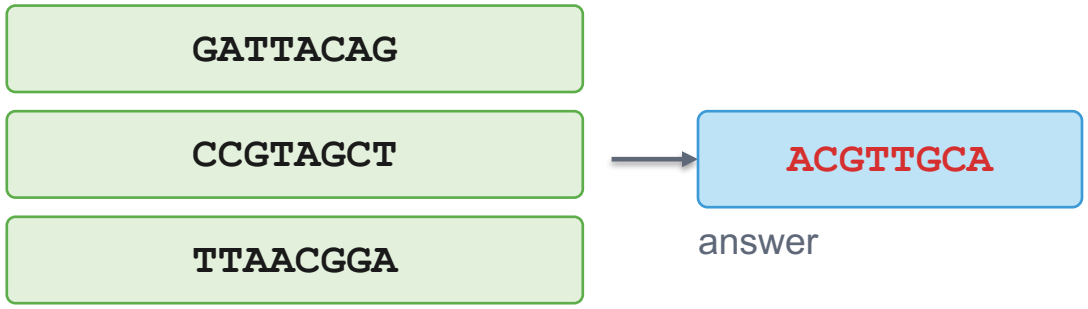
So: delimiter tokens help the model learn the format, most of all early: right after DA (CPT), 0.693 → 0.875 on Ant.

Similar examples in the prompt help

Similar examples: a small step



Random examples: a big jump



Examples in the prompt	TF Bind 8	Ant
Random	0.730 ± 0.001	0.906 ± 0.008
Similar (ours)	0.946 ± 0.043	0.932 ± 0.022

Nearby examples show the model one small change at a time, which it can learn from.

So: similar examples in the prompt matter: with random ones, TF Bind 8 drops from 0.946 to 0.730.

Summary

- Black-box optimization: better designs from limited past data
- Diffusion LLMs see both sides, which suits designs whose parts depend on each other
- DiBO trains one with DA (CPT) → SFT → RL and ranks best overall on our four tasks
- Every training stage, delimiter tokens and similar examples help; diffusion helps most on DNA

D i B O

Paper

Code

Thank you!

zipeng.sun@mila.quebec