Evolutionary Optimization of Model Merging Recipes

Takuya Akiba, Makoto Shing, Yujin Tang, Qi Sun, David Ha

sakana.ai

目次

1. モデルマージ

2. **進化的**モデルマージ

1

モデルマージ

画像はOmar Sanseviero さん作

モデルマージとは？

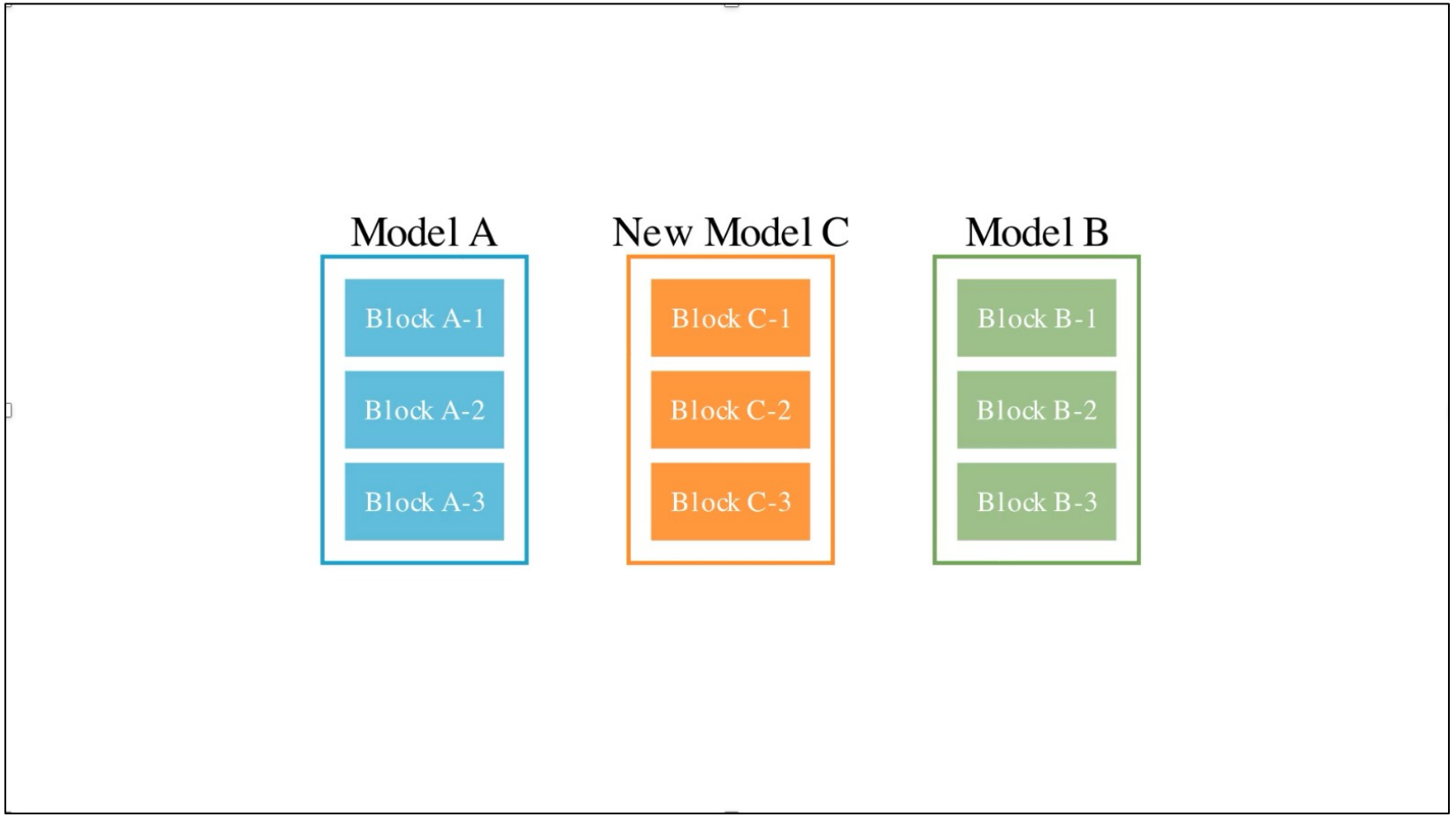
$$\mathrm{LLM}_{\mathrm{new}} = \mathrm{Merge}(\mathrm{LLM}_1, \mathrm{LLM}_2, \mathrm{LLM}_3, \dots)$$

2つ以上のモデルを元に1つの新たなモデルを作る

アプローチ2つ

1. 重みレベルのモデルマージ
2. レイヤーレベルのモデルマージ

アプローチ1
重みレベルのモデルマージ



↑動画はこちら：https://sakana.ai/evolutionary-model-merge-jp/

アプローチ1
重みレベルのモデルマージ

手法1: 線形補間

$$\theta_{\text{new}} = \alpha\theta_1 + (1-\alpha)\theta_2$$

($\theta_1, \theta_2, \theta_{\text{new}}$：LLMの重み、 α ：混ぜ合わせの設定パラメタ)

- 既存の重みの線形補完で新たなモデルを作る
- 同じアーキテクチャのモデル同士でないとできない
- 同じbase modelからのfine-tuneでないと基本成功しない

アプローチ1
重みレベルのモデルマージ

- 同様に、重みをelement-wiseでマージする手法がいくつかある
- 大体、線形補間のちょい発展版みたいなもんだと思っておけば一旦OK

[2203.05482] Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time

[2212.04089] Editing Models with Task Arithmetic

[2306.01708] TIES-Merging: Resolving Interference When Merging Models

[2311.03099] Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch

[2403.19522] Model Stock: All we need is just a few fine-tuned models

アプローチ1
重みレベルのモデルマージ

実例：

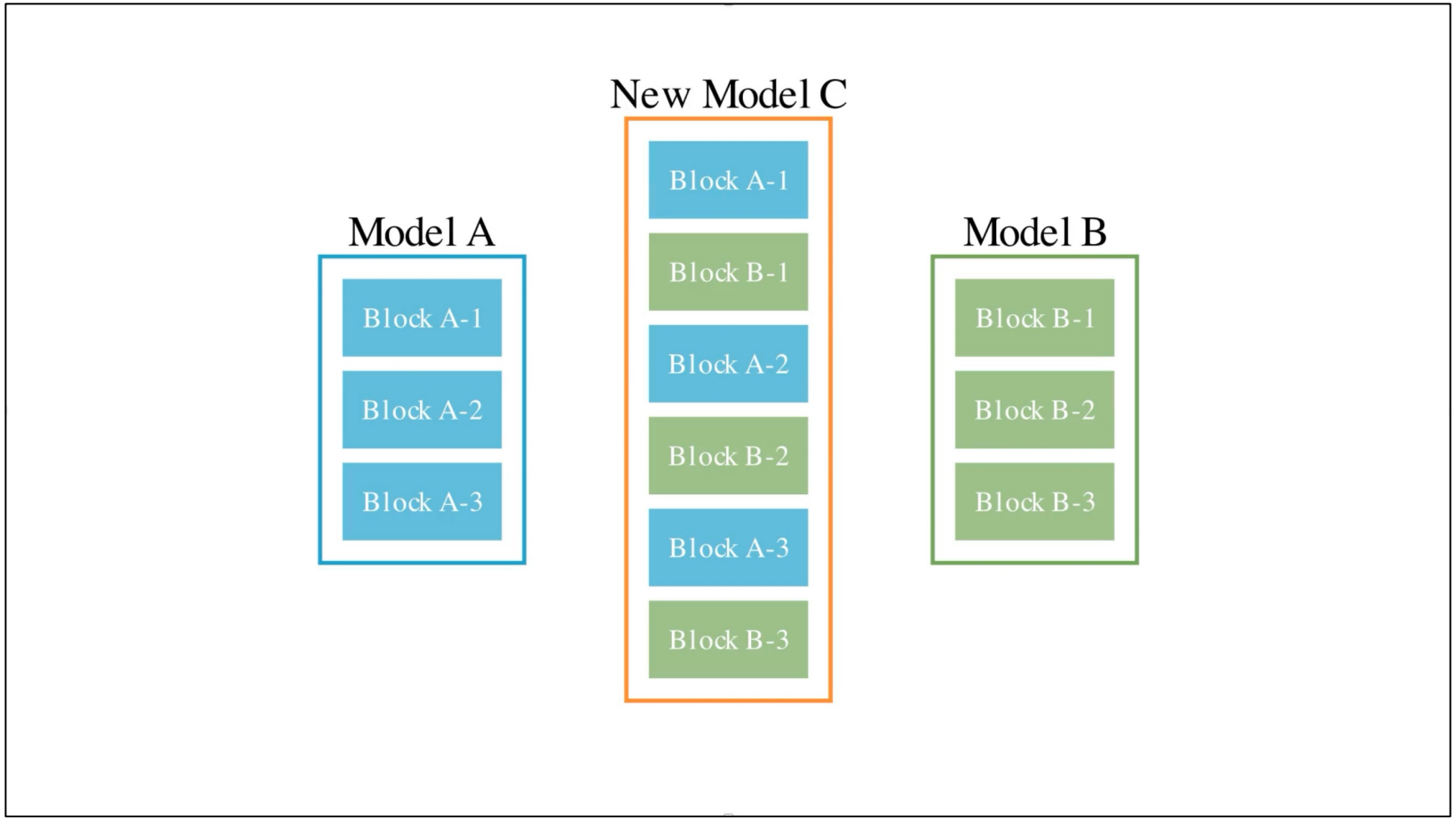
- Open LLM Leaderboard を見れば無数にある

- 例えば Mistral-7B-Merge-14-v0.1は
  Mistral 7Bのfine-tuneを**14個**マージ
  （そしてこいつが更にマージされ別のモデルが作られているというカオス）

- Stable DiffusionのLoRAのマージとかも同じ話

アプローチ2

レイヤーレベルのモデルマージ



↑動画はこちら：https://sakana.ai/evolutionary-model-merge-jp/

アプローチ2

レイヤーレベルのモデルマージ

- 重み自体は弄らず、レイヤーを再配置する
  （レイヤー = transformer block）

- パラメータ数が増減する

アプローチ2

レイヤーレベルのモデルマージ

論文

- [2312.15166] SOLAR 10.7B: Scaling Large Language Models with Simple yet Effective Depth Up-Scaling
- [2401.02415] LLaMA Pro: Progressive LLaMA with Block Expansion

モデル

- alpindale/goliath-120b
- cognitivecomputations/MegaDolphin-120b
- Undi95/Mistral-11B-v0.1

人は何を求めてモデルをマージする？

①アンサンブル（的な何か）

モデルをマージするとOpen LLM Leaderboardのスコアが上がる。

しかも学習不要で新しいモデルが作れる。

皆こぞって謎のマージを作りLeaderboardに提出。

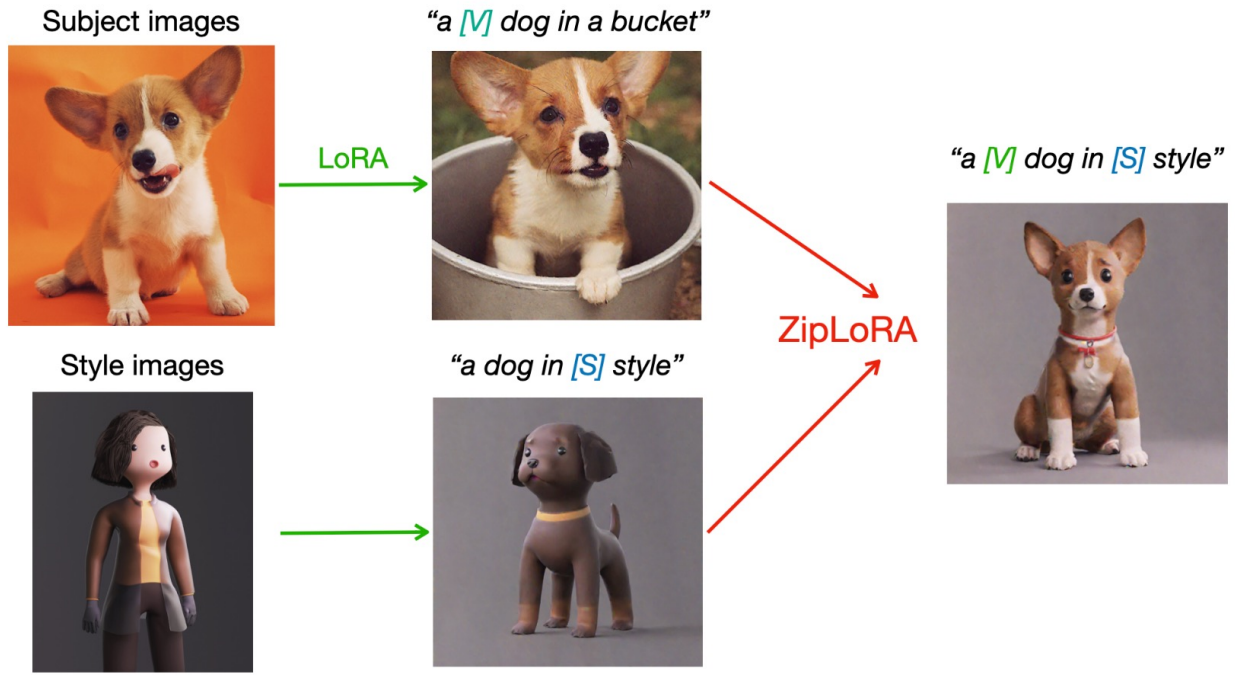
（※ただし後述の通りこれはめっちゃLeaderboardにoverfitしてると思う）

[1803.05407] Averaging Weights Leads to Wider Optima and Better Generalization

[2203.05482] Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time

人は何を求めてモデルをマージする？

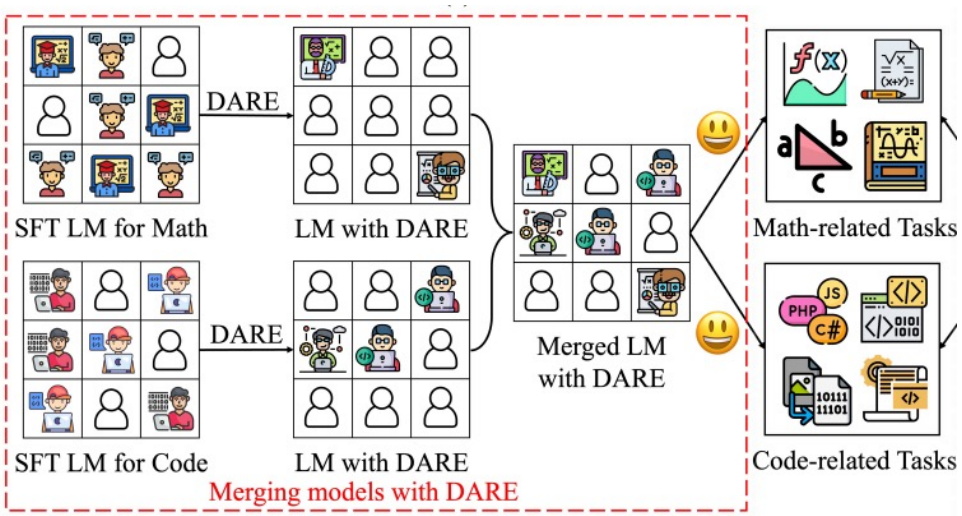
②複数の能力を統合



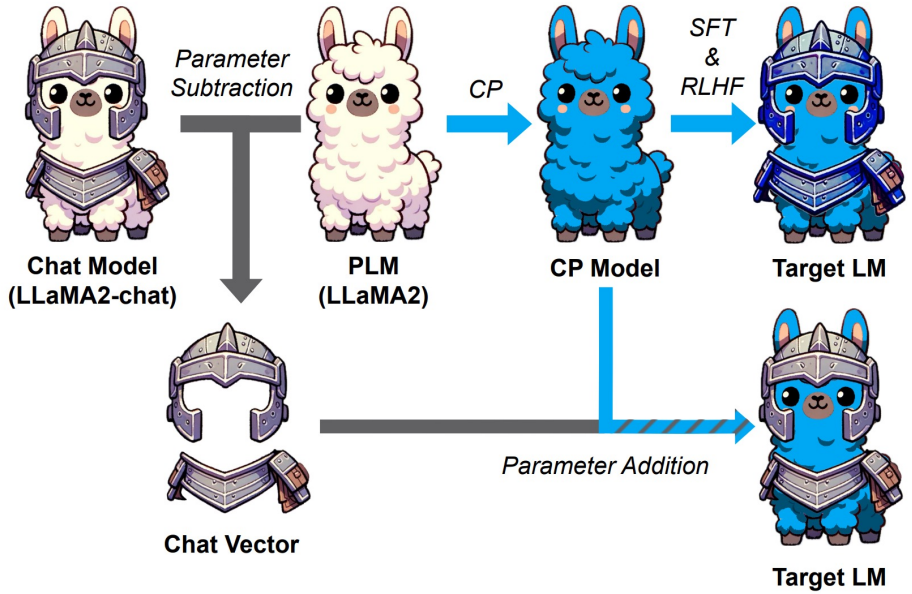
https://github.com/mkshing/ziplora-pytorch

人は何を求めてモデルをマージする？

②複数の能力を統合



[2311.03099] Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch



[2310.04799] Chat Vector: A Simple Approach to Equip LLMs with Instruction Following and Model Alignment in New Languages

モデルマージは何故うまく行く？

重みレベルの議論の例

[2305.12827] Task Arithmetic in the Tangent Space: Improved Editing of Pre-Trained Models

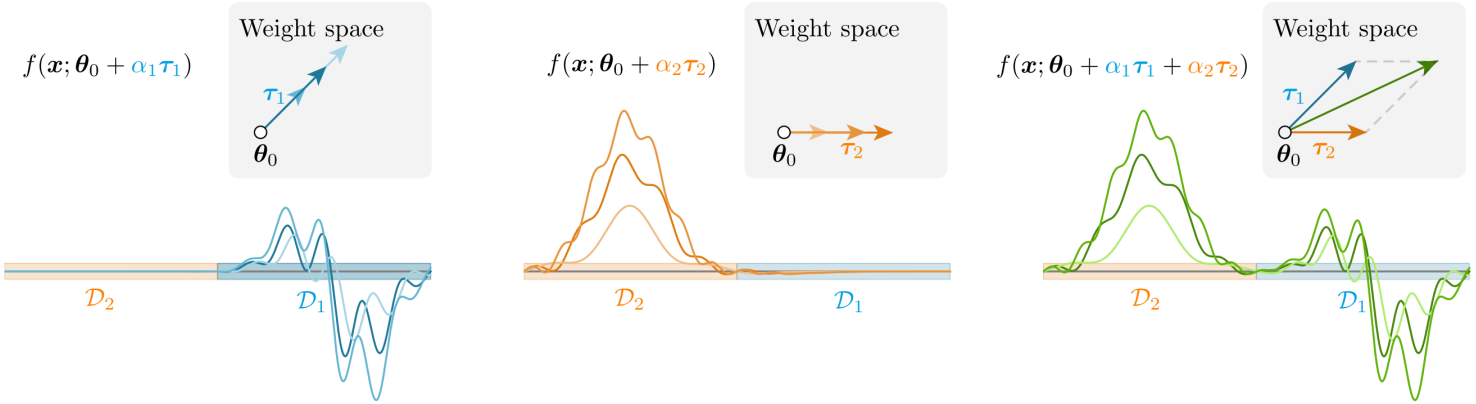


Figure 1: **Illustration of weight disentanglement**, where distinct directions in the weight space, $\boldsymbol{\tau}_t$, are associated with localized areas of the input space, $\mathcal{D}_t$. This allows a model, f, to manipulate these areas independently by adding linear combinations of $\boldsymbol{\tau}_t$'s to a pre-trained checkpoint $\boldsymbol{\theta}_0$.

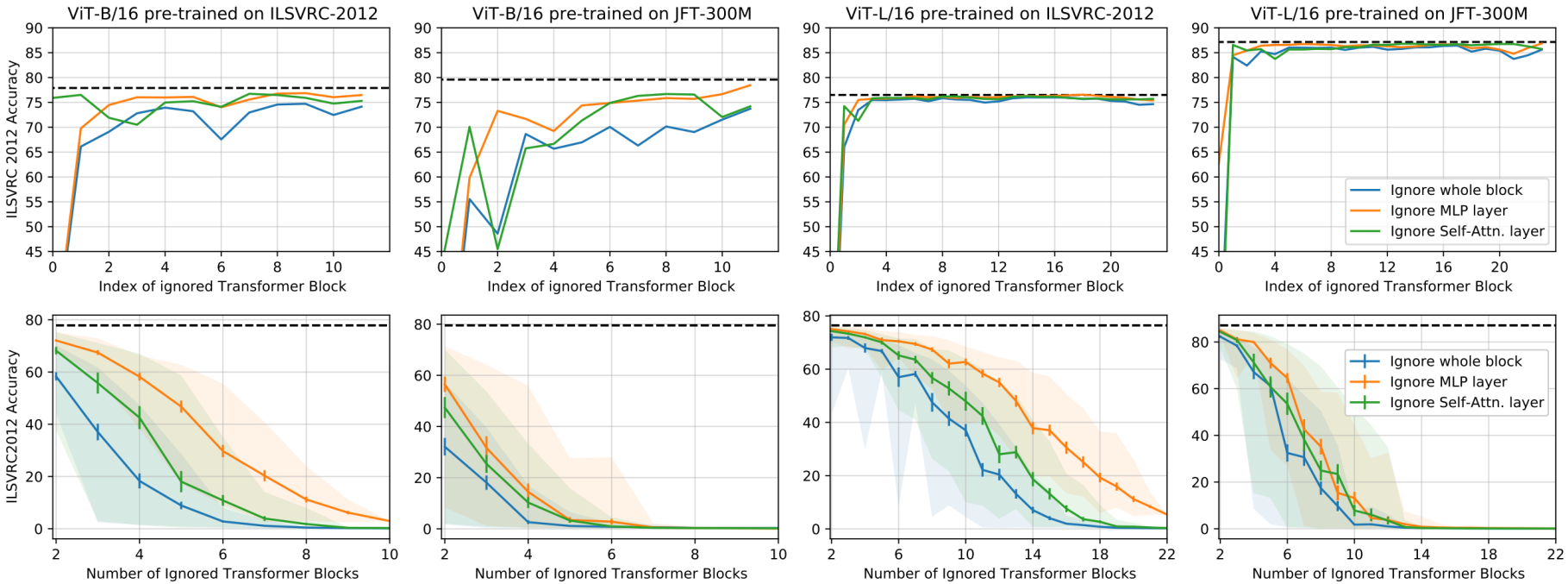
NII 佐藤先生の解説ブログが神

https://joisino.hatenablog.com/entry/2024/01/09/174517

モデルマージは何故うまく行く？

レイヤーレベルの議論の例

[2103.14586] Understanding Robustness of Transformers for Image Classification



日本での盛り上がりは今一つ？

英語圏では無数にマージモデルが日々生み出されている一方、
日本語圏ではそこまででもない。何故？

理由

- マージ元のモデルが少ない（1つの日本語base modelからの派生に限ると）
- マージが難しい（より広い範囲のモデルを使おうとすると）

日本での盛り上がりは今一つ？

何故マージが難しい？

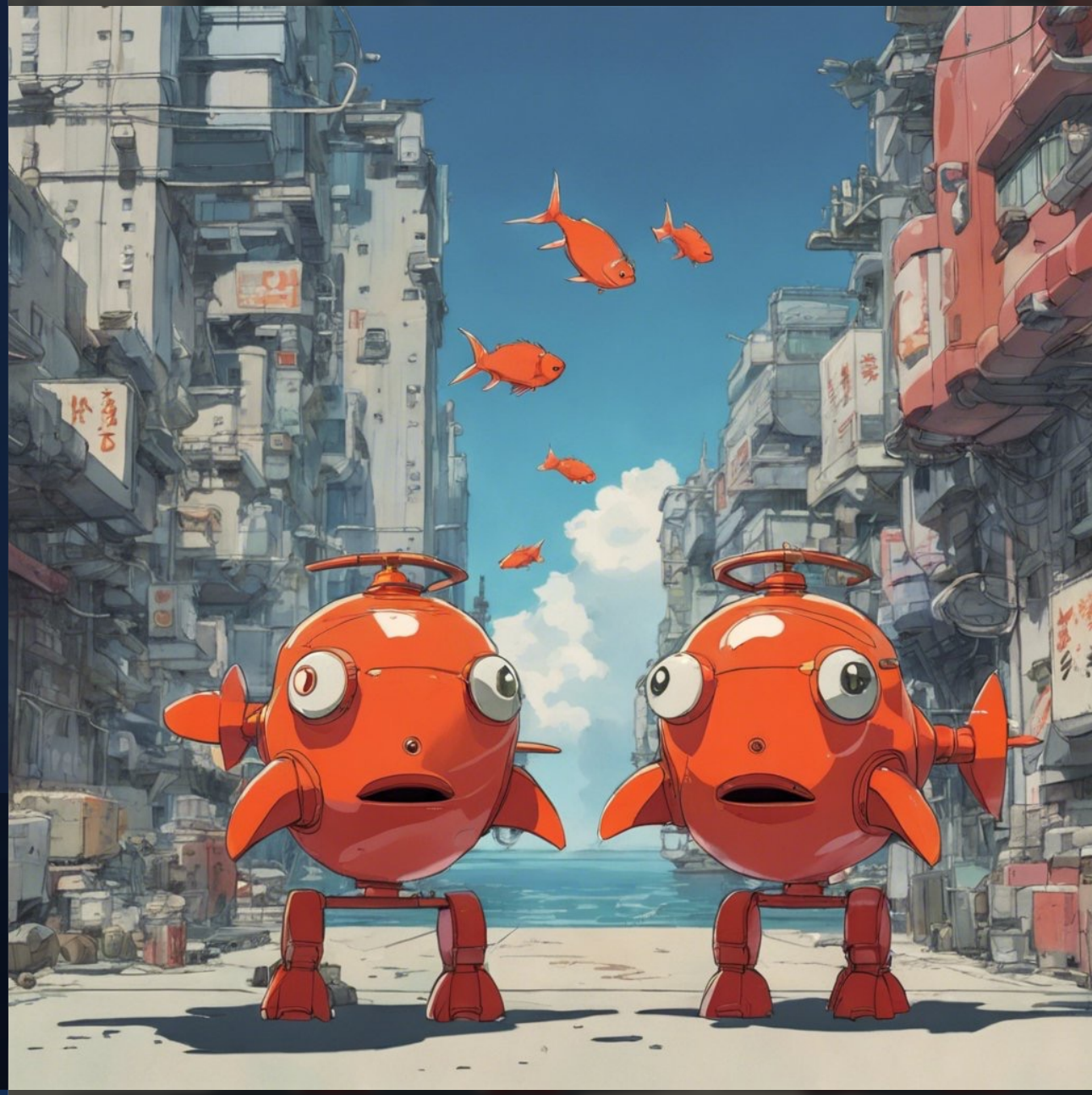
- 多くの日本語LLMは英語LLMからの**継続事前学習**で作られている
- 継続事前学習では数十B～数百Bトークンの学習が行われ、
  元の英語LLMの重みからだいぶ離れてしまっている

DAREの論文でもCodeLlama (Llama-2から500B token学習) の
マージが難しい話が同様に議論されている

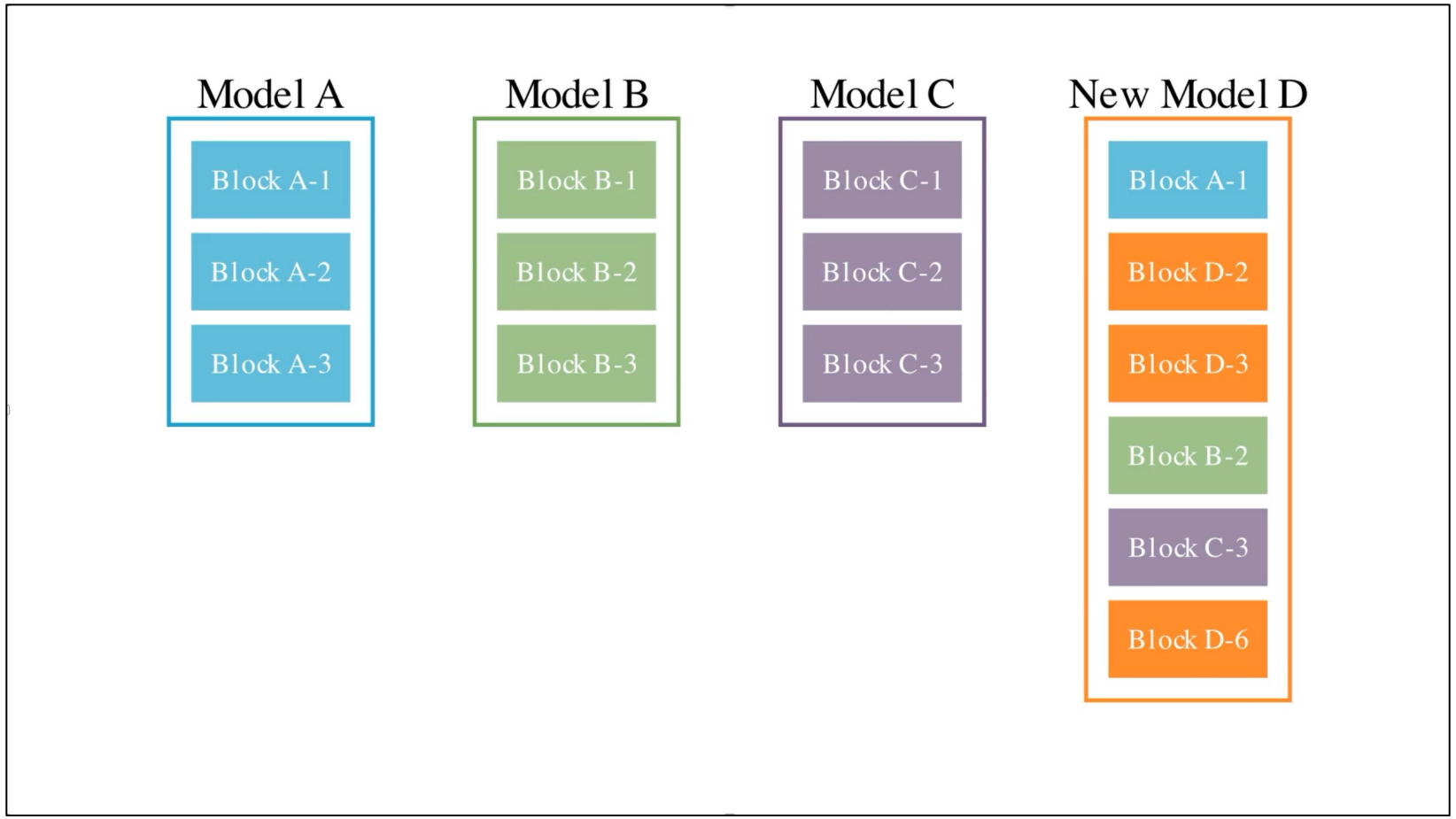
実際、私も最初は手動でマージのレシピを探ろうとしましたが、
なかなか上手くいかない…… → **自動探索！**

2

進化的
モデルマージ

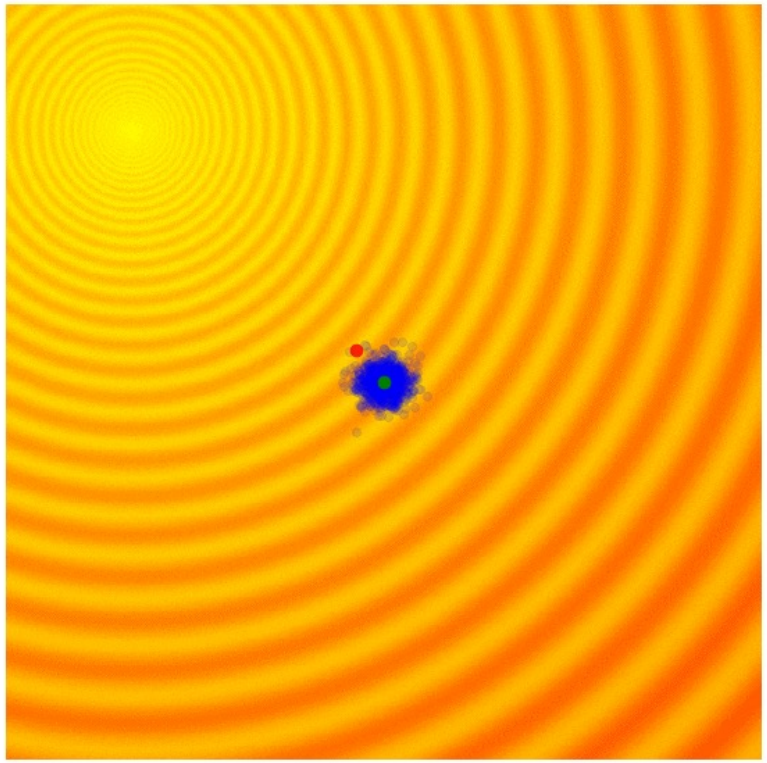


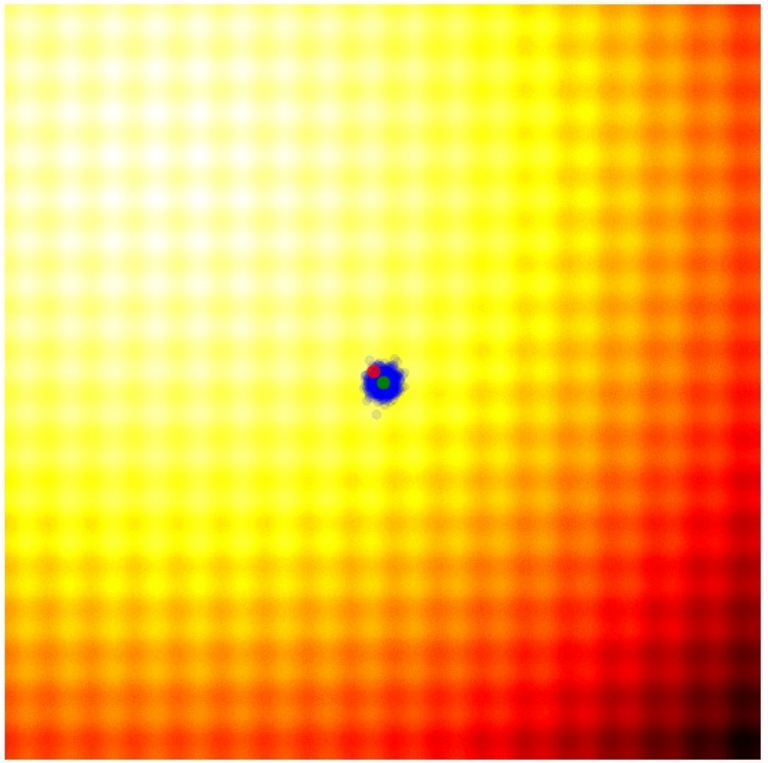
進化的モデルマージ



↑動画はこちら：https://sakana.ai/evolutionary-model-merge-jp/

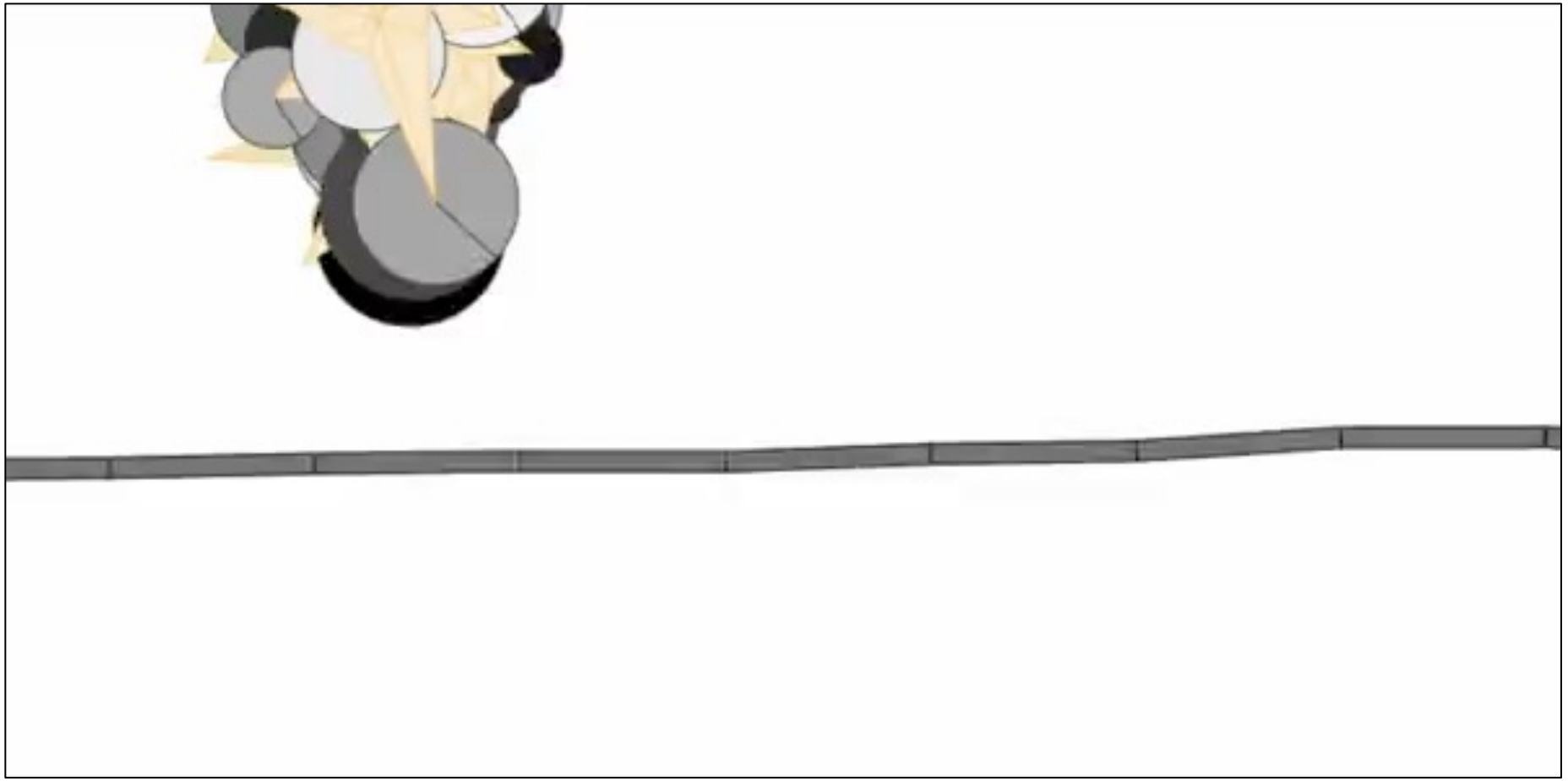
進化的アルゴリズム





A Visual Guide to Evolution Strategies
https://blog.otoro.net/2017/10/29/visual-evolution-strategies/

進化的アルゴリズム



HTML5 Genetic Algorithm 2D Car Thingy https://rednuht.org/genetic_cars_2/

結果1: 日本語数学LLM

日本語LLM + 英語数学LLM → 日本語数学LLM

Id.	Model	Type	Size	MGSM-JA (acc ↑)
1	Shisa Gamma 7B v1	JA general	7B	9.6
2	WizardMath 7B v1.1	EN math	7B	18.4
3	Abel 7B 002	EN math	7B	30.0
4	**Ours (PS)**	1 + 2 + 3	7B	**52.0**
5	**Ours (DFS)**	3 + 1	10B	**36.4**
6	**Ours (PS+DFS)**	4 + 1	10B	**55.2**
7	Llama 2 70B	EN general	70B	18.0
8	Japanese StableLM 70B	JA general	70B	17.2
9	Swallow 70B	JA general	70B	13.6
10	GPT-3.5	commercial	-	50.4

元の「日本語能力」と「英語数学能力」を**維持**出来ただけでなく、
その両方を**日本語数学能力**として**組合せて発揮**出来る

結果1: 日本語数学LLM

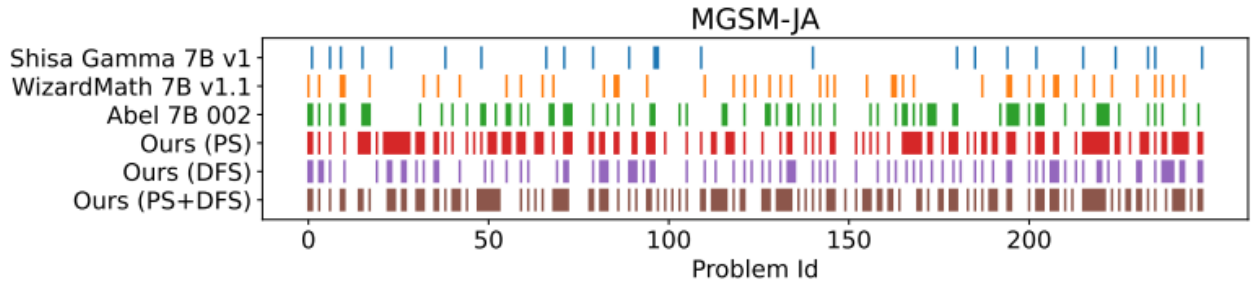


Figure 2: **Performance Overview.** The figure depicts the success of various models on the MGSM-JA task, with each of the 250 test problems represented along the x-axis by problem ID. Correct answers are indicated by colored markers at the corresponding positions.

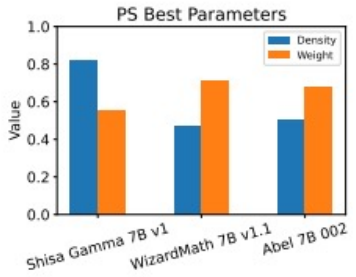


Figure 3: **Evolved Configurations for PS merging.** Although the weights are similar across the three source models, the pronounced density from the Japanese LLM underscores its pivotal role in our merged model.

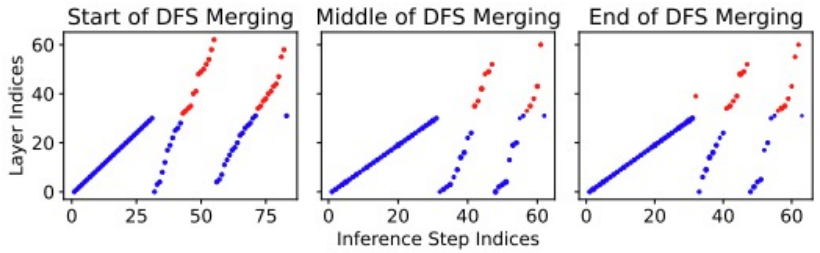


Figure 4: **Evolved Configurations for DFS Merging of models A and B.** The three figures depict the evolution of the inference path on the MGSM-JA task. The y-axis represents the layer index $l \in [1, M]$, and the x-axis corresponds to the path index $t \in [1, T]$. Blue markers indicate path steps utilizing layers from model A, and red markers denotes those from B. Marker size reflects the magnitude of the scaling factor W_{ij}. The evolutionary search result includes most layers in A at an early stage and then alternates between layers from both models. This result is from our 10B model (PS+DFS).

結果2: 日本語VLM (Vision Language Model)

日本語LLM + 英語VLM → 日本語VLM

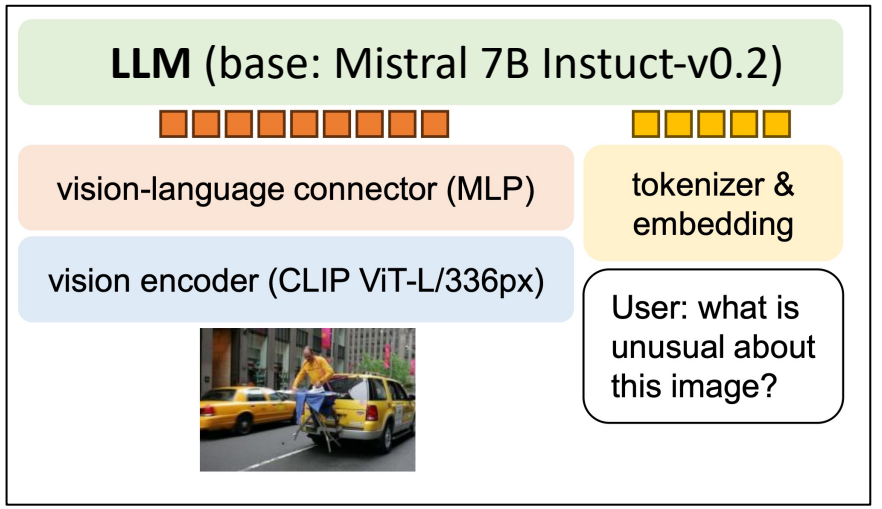
前提知識

LLaVaというVLMは、vision encoder (ViT)とLLMをくっつけた上で
fine-tuneして作られている

我々のマージ

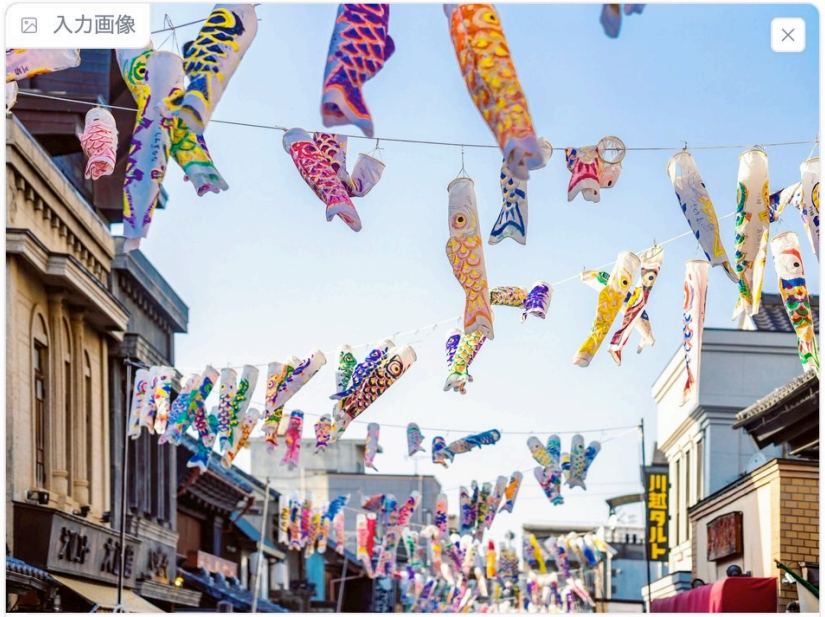
- LLaVaのLLM部分に日本語LLMをブレンド
- 他の部分はLLaVaのまま

Llava-v1.6-mistral-7b



結果2: 日本語VLM (Vision Language Model)

入力画像



質問

この画像に映っている魚の形をした飾りは、日本では何と呼ばれていますか？

モデルからの返答

この画像に映っている魚の形をした飾りは、日本では「鯉のぼり」と呼ばれています。

結果2: 日本語VLM (Vision Language Model)

入力画像

質問

この信号機の点滅している色は何色ですか?

モデルからの返答

この信号機の点滅している色は青です。

結果2: 日本語VLM (Vision Language Model)

Output

```
{
  Ours:  "この画像のキャラクターは「涼宮ハルヒ」です。彼女は人気アニメ「涼宮ハルヒの憂鬱」の主人公です。",
  JSVLM:  "アニメキャラクター",
  LLaVA-1.6-Mistral-7B:  "この画像のキャラクターは、アニメや漫画で知られているキャラクターです。しかし、具体的なキャラクター名前を提供することはできません。"
}
```

ただ日本語が流暢になっているということではなく、
元のVLM (LLaVa-1.6-Mistral-7B) が知らない知識を追加し利用出来ているかも？

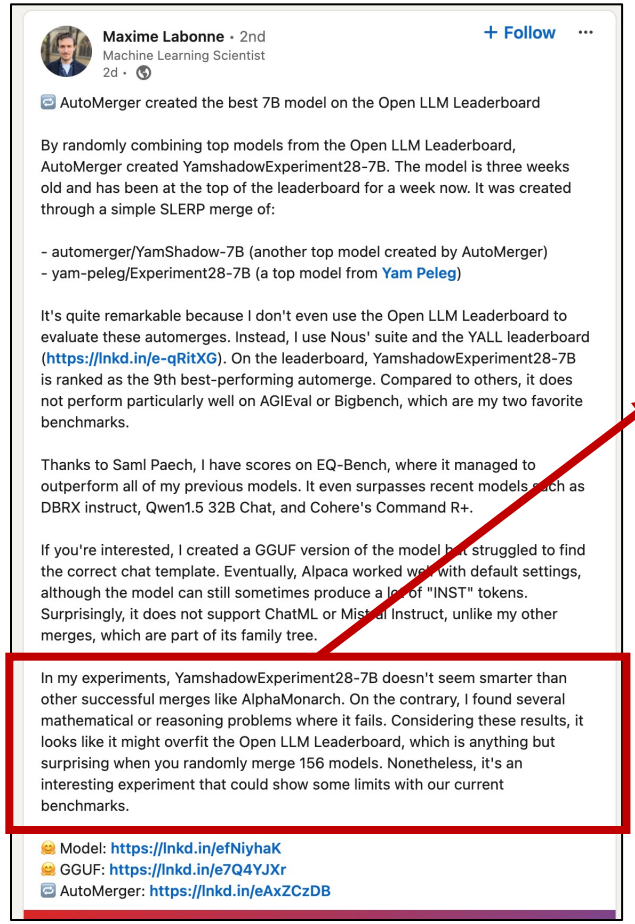
結果3: 日本語画像生成モデル

Id.	Model	Type	Inference steps	FID ↓	HPS ↑
1	SDXL 1.0	Original SDXL	40	57.70	18.66
2	Juggernaut-XL-v9	Fine-tuned SDXL	40	35.56	18.02
3	SDXL-DPO	RLHF SDXL	40	48.54	19.38
4	JSDXL	JA SDXL	40	18.20	22.78
5	SDXL-Lightning	4-step SDXL	4	42.56	21.62
6	**Ours (Base)**	1 + 2 + 3 + 4	40	**13.51**	**27.19**
7	**Ours (4-step)**	5 + 6	4	**15.88**	**28.85**

「波に乗っている1個のりんご」、「雲を泳ぐ魚のポートレート」と入力した生成結果

「折り紙味噌汁」、「折り紙弁当」と入力した生成結果

余談：過適合に注意



Maxime Labonne • 2nd
Machine Learning Scientist
2d •
+ Follow ···

AutoMerger created the best 7B model on the Open LLM Leaderboard

By randomly combining top models from the Open LLM Leaderboard, AutoMerger created YamshadowExperiment28-7B. The model is three weeks old and has been at the top of the leaderboard for a week now. It was created through a simple SLERP merge of:

- automerger/YamShadow-7B (another top model created by AutoMerger)
- yam-peleg/Experiment28-7B (a top model from Yam Peleg)

It's quite remarkable because I don't even use the Open LLM Leaderboard to evaluate these automerges. Instead, I use Nous' suite and the YALL leaderboard (https://lnkd.in/e-qRitXG). On the leaderboard, YamshadowExperiment28-7B is ranked as the 9th best-performing automerge. Compared to others, it does not perform particularly well on AGIEval or Bigbench, which are my two favorite benchmarks.

Thanks to Saml Paech, I have scores on EQ-Bench, where it managed to outperform all of my previous models. It even surpasses recent models such as DBRX instruct, Qwen1.5 32B Chat, and Cohere's Command R+.

If you're interested, I created a GGUF version of the model but struggled to find the correct chat template. Eventually, Alpaca worked well with default settings, although the model can still sometimes produce a lot of "INST" tokens. Surprisingly, it does not support ChatML or Mistral Instruct, unlike my other merges, which are part of its family tree.

In my experiments, YamshadowExperiment28-7B doesn't seem smarter than other successful merges like AlphaMonarch. On the contrary, I found several mathematical or reasoning problems where it fails. Considering these results, it looks like it might overfit the Open LLM Leaderboard, which is anything but surprising when you randomly merge 156 models. Nonetheless, it's an interesting experiment that could show some limits with our current benchmarks.

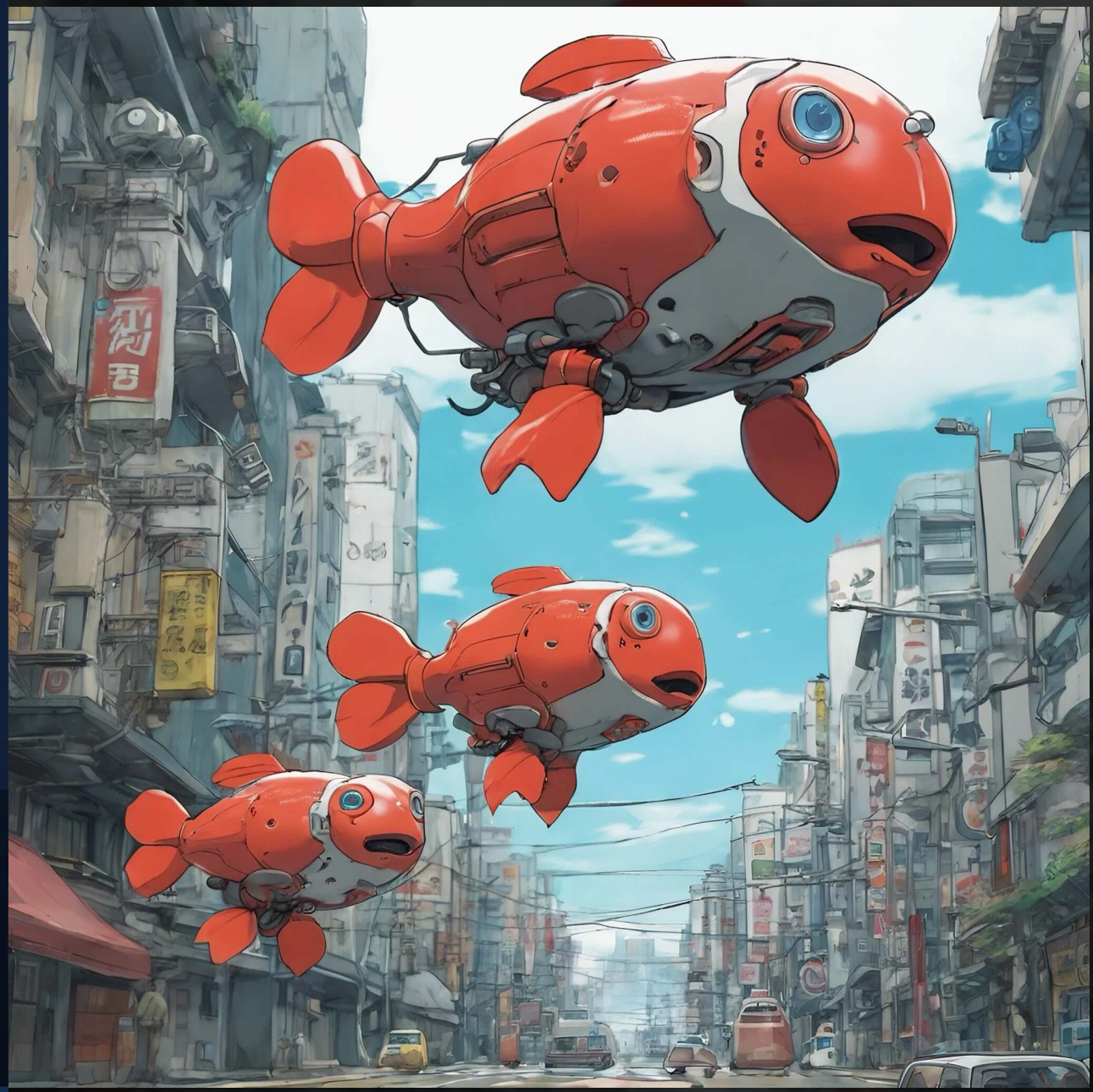
Model: https://lnkd.in/efNiyhaK
GGUF: https://lnkd.in/e7Q4YJXr
AutoMerger: https://lnkd.in/eAxZCzDB

In my experiments, YamshadowExperiment28-7B doesn't seem smarter than other successful merges like AlphaMonarch. On the contrary, I found several mathematical or reasoning problems where it fails. Considering these results, it looks like it might overfit the Open LLM Leaderboard, which is anything but surprising when you randomly merge 156 models. Nonetheless, it's an interesting experiment that could show some limits with our current benchmarks.

我々はtest dataは最適化時に利用していません。
実際、設定によりtrain dataに簡単に過適合するのを観測してました。
（なので、test scoreを見てマージしているOpen LLM Leaderboard勢は……）

3

おわりに



まとめ

1. モデルマージ

モデルマージのアプローチ2つ

何故マージするのか？何故上手くいくのか？

2. 進化的モデルマージ

進化的アルゴリズムを利用し、
今までにないカテゴリのモデルマージに成功