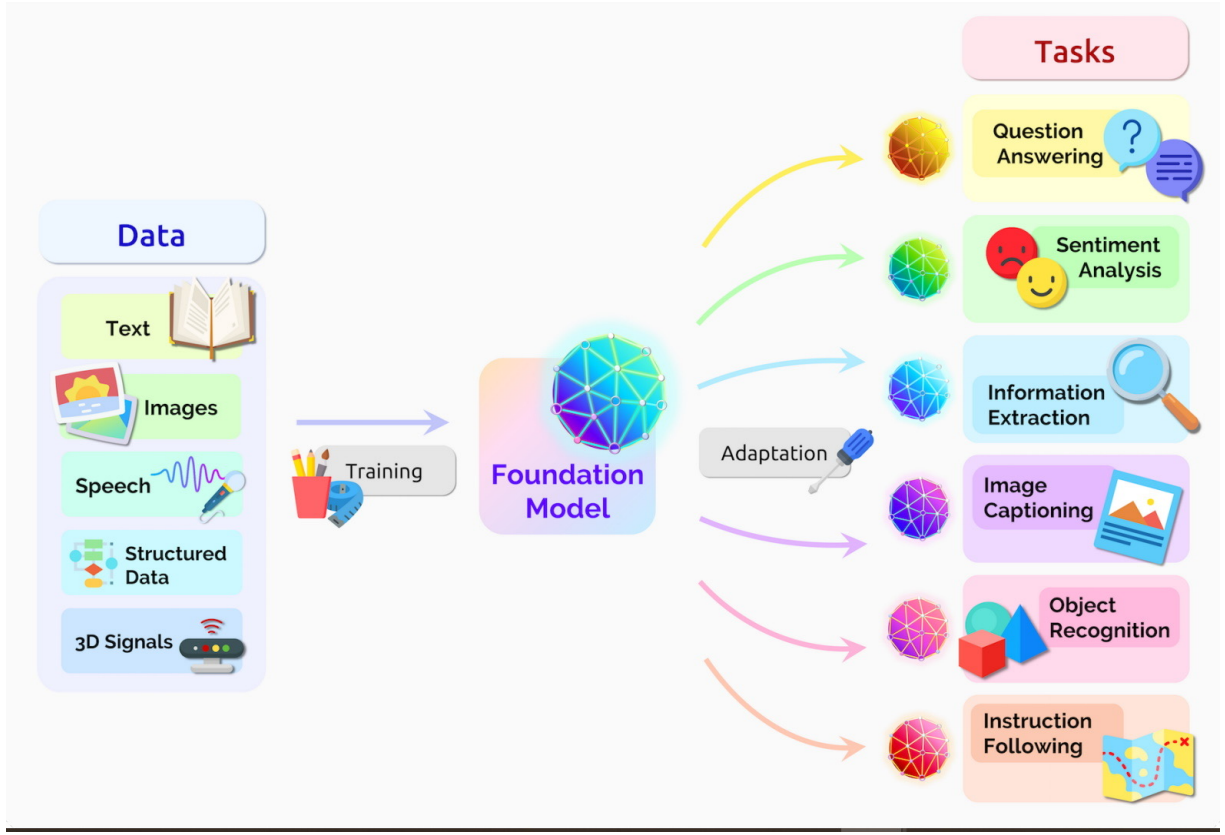
基盤モデル時代における Generalizable, Efficientな 画像認識モデルの構築

OMRON SINIC X
Senior Researcher
齋藤邦章

基盤モデルとは



- **(アノテーションコストの低い) 大量の**データから教師無しで学習
- 幅広いタスク、多様な入力に適応可能

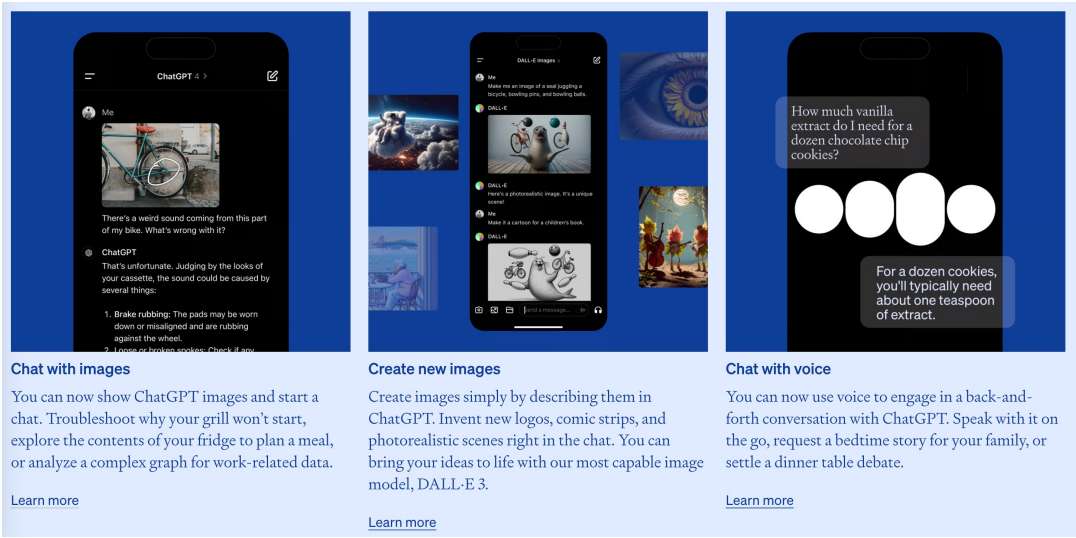
https://blogs.nvidia.co.jp/2023/06/19/what-are-foundation-models/

https://www.aidanscannell.com/project/amazon-picking-challenge/featured.jpg

基盤モデルの例

CLIP : Vision and Language

ChatGPT: Vision, language, and voice



- 様々なモデルが公開されている。
- Vision, Vision-languageタスクに使える

- APIを通してやりとり
- モデルパラメータは公開されていない

今日の話は認識の部分がメイン！

https://openai.com/research/clip,
https://openai.com/chatgpt

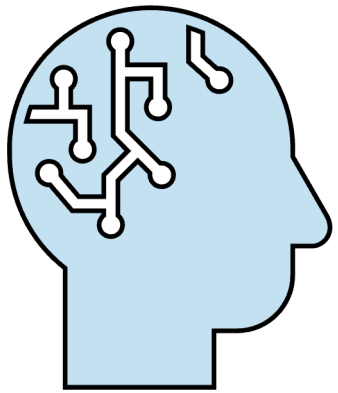
なぜ、今、基盤モデルがホットなのか

- Universalなモデルという考え方は昔からあった
- 基盤モデルのアイデアを実現した技術とは

なにが基盤モデルを支えているのか

事前学習

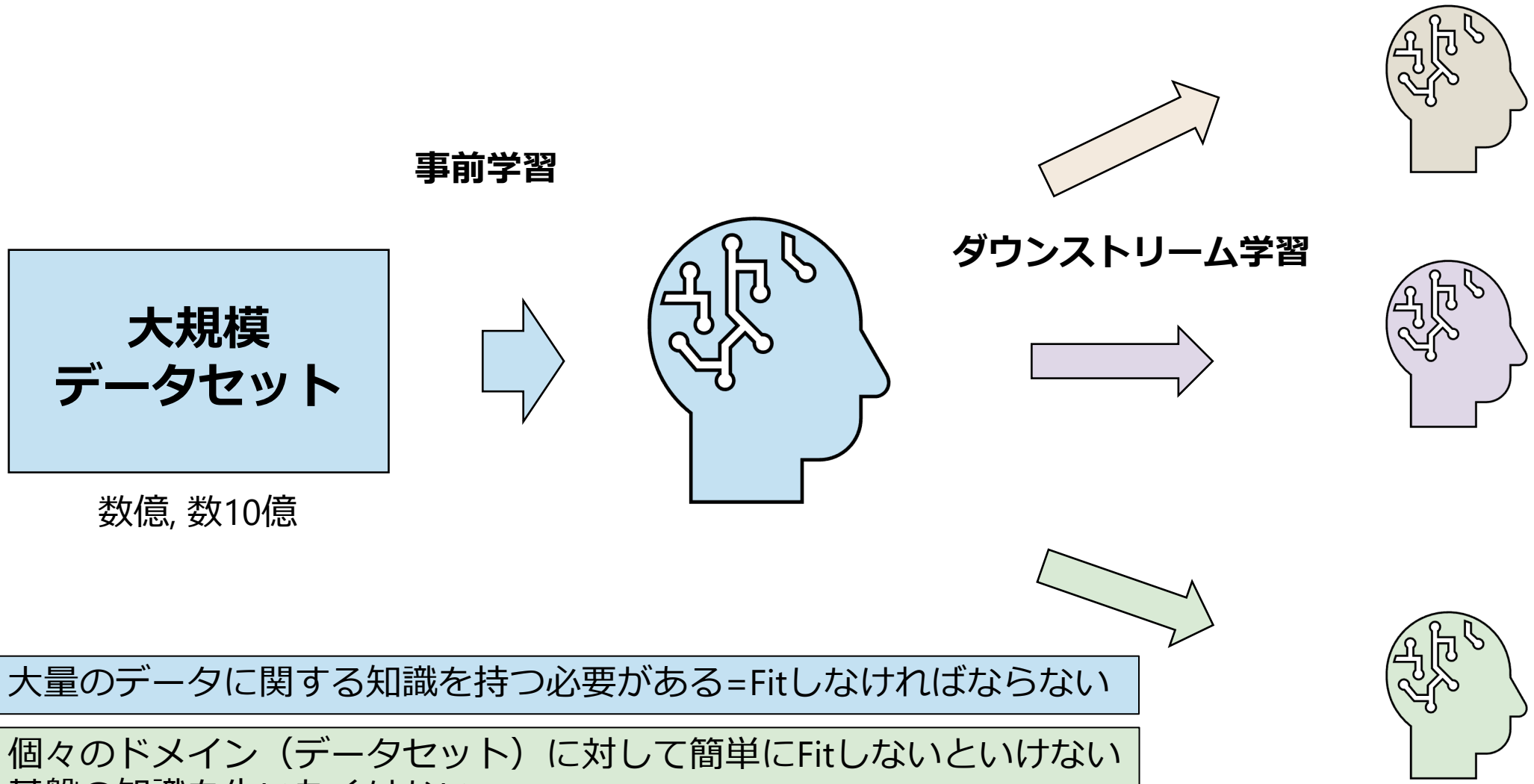
大規模
データセット



数億, 数10億

- 大量のデータに関する知識を持つ必要がある=Fitしなければならない

なにが基盤モデルを支えているのか



- 大量のデータに関する知識を持つ必要がある=Fitしなければならない

- 個々のドメイン（データセット）に対して簡単にFitしないといけない
- 基盤の知識を失いたくはない

なにが基盤モデルを支えているのか

- **事前学習**
  - 大量のデータ
    - 数億、数十億サンプル
  - Transformer等のアーキテクチャの進化

- **ダウンストリームタスクに適合する技術の進化**
  - パラメータEfficientな方法
  - ロバストな適合方法

なにが基盤モデルを支えているのか

- **事前学習**
  - 大量のデータ
    - 数億、数十億サンプル
  - Transformer等のアーキテクチャの進化

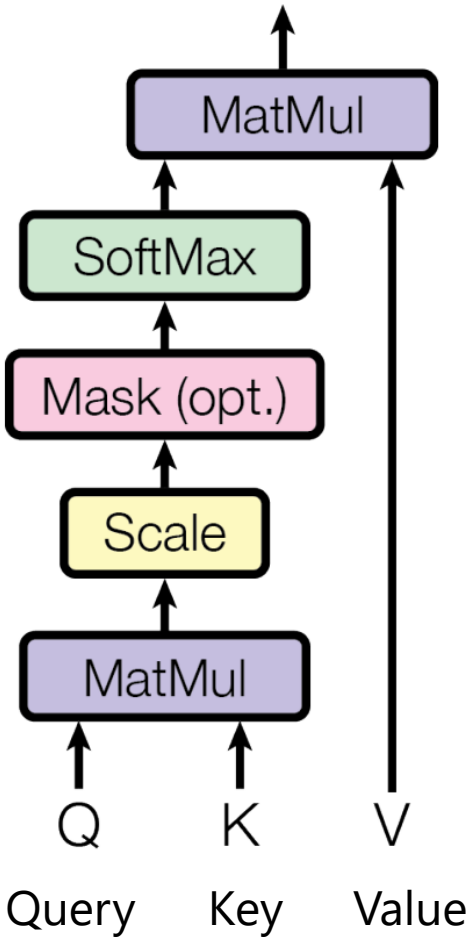
- **ダウンストリームタスクに適合する技術の進化**
  - パラメータEfficientな方法
  - ロバストな適合方法

Generalizeという視点を中心に!

Transformer

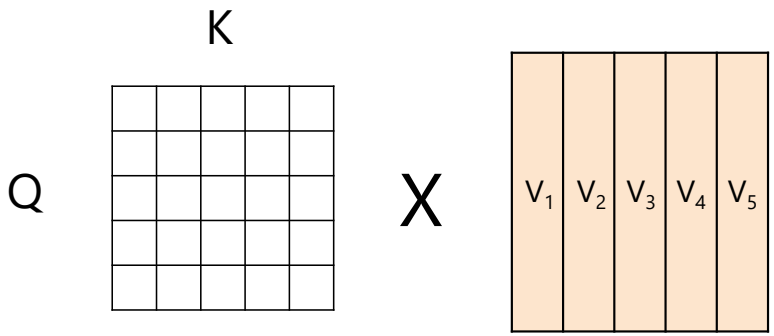
Transformer (Self-Attention)

Self-Attention



Q, K, V: 単語特徴量

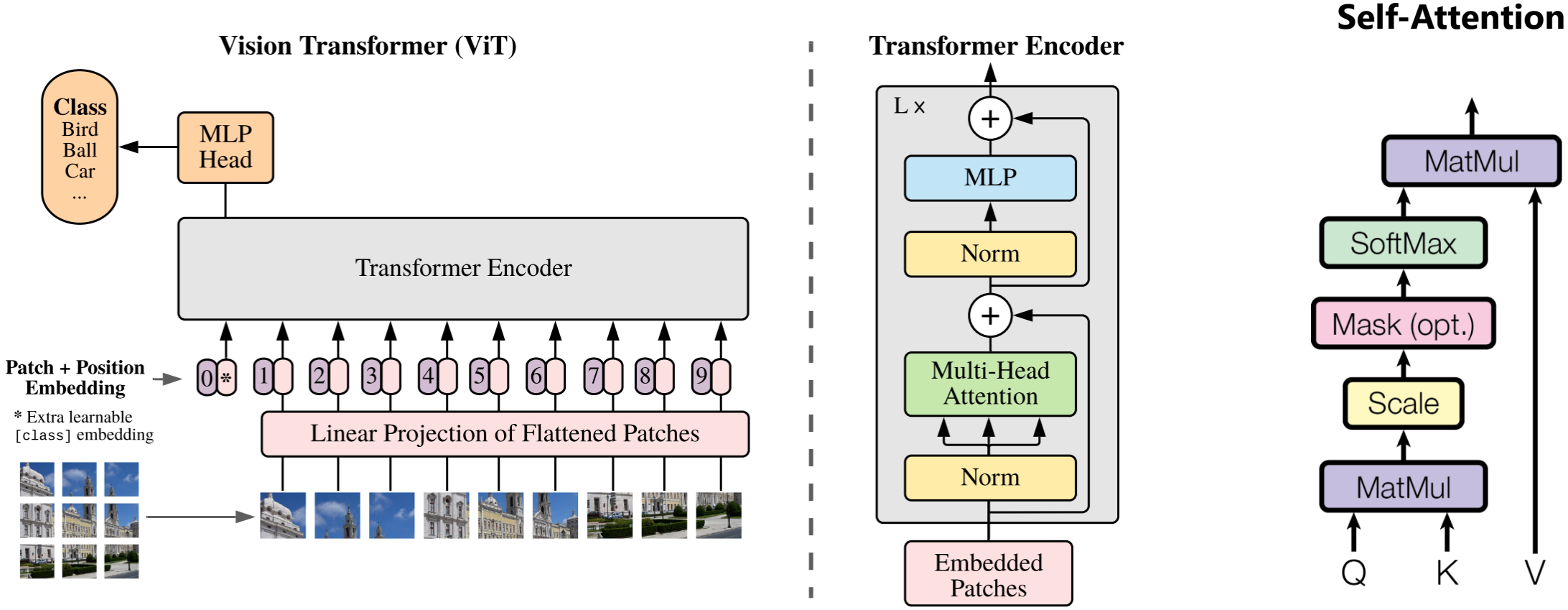
$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$



要するに

QueryとKeyとの類似度 (単語間類似度）に基づいて、
どうValueを持ってくるのか (特徴をアップデート)を決定する。

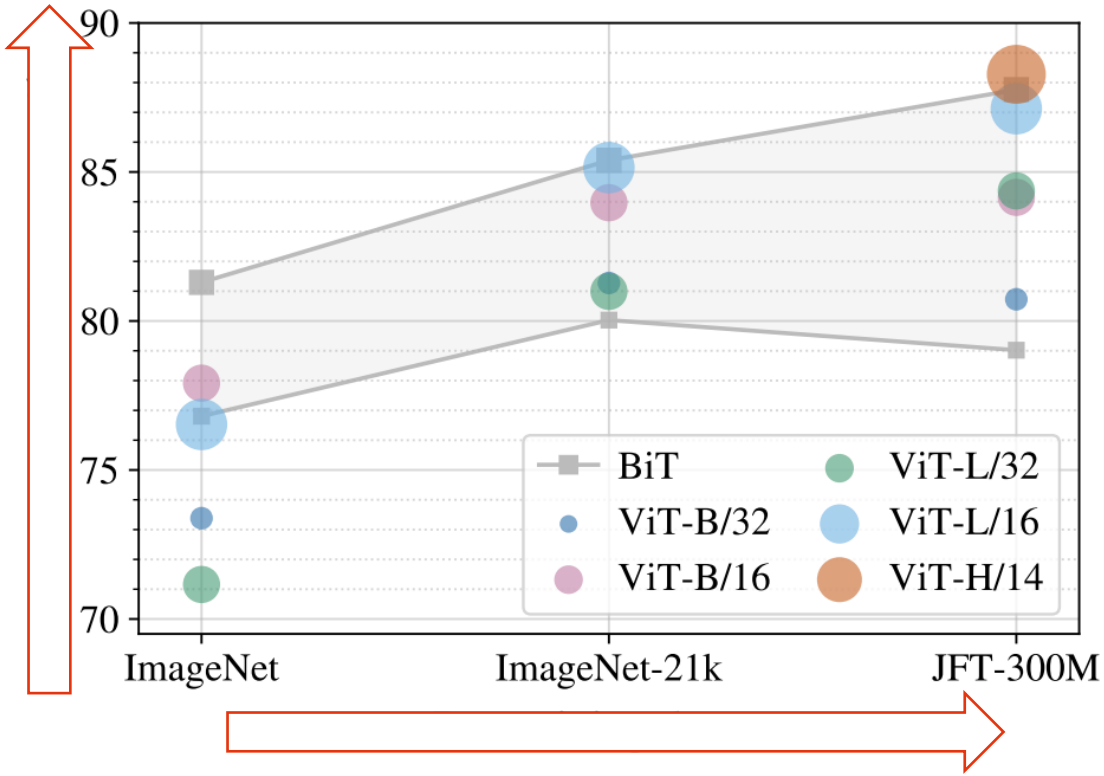
Vision Transformer (ViT)

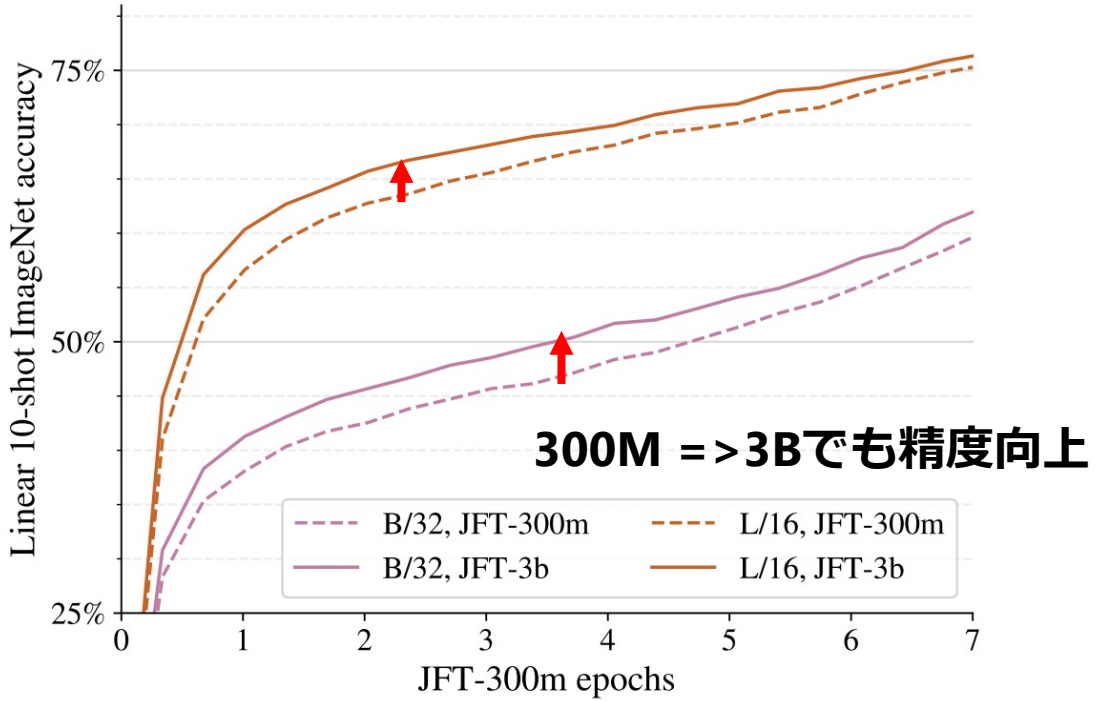


- Patchを1トークンとみなし、Self-Attentionによって他トークンと合わせてEncoding
- Convolutionと比較すると、それぞれのPatchから見たattention領域の範囲に制約がない

Data-hungryなViT

Accuracy

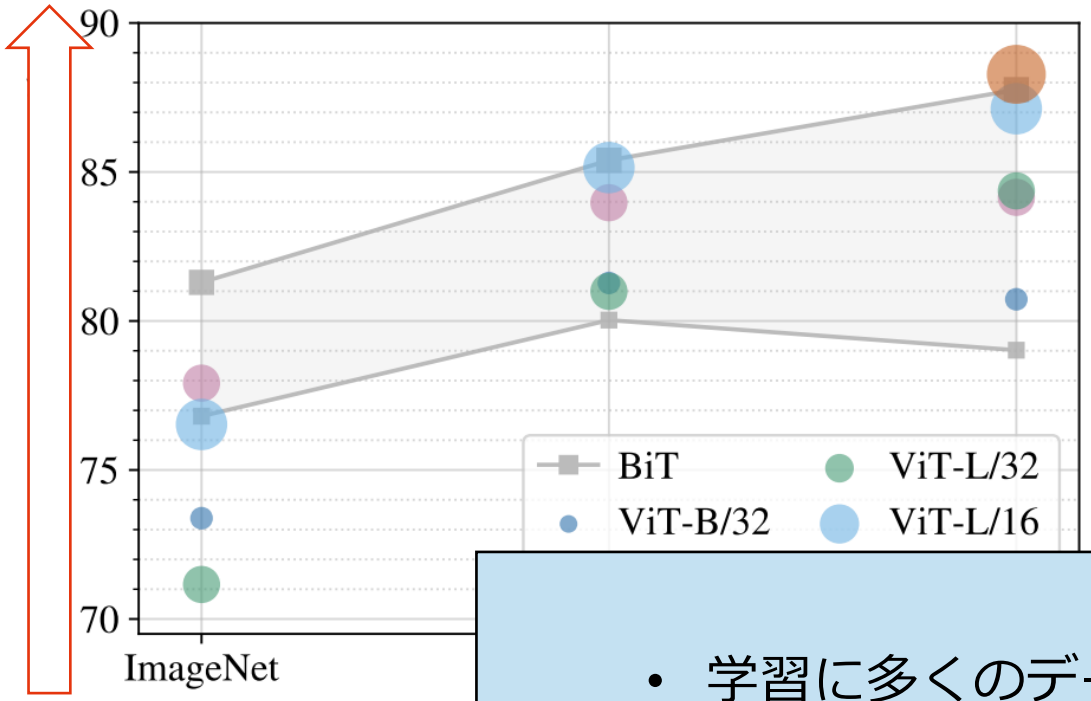


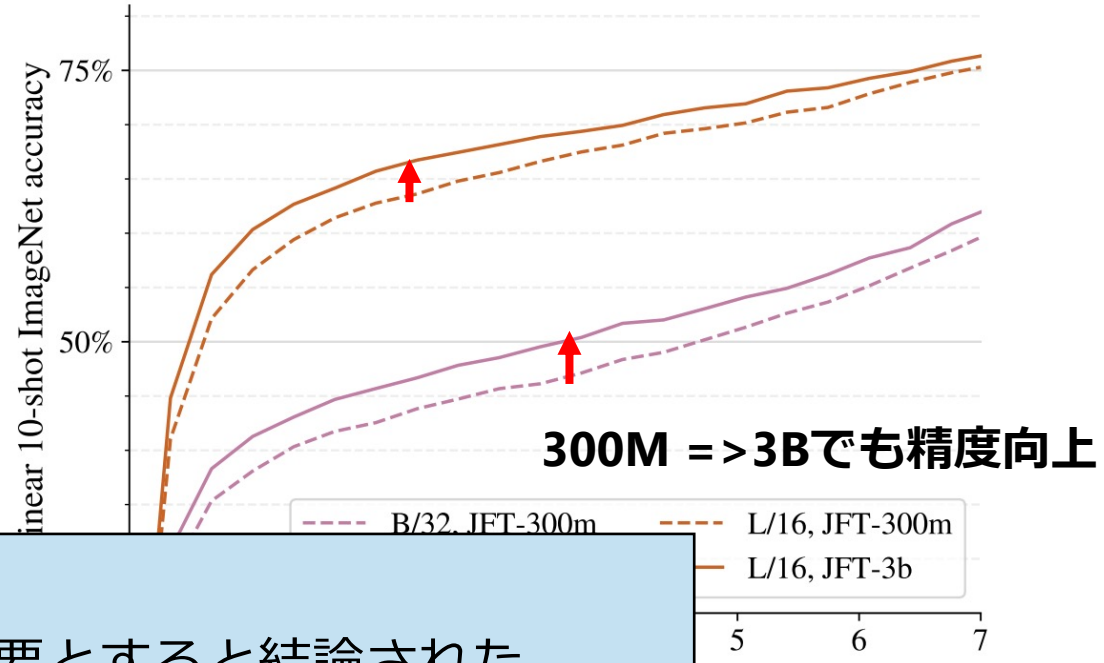


AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE, ICLR2021
Scaling Vision Transformers, CVPR2022

Data-hungryなViT

Accuracy





300M =>3Bでも精度向上

デ

- 学習に多くのデータを必要とすると結論された
- データサイズに対して精度がスケールするとも言えそう

AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE, ICLR2021
Scaling Vision Transformers, CVPR2022

ViTはRobustな識別器

- **そもそもRobustnessとは**
  - 学習時には与えられていないタイプの入力に対する精度

学習時

Cityscapes

COCO

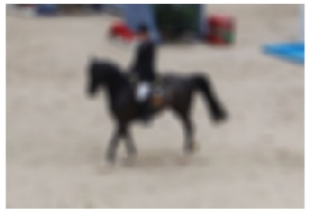
テスト時

Weather Change

Location Change

Gaussian Noise

Blur



Pixelate

ViTはRobustな識別器

SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers, NeurIPS2021

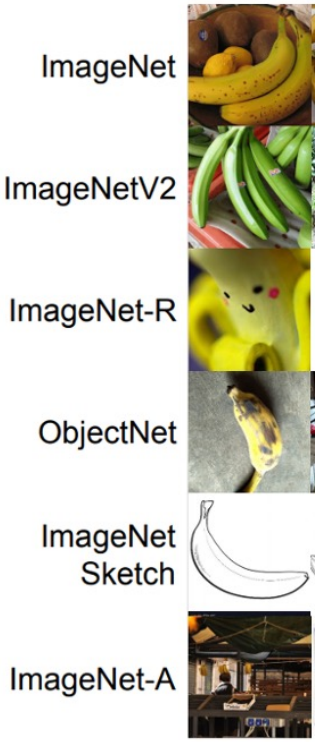
ConvnetとViT、どちらがRobustなのか

学習方法等	ResNet50	ViT
Augmentation	Flip, Crop	MixUp, CutMix
Optimizer	SGD	AdamW
Activation	ReLU	GELU

Are Transformers More Robust Than CNNs? NeurIPS2021

ConvnetとViT、どちらがRobustなのか

学習方法等	ResNet50	ViT
Augmentation	Flip, Crop	MixUp, CutMix
Optimizer	SGD	AdamW
Activation	ReLU	GELU



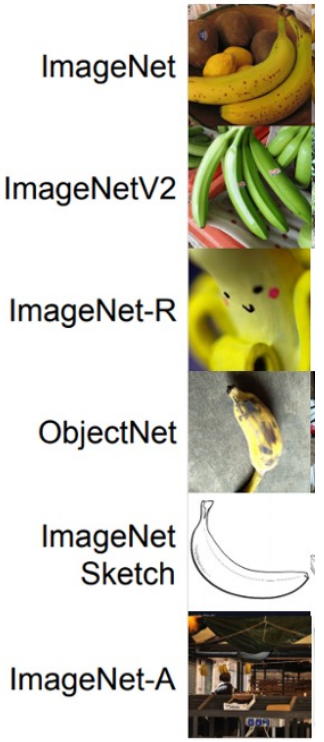
Optimizerに対する検証

	Optimizer-LR Scheduler	ImageNet ↑	ImageNet-A ↑	ImageNet-C ↓	Stylized-ImageNet ↑
ResNet-50	SGD-Step	76.9	3.2	57.9	8.3
	SGD-Cosine	77.4	3.3	56.9	8.4
	AdamW-Cosine	76.4	3.1	59.3	8.1
DeiT-S	AdamW-Cosine	76.8	**12.2**	**48.0**	**13.0**

Are Transformers More Robust Than CNNs? NeurIPS2021

ConvnetとViT、どちらがRobustなのか

学習方法等	ResNet50	ViT
Augmentation	Flip, Crop	MixUp, CutMix
Optimizer	SGD	AdamW
Activation	ReLU	GELU



Optimizerに対する検証

	Optimizer-LR Scheduler	ImageNet ↑	ImageNet-A ↑	ImageNet-C ↓	Stylized-ImageNet ↑
ResNet-50	SGD-Step	76.9	3.2	57.9	8.3
	SGD-Cosine	77.4	3.3	56.9	8.4
	AdamW-Cosine	76.4	3.1	59.3	8.1
DeiT-S	AdamW-Cosine	76.8	**12.2**	**48.0**	**13.0**

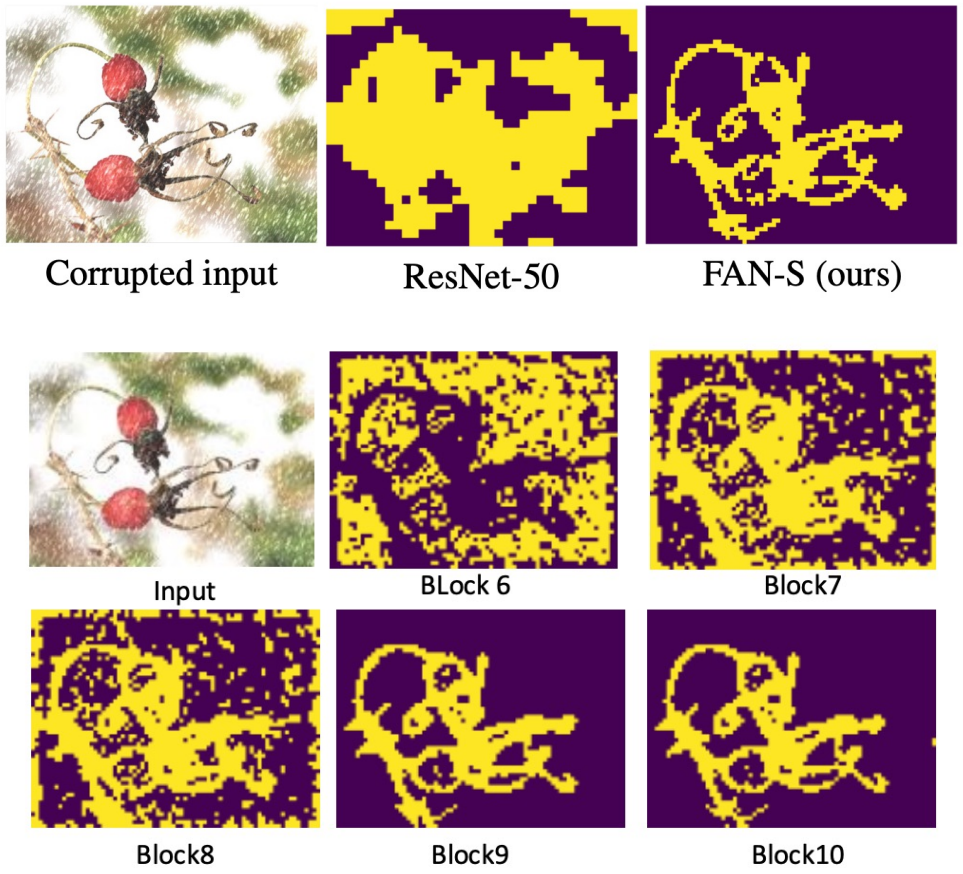
結論

- Self-attentionに起因しない、学習レシピの部分でViTが強く見せられている部分があった。
- しかし、学習レシピを揃えてもViTが強いケースは存在する。
- Self-attentionに起因するロバストさがあるのではないか？

Are Transformers More Robust Than CNNs? NeurIPS2021

Self-Attention=Clustering→Robust?

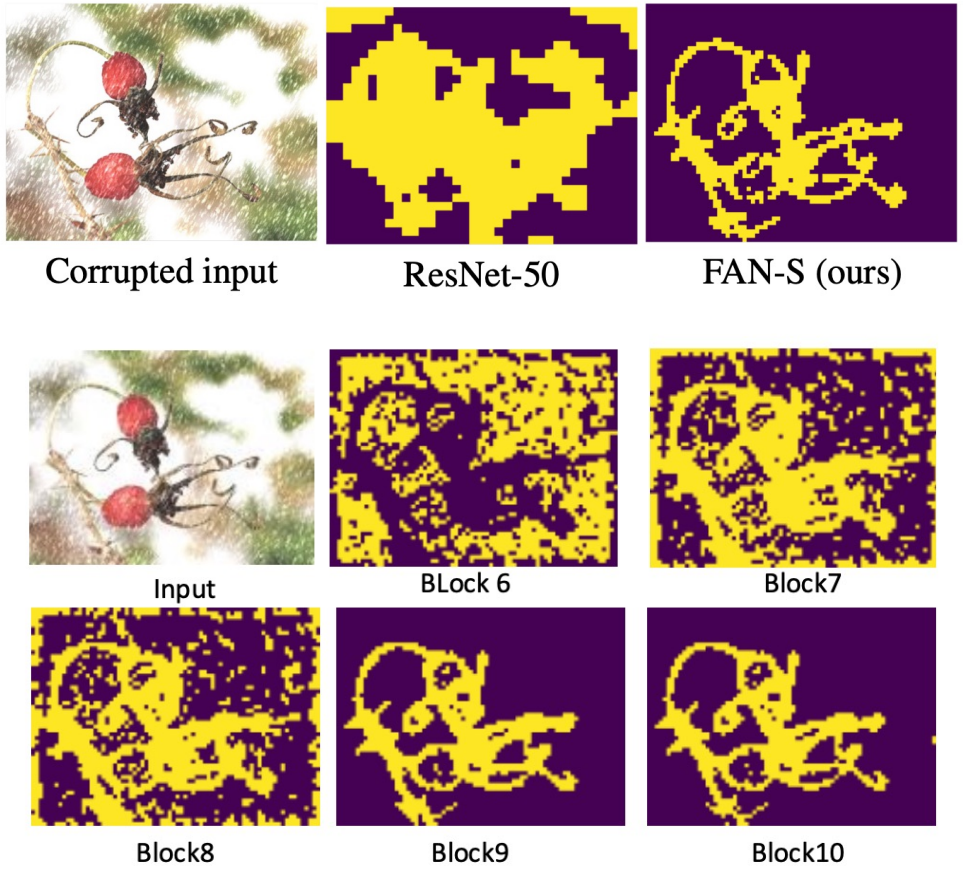
背景 vs 全景の Clusterが出現する



Understanding The Robustness in Vision Transformers, ICML2022

Self-Attention=Clustering→Robust?

背景 vs 全景の Clusterが出現する



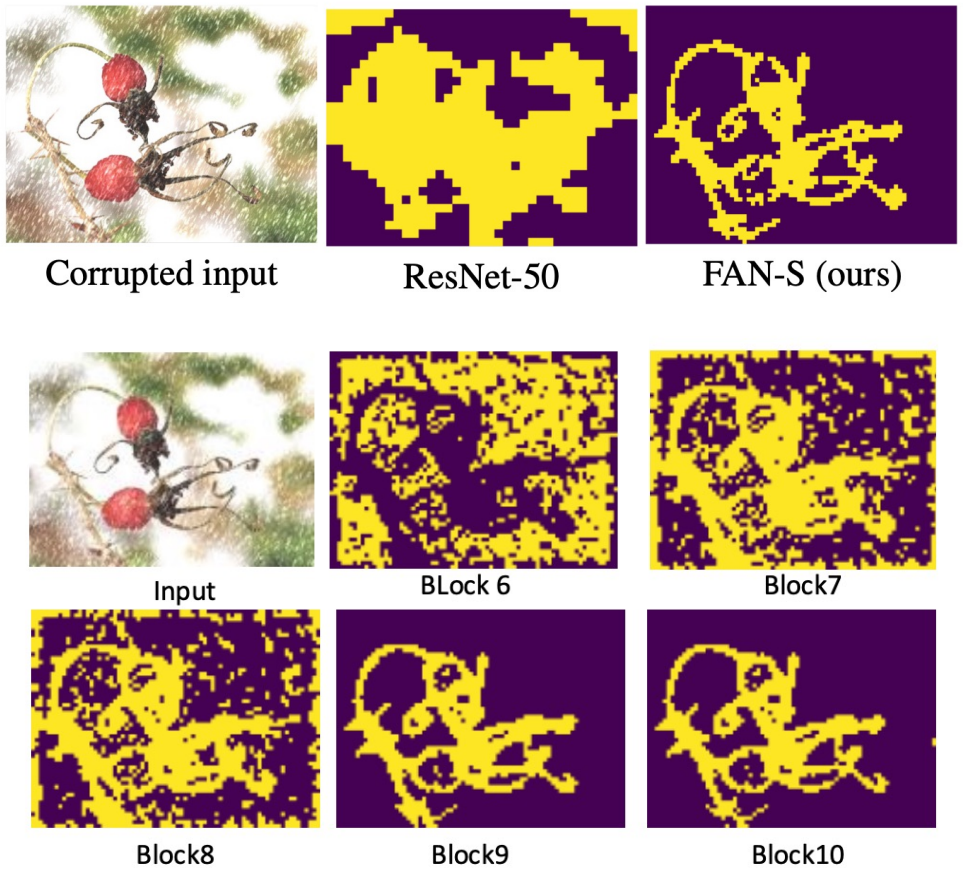
Self-attention Layer

$$Z^\top = \mathrm{SA}(X) = \mathrm{Softmax}\left(\frac{Q^\top K}{\sqrt{d}}\right) V^\top W_L,$$

Understanding The Robustness in Vision Transformers, ICML2022

Self-Attention=Clustering→Robust?

背景 vs 全景の Clusterが出現する



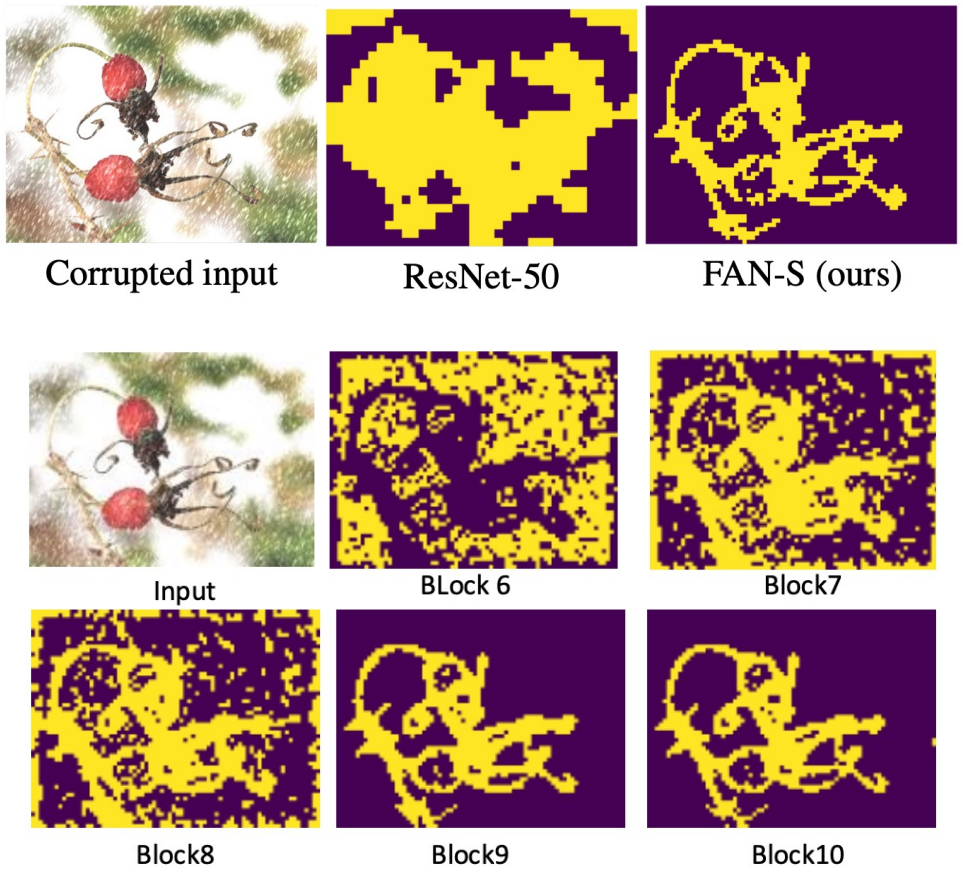
Self-attention Layer

$$Z^\top = \mathrm{SA}(X) = \underbrace{\mathrm{Softmax}\left(\frac{Q^\top K}{\sqrt{d}}\right)}_{\text{Clusterへの割り当てを計算}} \underbrace{V^\top W_L}_{\text{各Clusterの特徴量}},$$

Understanding The Robustness in Vision Transformers, ICML2022

Self-Attention=Clustering→Robust?

背景 vs 全景の Clusterが出現する



Self-attention Layer

$$Z^\top = \mathrm{SA}(X) = \mathrm{Softmax}\left(\frac{Q^\top K}{\sqrt{d}}\right) V^\top W_L,$$

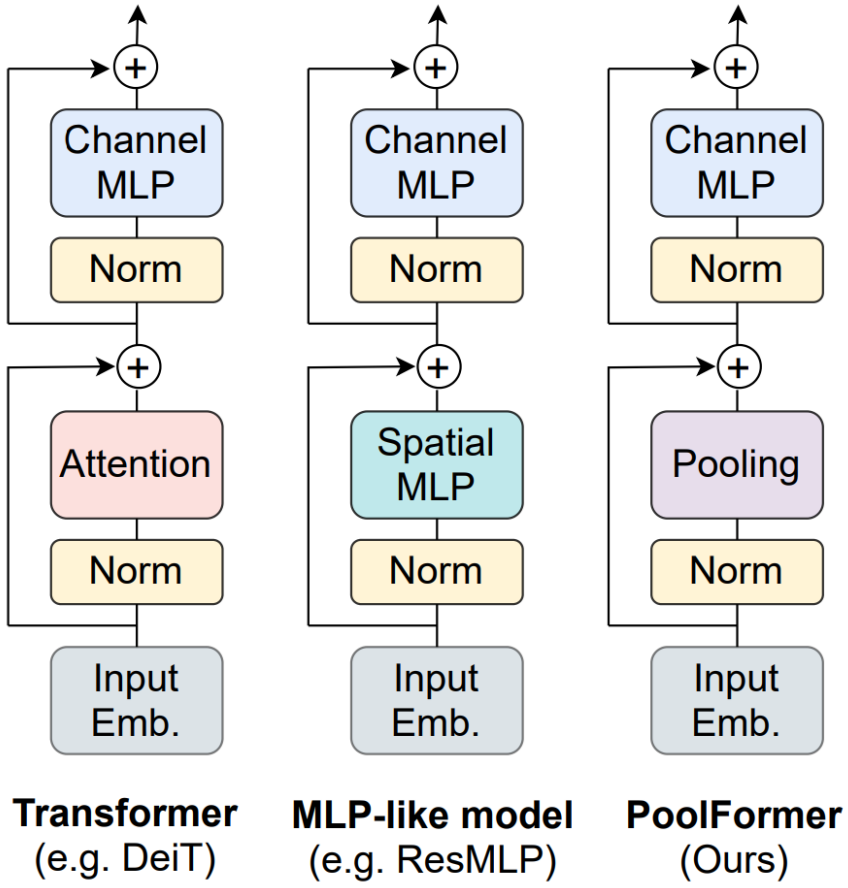
Clusterへの割り当てを計算　　各Clusterの特徴量

雑にまとめると

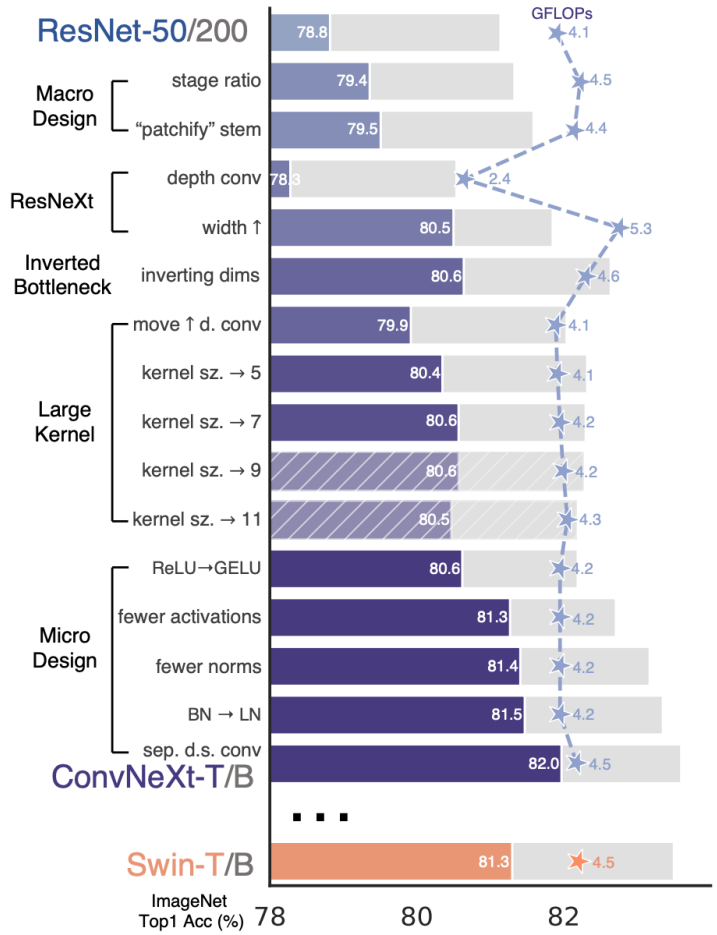
Input -> 各Clusterの特徴でZを求める -> ノイズ除去

Understanding The Robustness in Vision Transformers, ICML2022

本当にSelf-Attentionが良いのか



MetaFormer Is Actually What You Need for Vision, CVPR2022



A ConvNet for the 2020s, CVPR2022

ここまでのまとめ

- **Architectureの進化が実現したもの**
  - 大量のデータにフィットできる
  - 汎化性能が高い
- **ViT > ConvNetかどうかは結論できなそう**

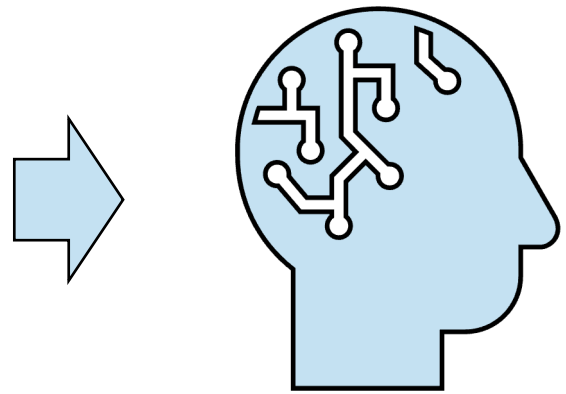
基盤モデルのチューニング: Parameter Efficientな方法

ダウンストリームタスクに
特化したモデル

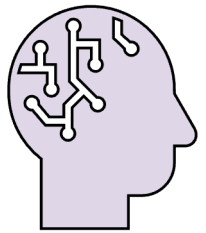
事前学習

大規模
データセット

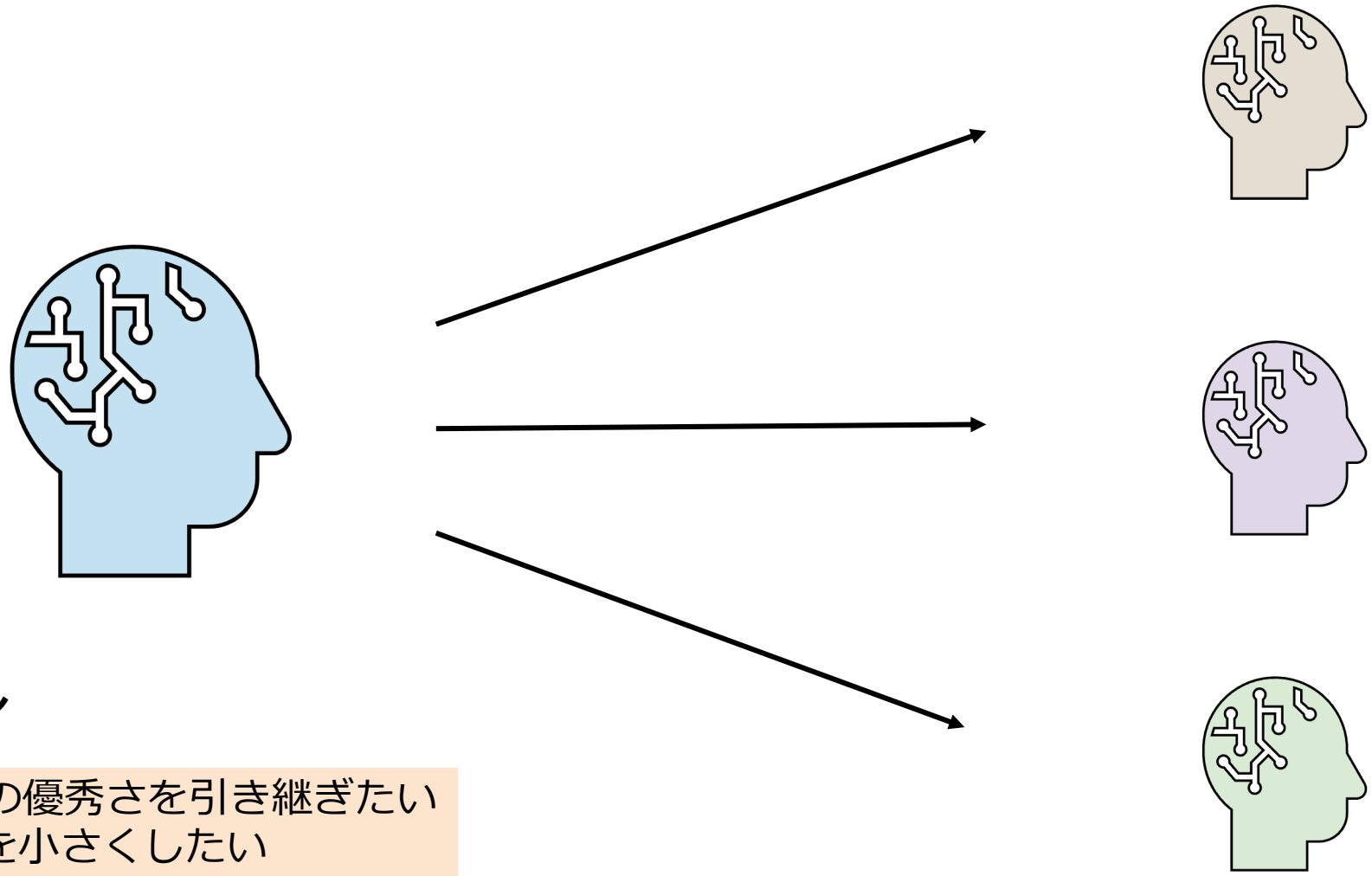
数億, 数10億



ダウンストリーム学習



ダウンストリームタスクに
特化したモデル

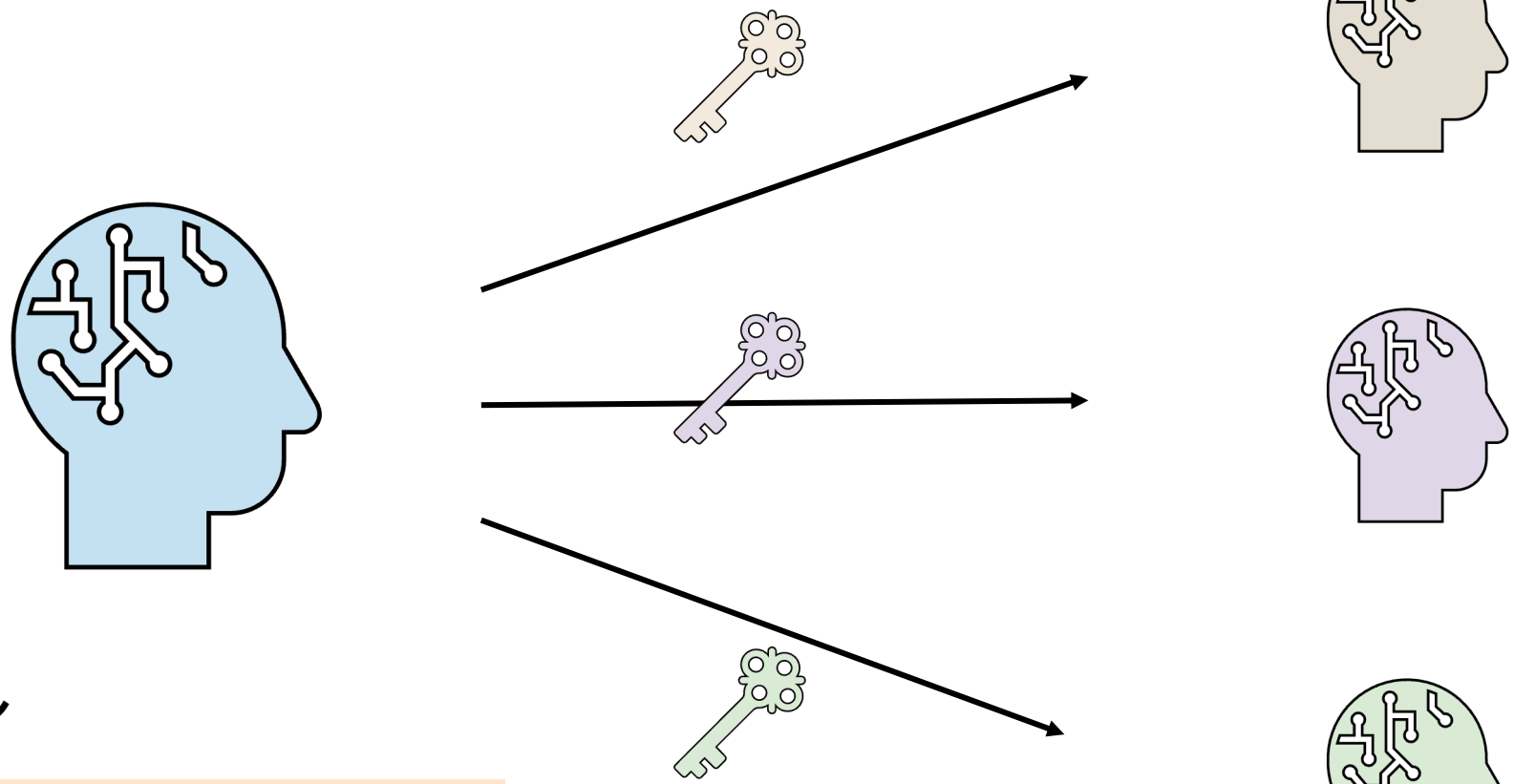


モチベーション

- 基盤モデルの優秀さを引き継ぎたい
- 学習コストを小さくしたい

ダウンストリームタスクに
特化したモデル

少量の学習パラメータ

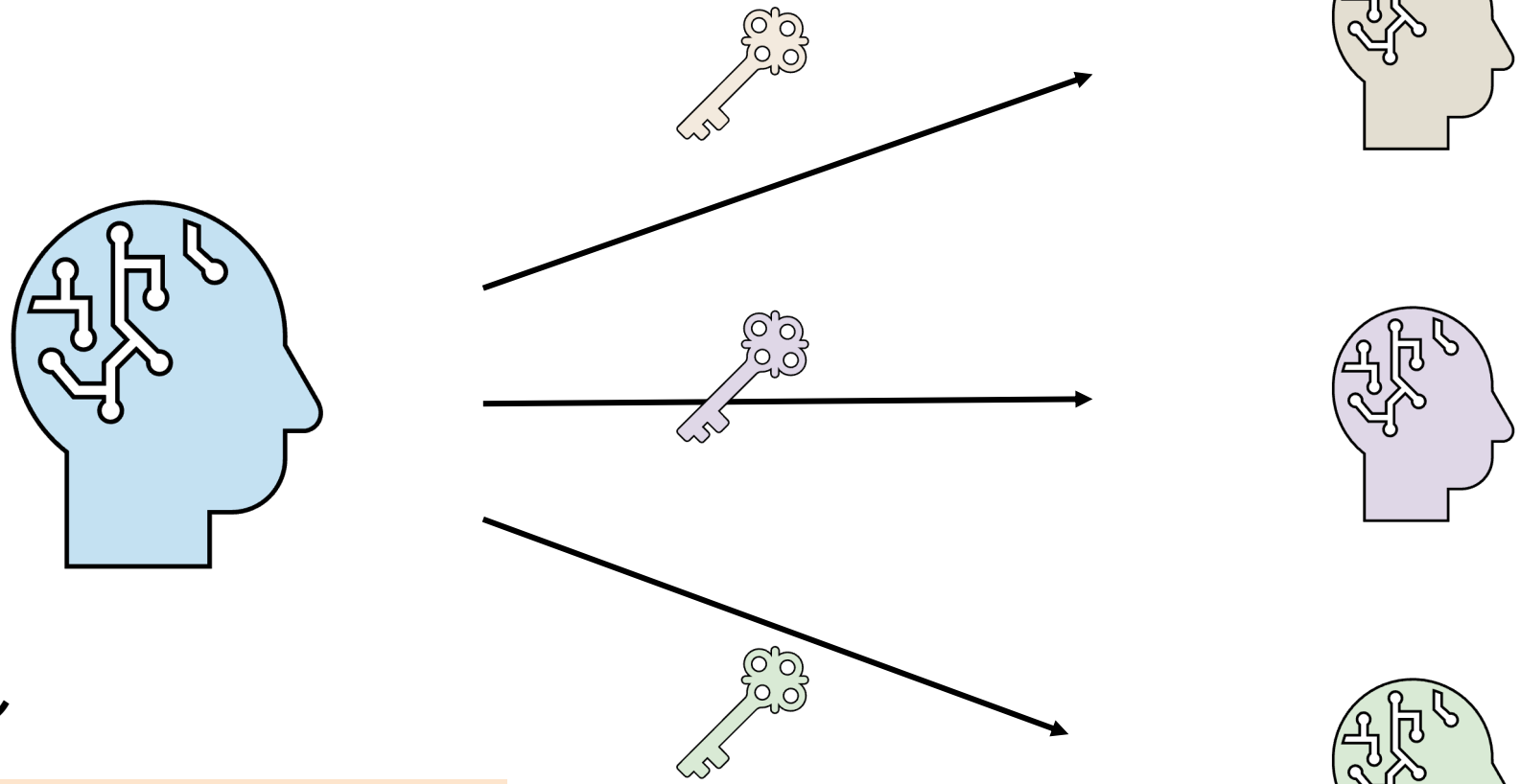


モチベーション

- 基盤モデルの優秀さを引き継ぎたい
- 学習コストを小さくしたい

ダウンストリームタスクに
特化したモデル

少量の学習パラメータ

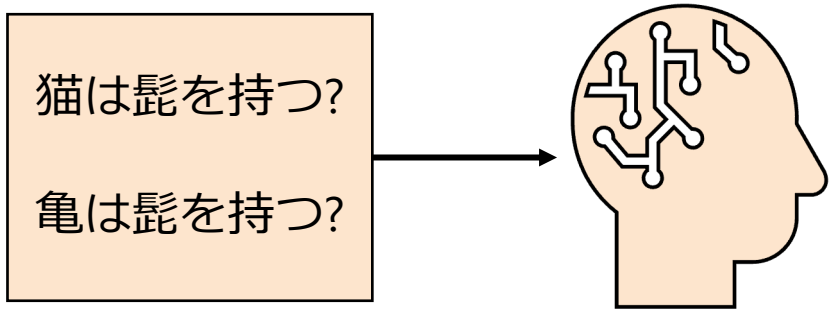


モチベーション

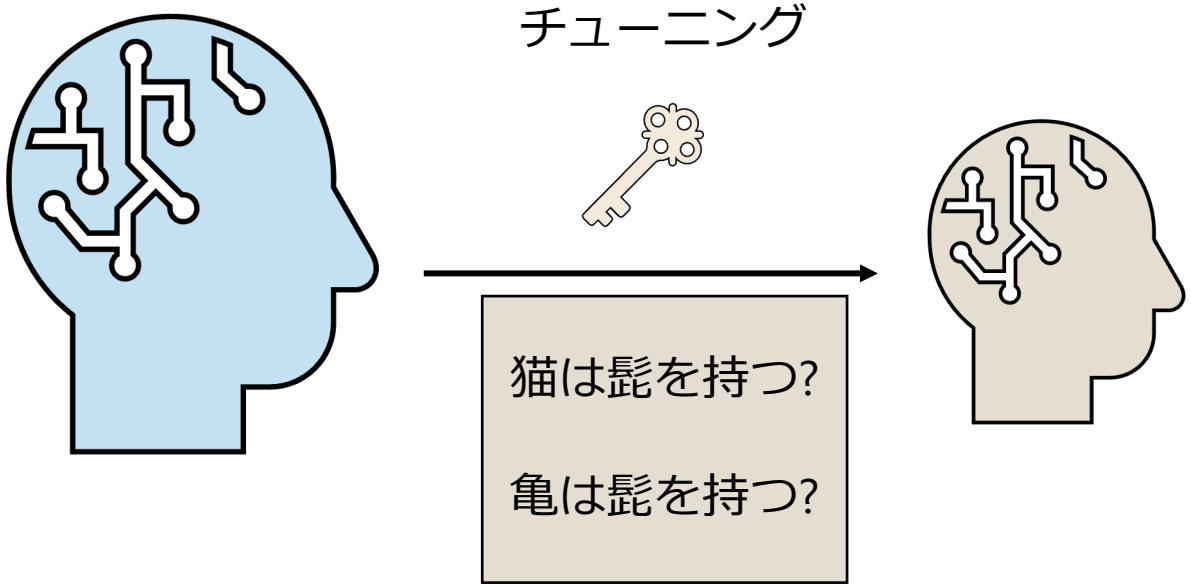
- **基盤モデルの優秀さ**を引き継ぎたい
- 学習コストを小さくしたい

基盤モデルの優秀さとは

Yes or No, QAデータセット

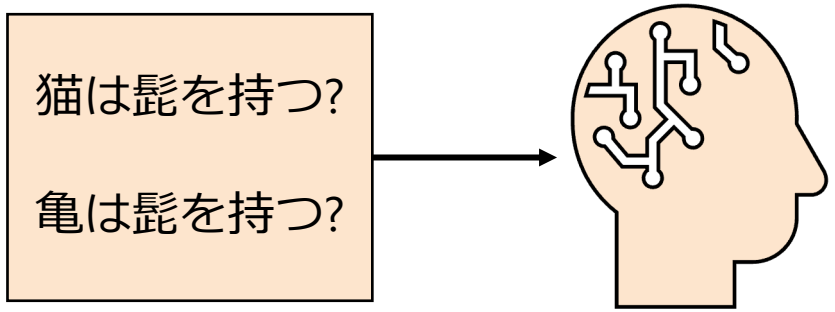


チューニング

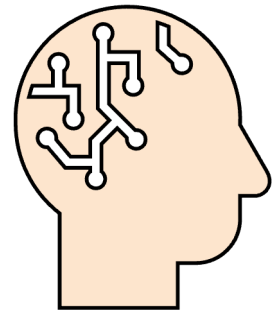


基盤モデルの優秀さとは

Yes or No, QAデータセット

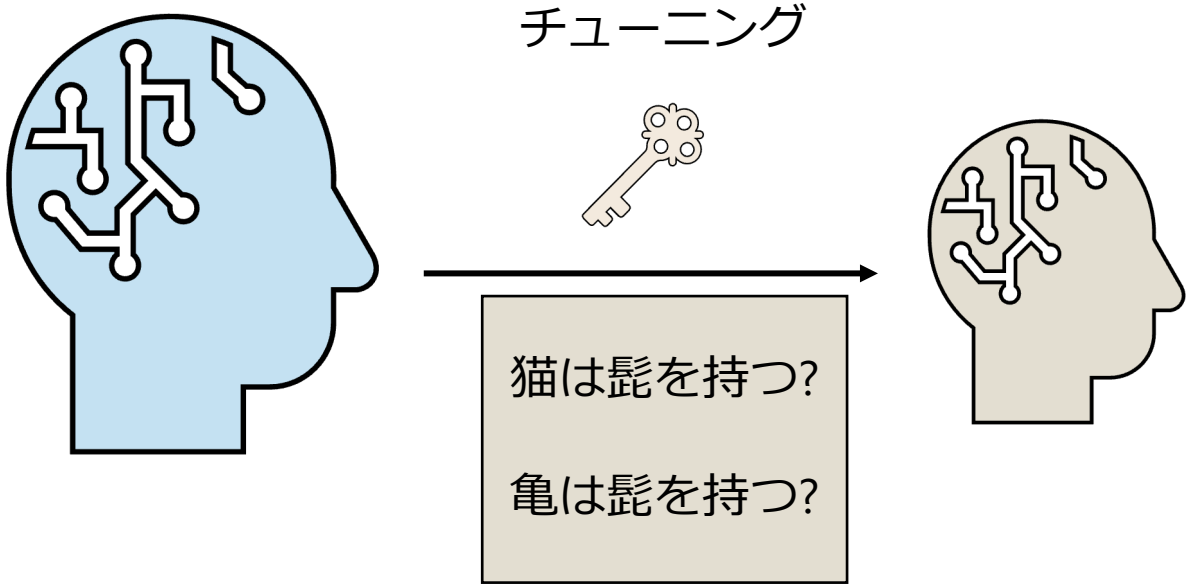


チーターは髭を持つ



???

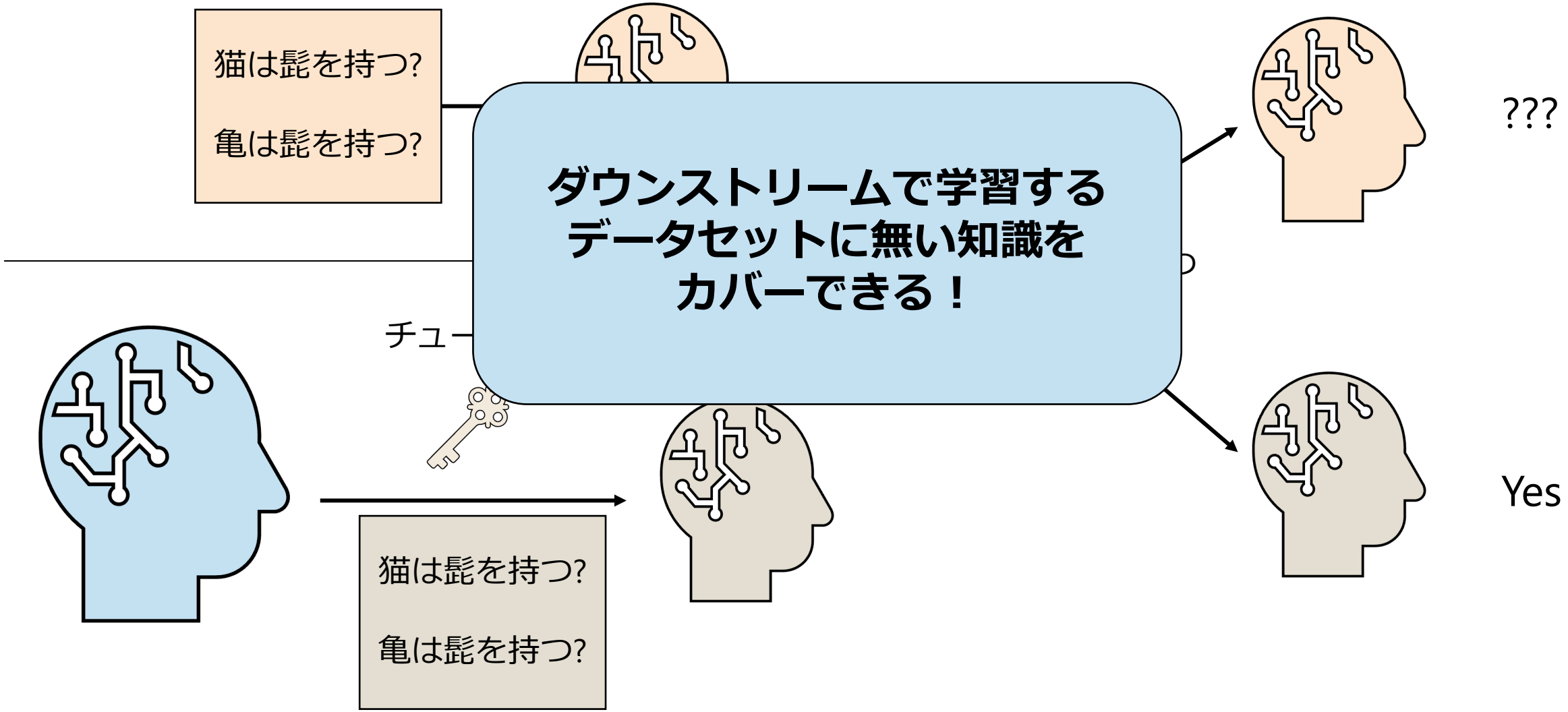
チューニング



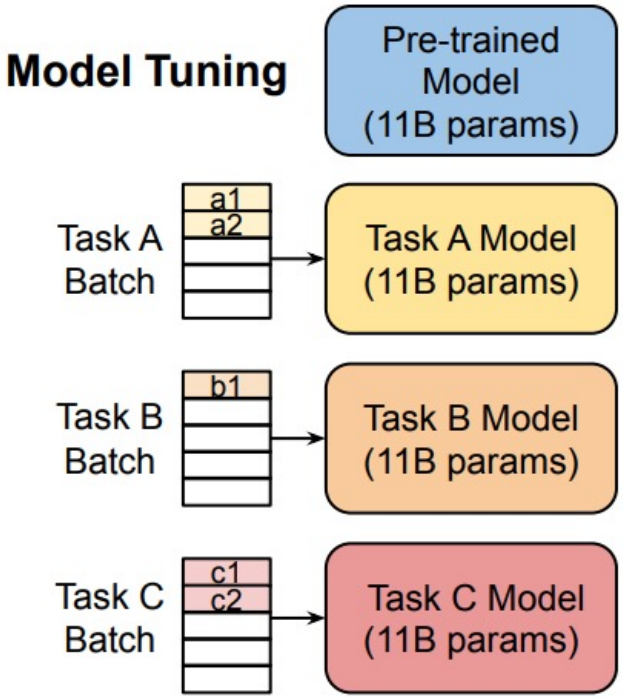
Yes

基盤モデルの優秀さとは

Yes or No, QAデータセット

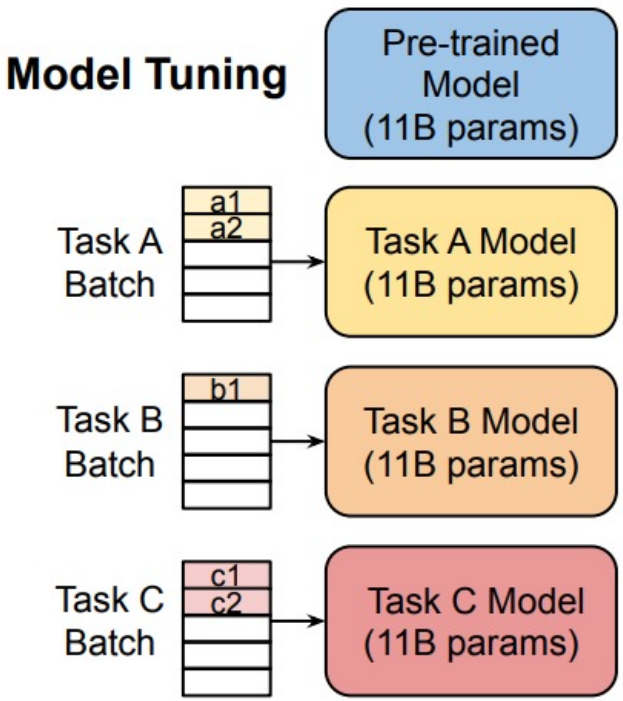


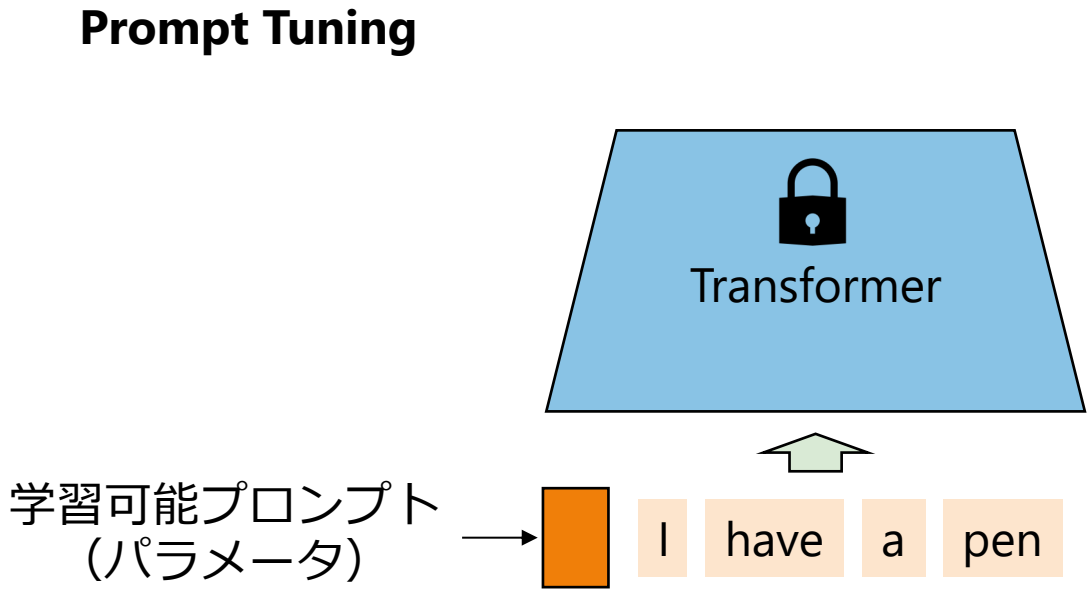
Taskを切り替えるPrompt学習



The Power of Scale for Parameter-Efficient Prompt Tuning, EMNLP 2021

Taskを切り替えるPrompt学習

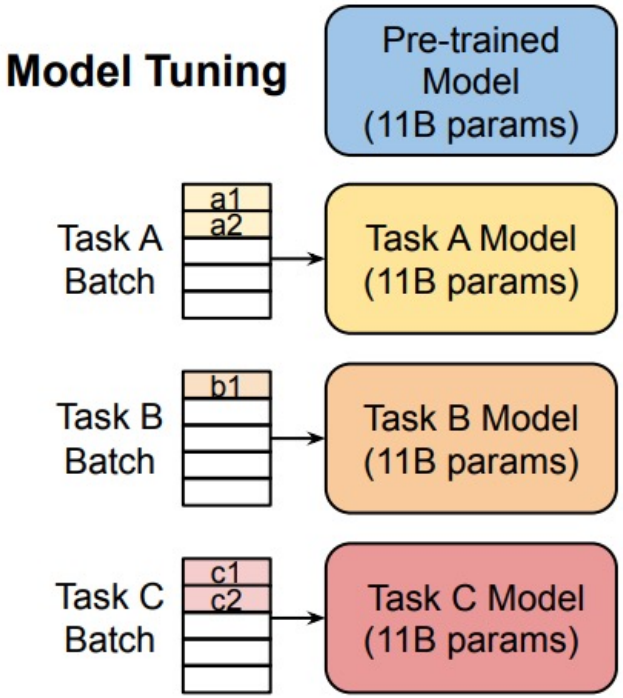


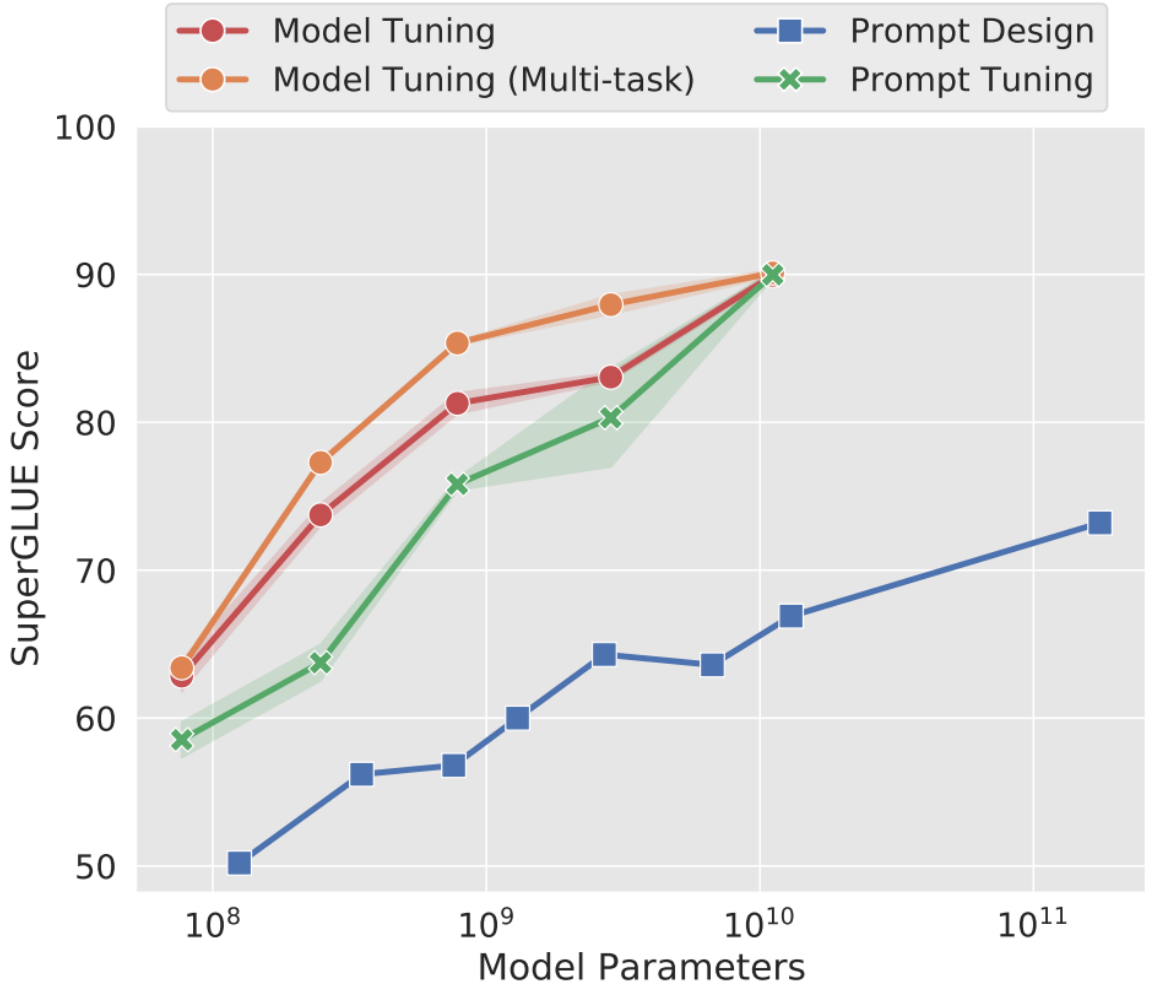


- 少量パラメータの学習でDownstreamタスクへの学習を可能にする
- モデルのタスクを切り替えるためのスイッチとも見なせる

The Power of Scale for Parameter-Efficient Prompt Tuning, EMNLP 2021

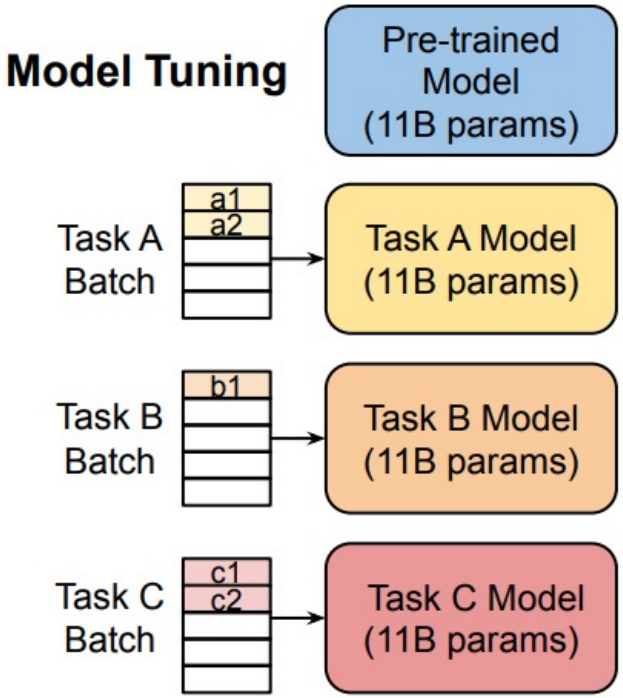
Taskを切り替えるPrompt学習

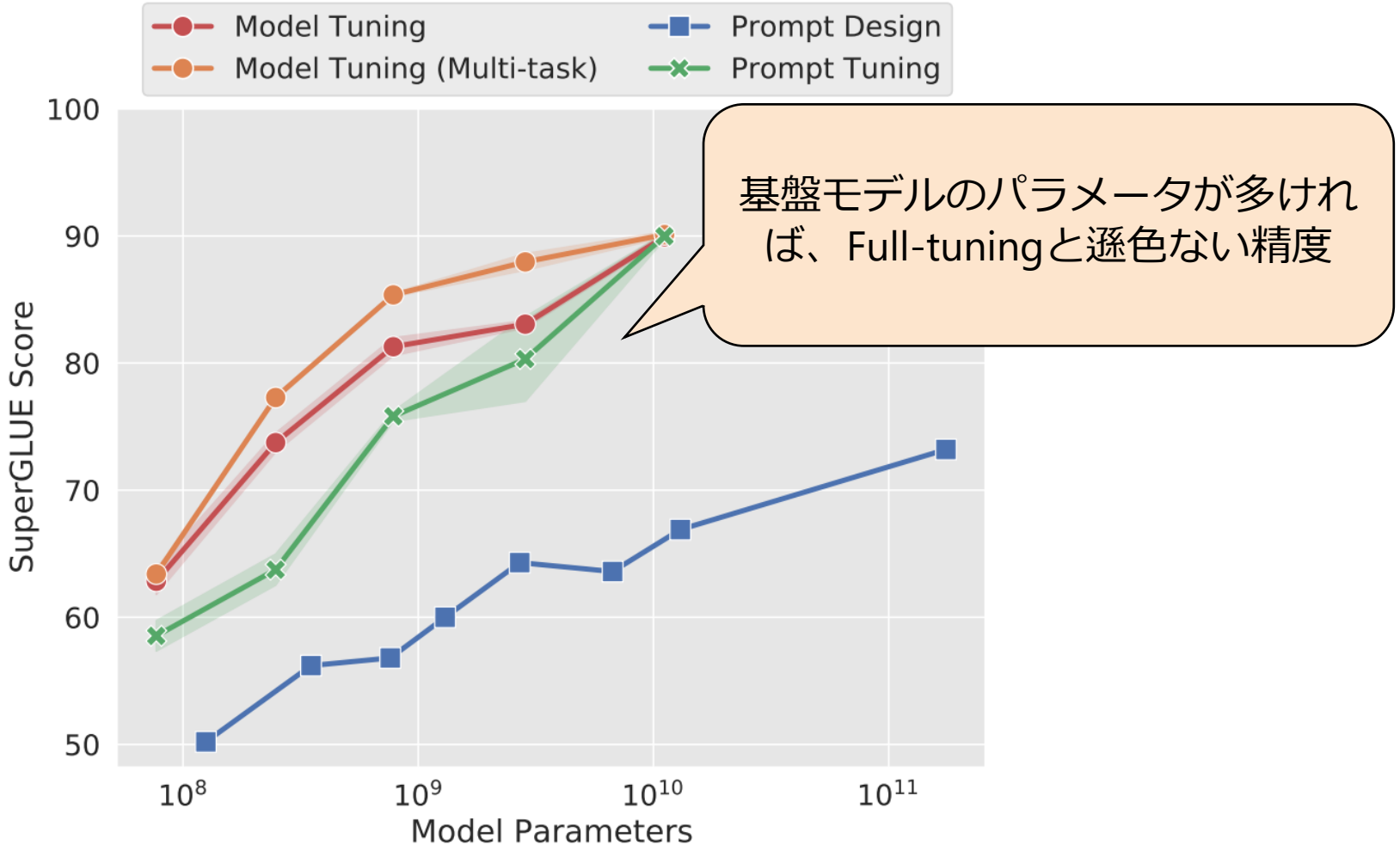




The Power of Scale for Parameter-Efficient Prompt Tuning, EMNLP 2021

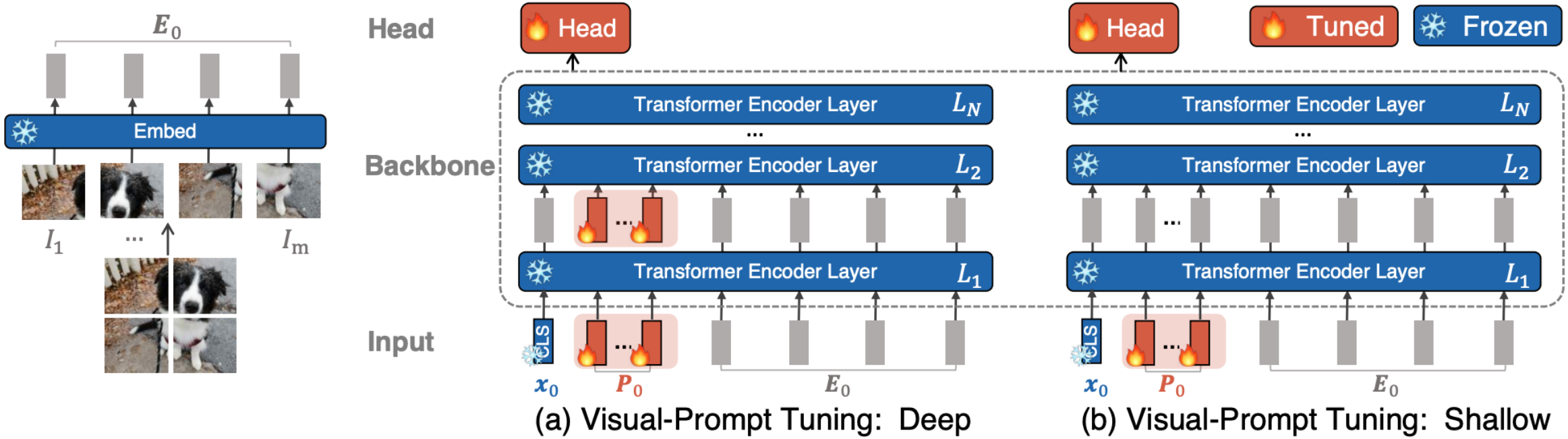
Taskを切り替えるPrompt学習





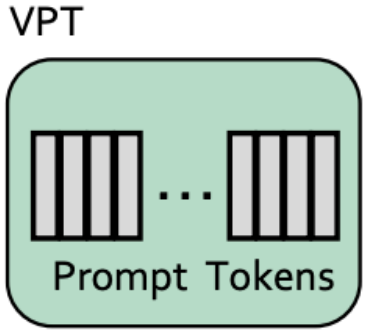
The Power of Scale for Parameter-Efficient Prompt Tuning, EMNLP 2021

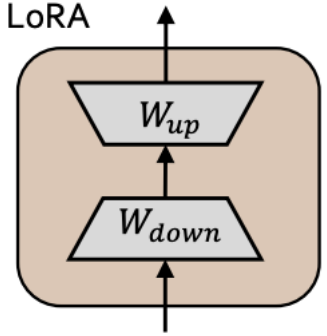
Visual Prompt Tuning: ViT用のPrompt学習

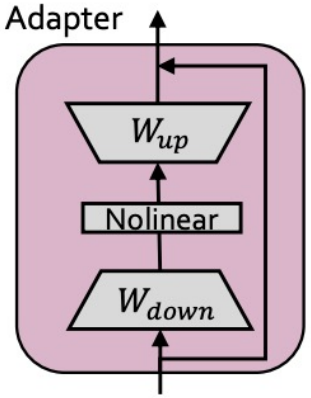


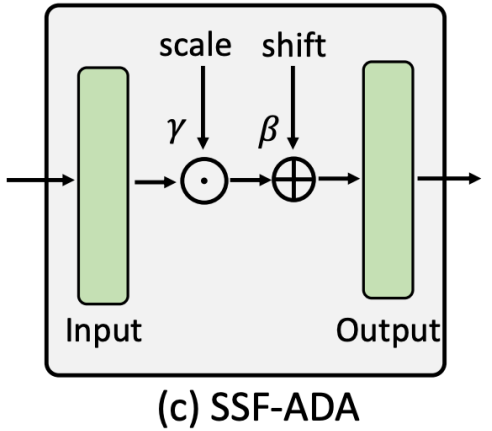
Visual Prompt Tuning, ECCV2022

様々なParameter-EfficientなTuning手法









- Prompt Tuning
- Input Layer

- 低ランクな行列アップデート
- MLP, Self-Attention層

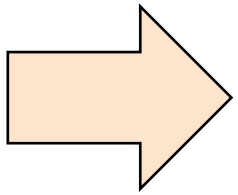
- スケールパラメータの学習
- 層と層の間

導入するモジュールの数（Promptの数等）を変えることで、アップデートをコントロールする。

Neural Prompt Search, Arxiv 2022
Scaling & Shifting Your Features: A New Baseline for Efficient Model Tuning, NeurIPS2022

結局どれが一番いいのか

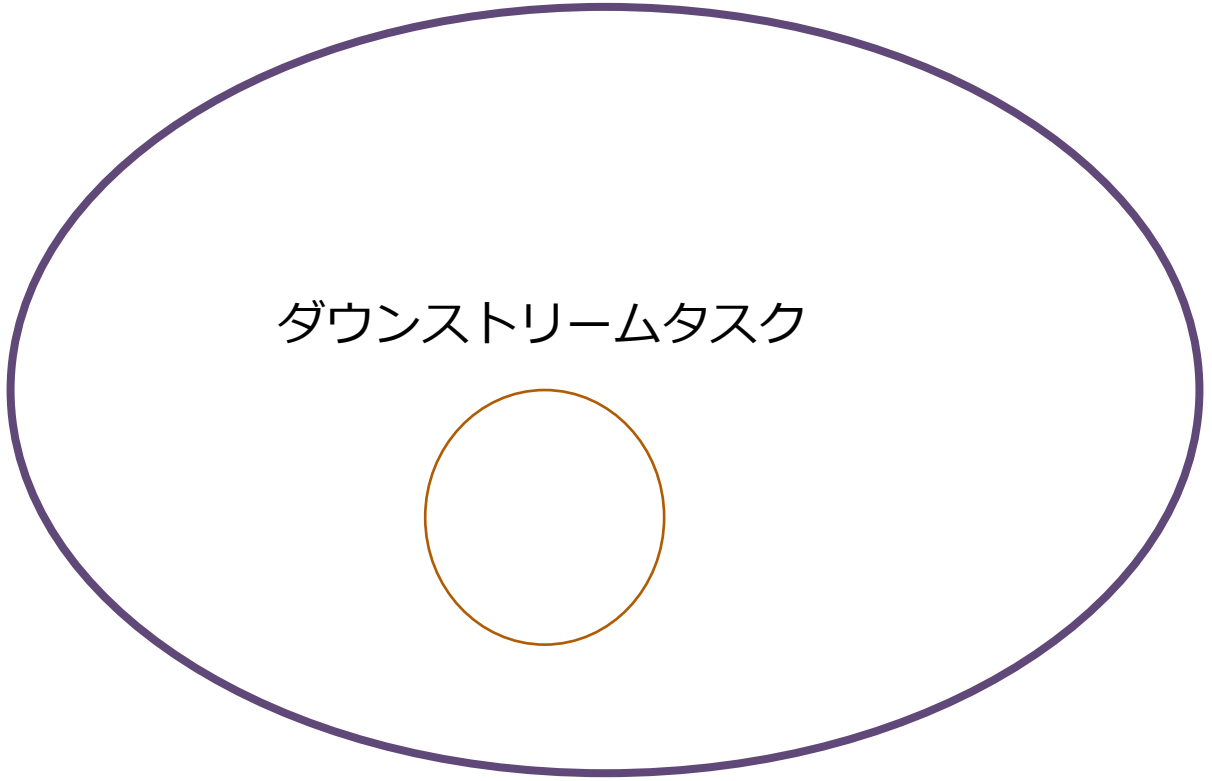
- **パラメータ効率**
  - 少ないParameter量でどれだけ精度があがるのか
- **学習効率**
  - 少ない学習コストでどれだけ精度があがるのか



データセット, 事前学習モデル依存である。

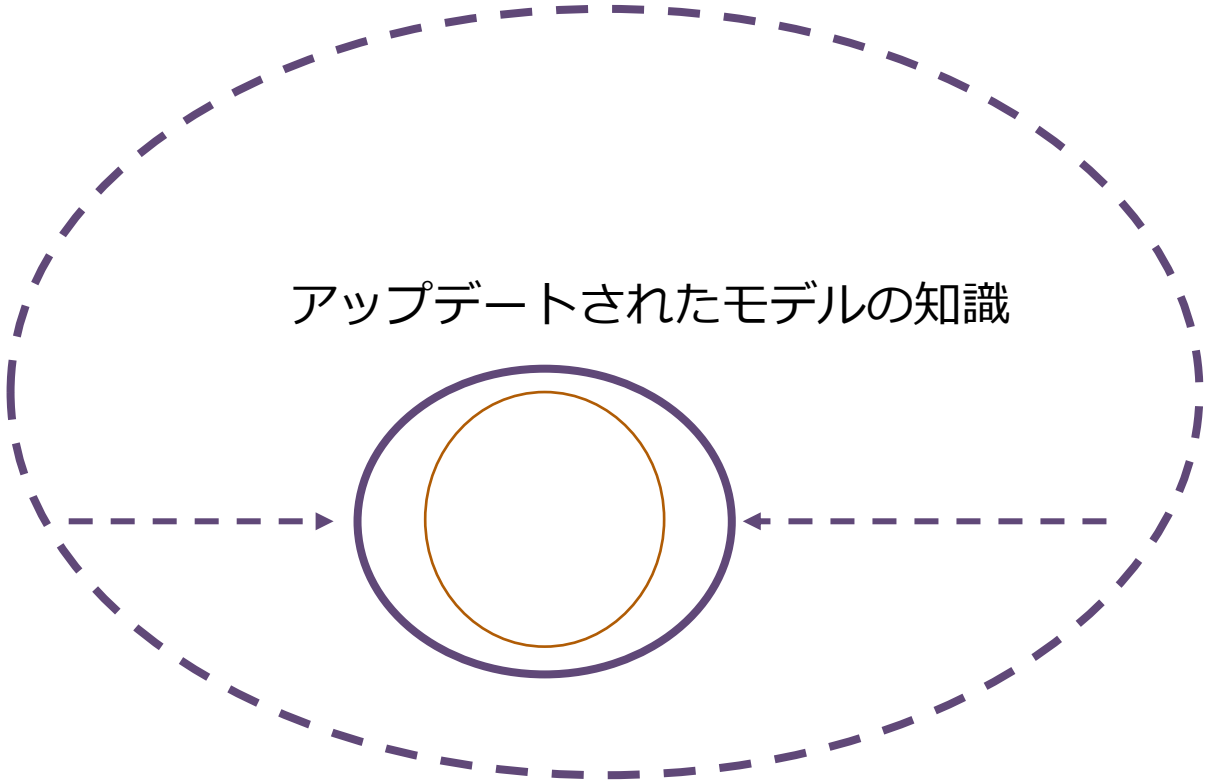
なぜなのか

事前学習データ（モデル）が
カバーしている知識



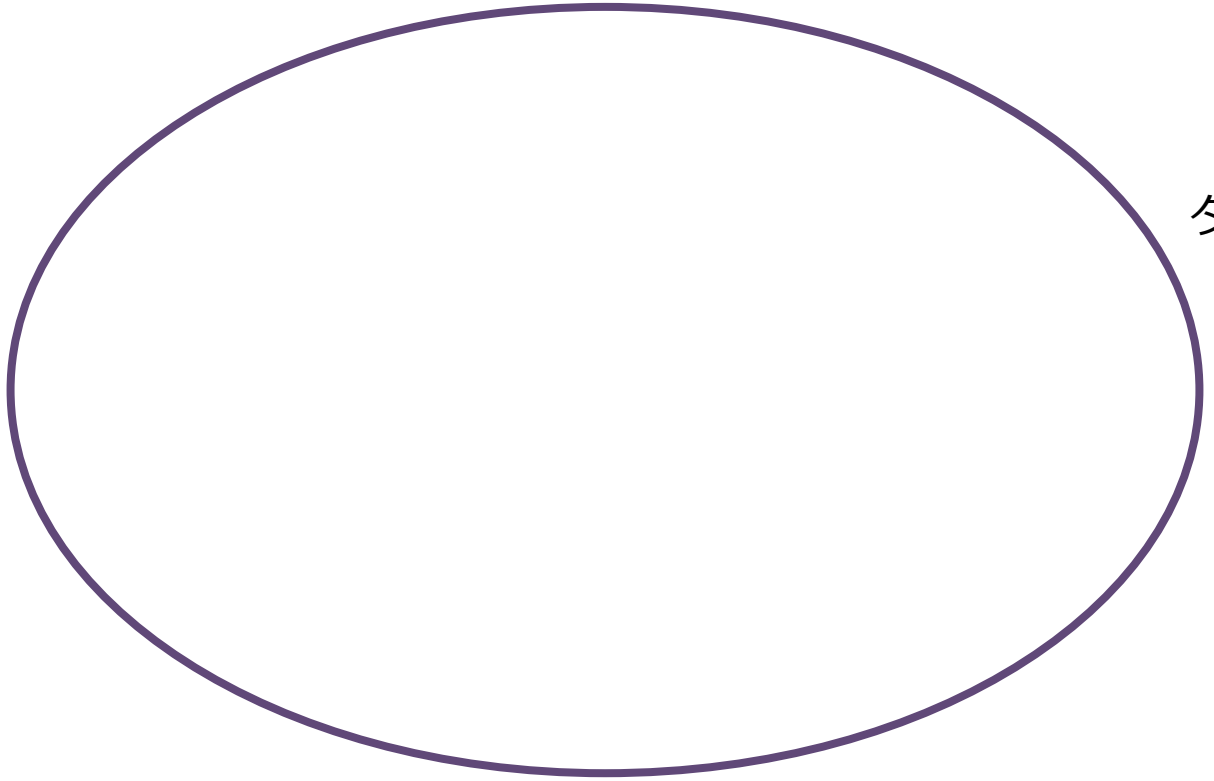
なぜなのか

事前学習データ（モデル）が
カバーしている知識

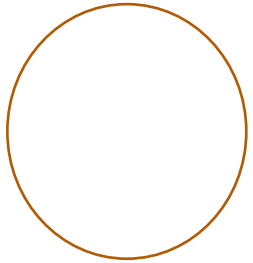


なぜなのか

事前学習データ（モデル）が
カバーしている知識



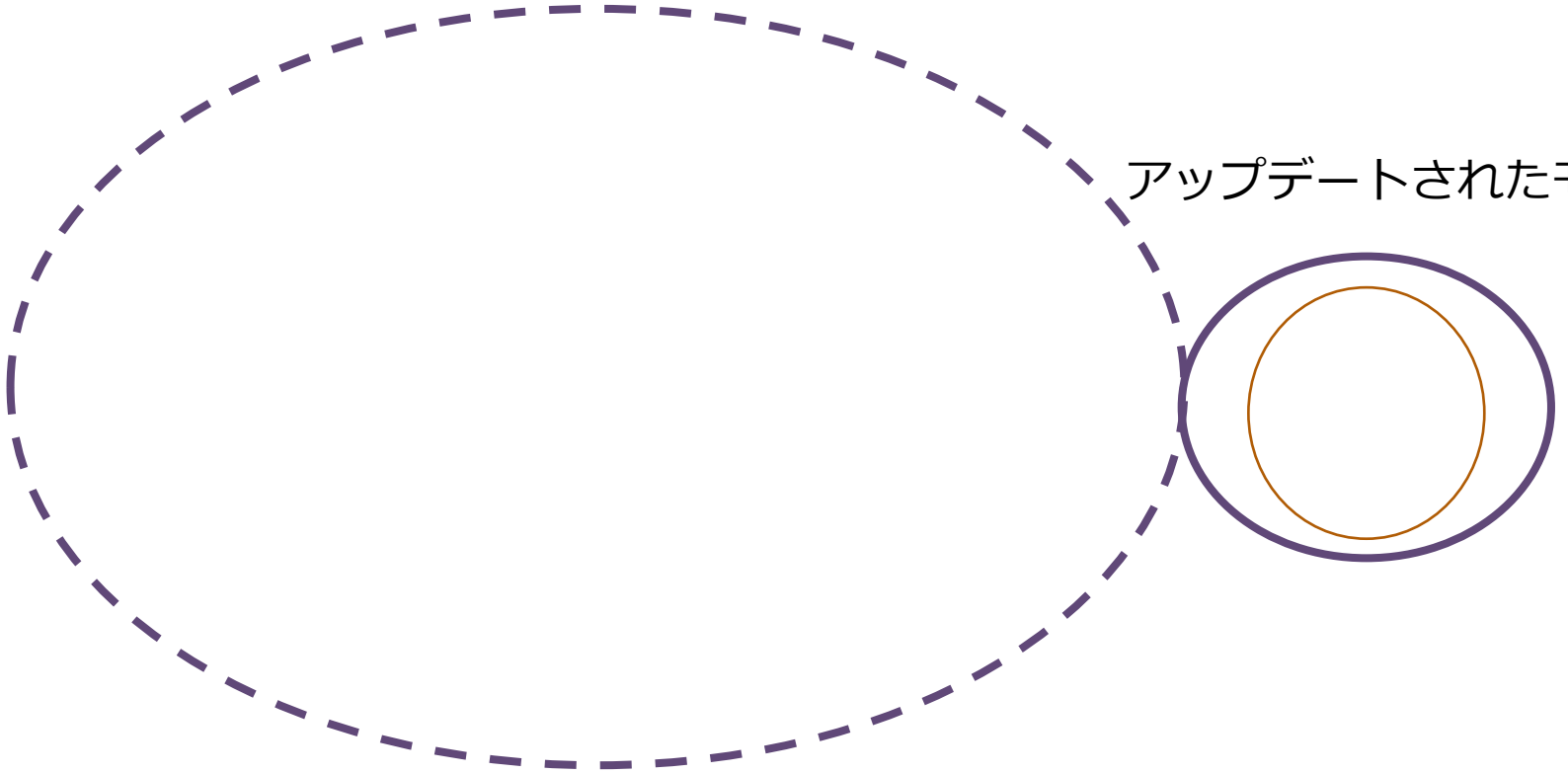
ダウンストリームタスク



なぜなのか

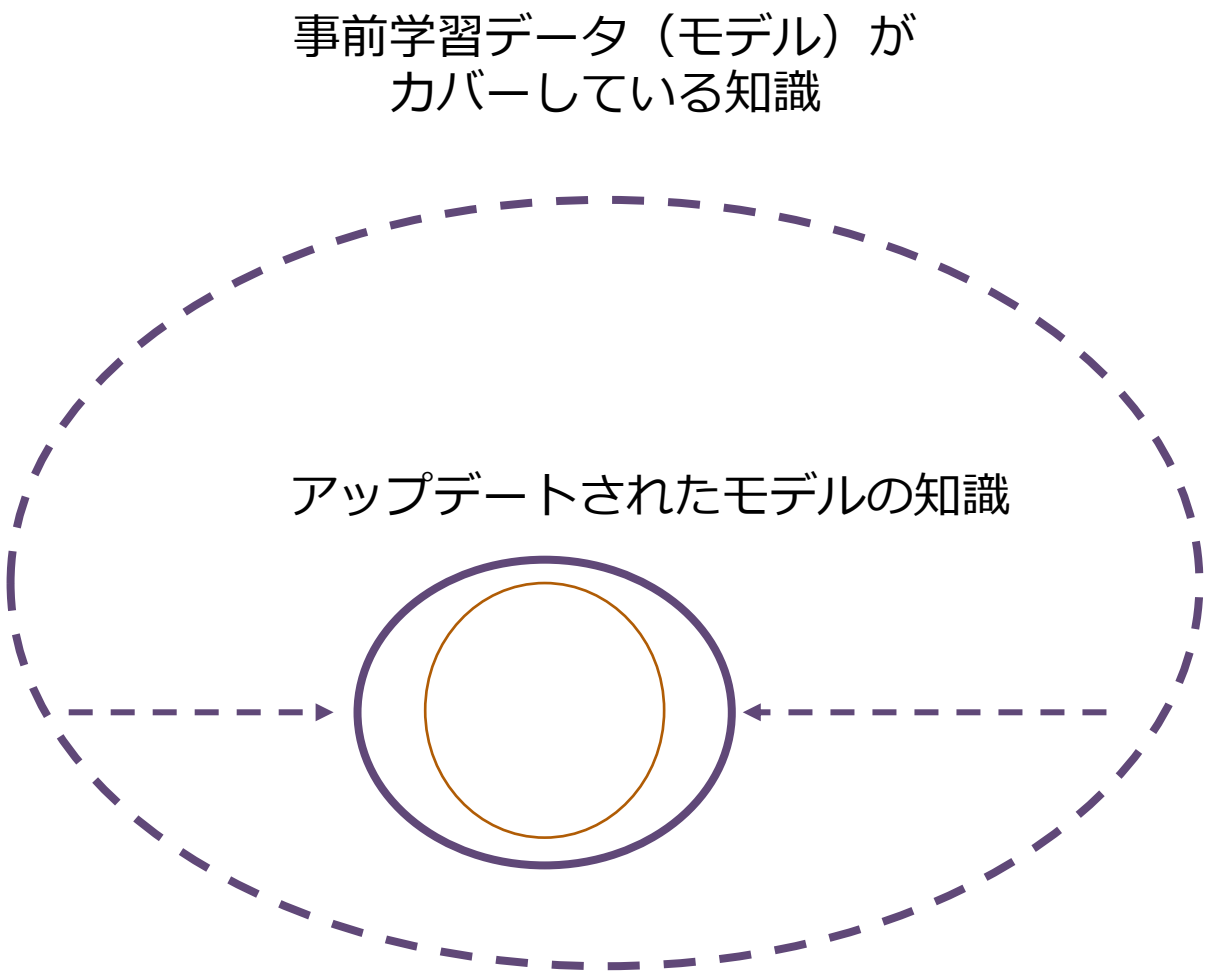
事前学習データ（モデル）が
カバーしている知識

アップデートされたモデルの知識



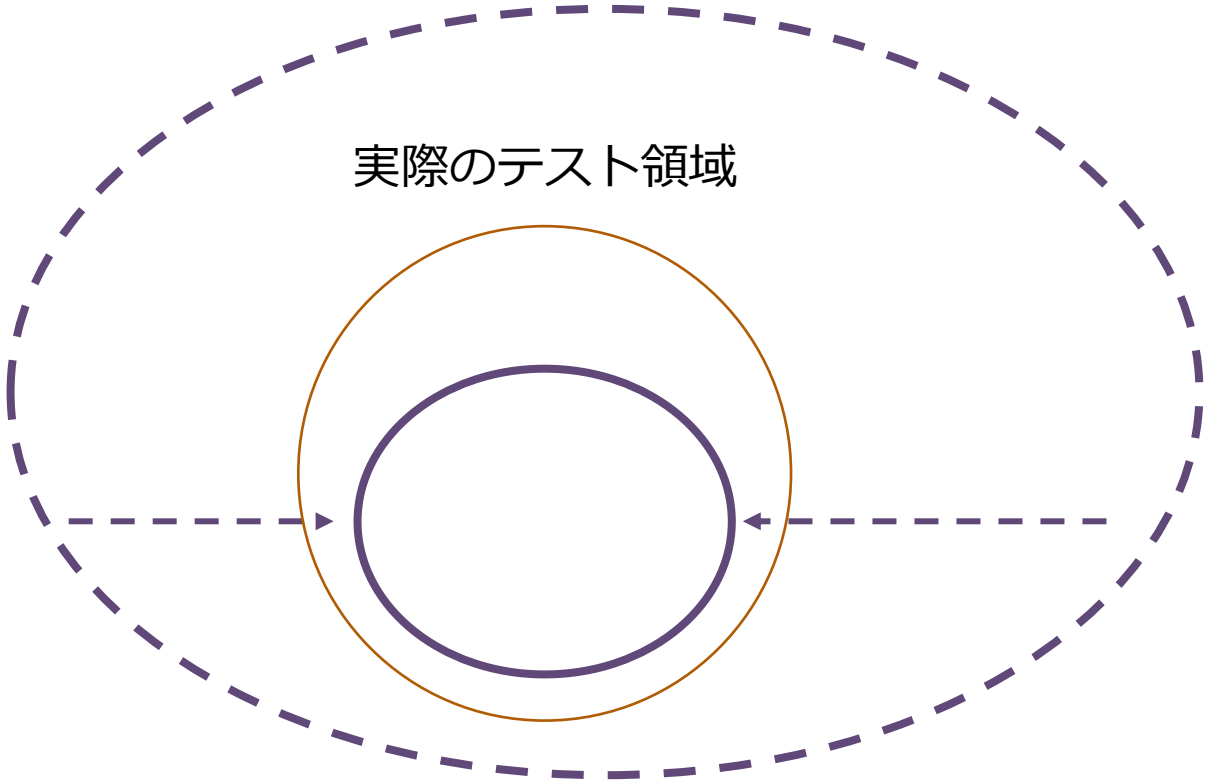
大きな知識のアップデートが求められる

なぜなのか



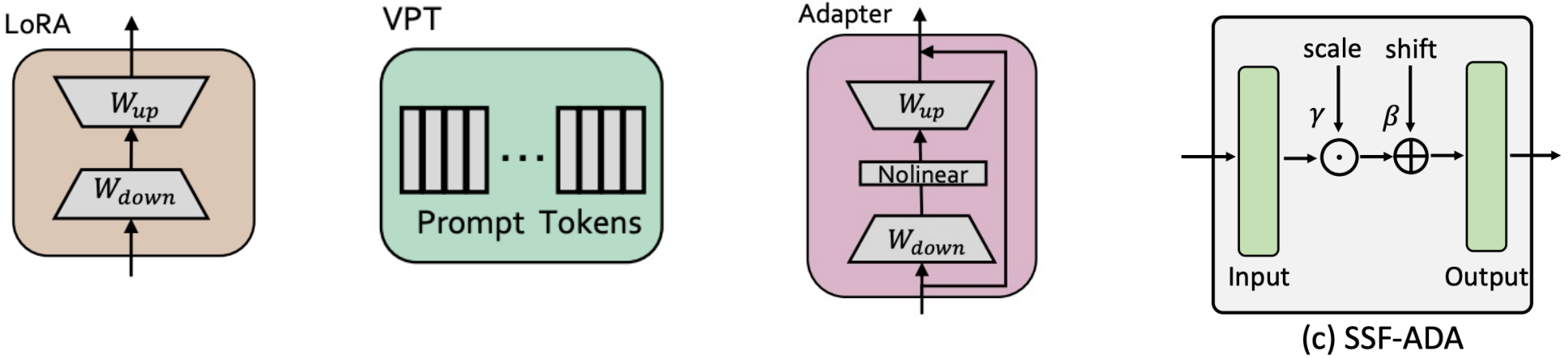
なぜなのか

事前学習データ（モデル）が
カバーしている知識



なぜなのか

- **知識アップデート能力は手法固有であるはず**
  - 能力大！ => アップデートしすぎる可能性がある (Over Fitting)
  - 能力小！ => アップデートしきれない (Under Fitting)



+ Neural Architecture Search

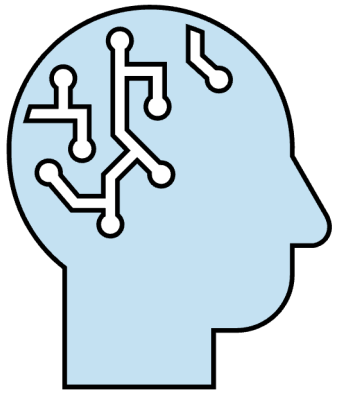
- どれがいいのかよく分からないので、組み合わせをSearchした

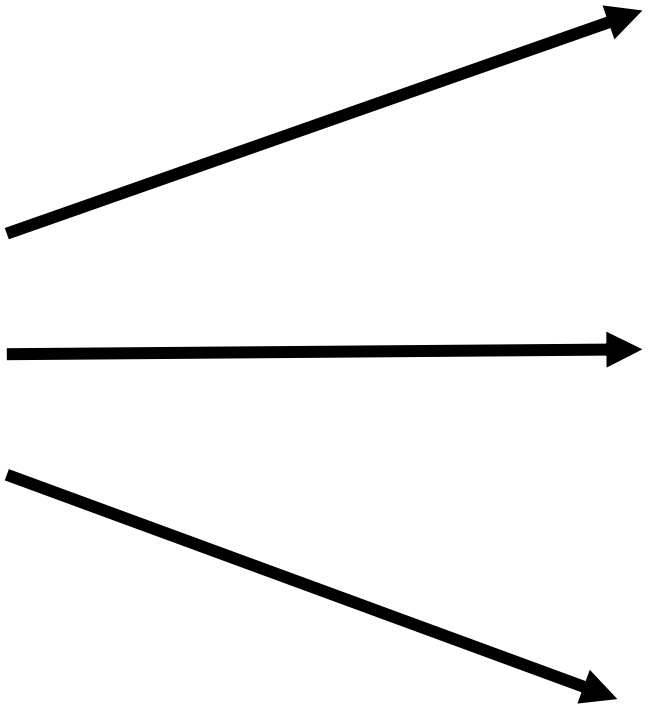
	# param (M)	Natural							Specialized				Structured								Average
		Cifar100	Caltech101	DTD	Flower102	Pets	SVHN	Sun397	Camelyon	EuroSAT	Resisc45	Retinopathy	Clevr-Count	Clevr-Dist	DMLab	KITTI-Dist	dSpr-Loc	dSpr-Ori	sNORB-Azim	sNORB-Ele	
Full [19]	85.8	**68.9**	**87.7**	**64.3**	87.2	**86.9**	**87.4**	38.8	**79.7**	**95.7**	**84.2**	73.9	**56.3**	**58.6**	**41.7**	**65.5**	**57.5**	**46.7**	**25.7**	**29.1**	**68.9**
Linear [19]	0.04	64.4	85.0	63.2	**97.0**	86.3	36.6	**51.0**	78.5	87.5	68.5	**74.0**	34.3	30.6	33.2	55.4	12.5	20.0	9.6	19.2	57.6
VPT [19]	0.64	**78.8**	90.8	65.8	98.0	88.3	78.1	49.6	81.8	**96.1**	83.4	68.4	68.5	60.0	46.5	72.8	73.6	47.9	**32.9**	37.8	72.0
Adapter [17]	0.16	69.2	90.1	68.0	98.8	89.9	82.8	**54.3**	84.0	94.9	81.9	75.5	80.9	65.3	48.6	78.3	74.8	**48.5**	29.9	41.6	73.9
LoRA [18]	0.29	67.1	91.4	69.4	98.8	**90.4**	85.3	54.0	**84.9**	95.3	**84.4**	73.6	**82.9**	**69.2**	49.8	78.5	75.7	47.1	31.0	44.0	74.5
NOAH	0.43	69.6	**92.7**	**70.2**	**99.1**	**90.4**	**86.1**	53.7	84.4	95.4	83.9	**75.8**	82.8	68.9	**49.9**	**81.7**	**81.8**	48.3	32.8	**44.2**	**75.5**

概ね良いが、単一手法に勝てないケースはある

Neural Prompt Search, Arxiv 2022

基盤
モデル

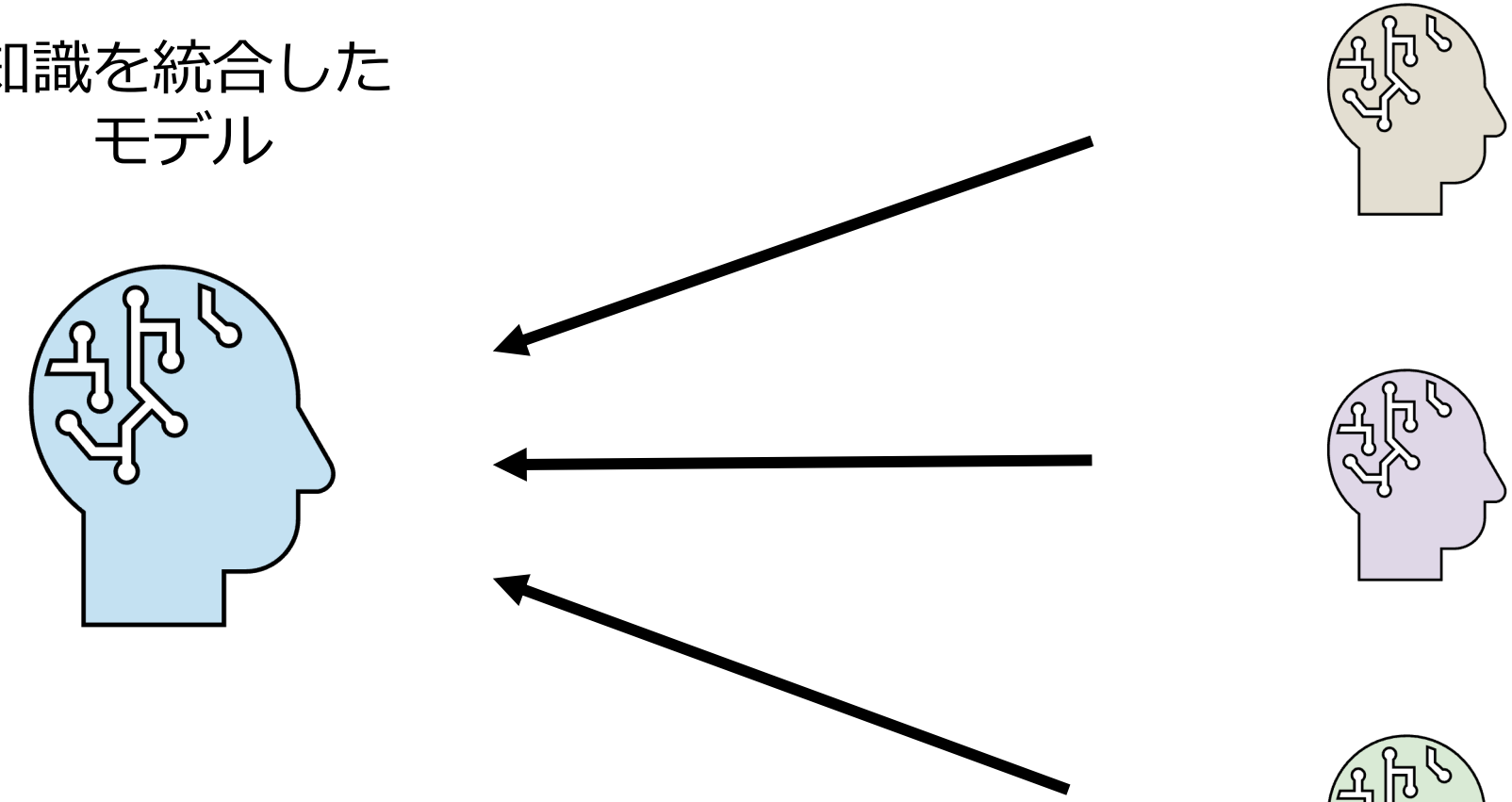




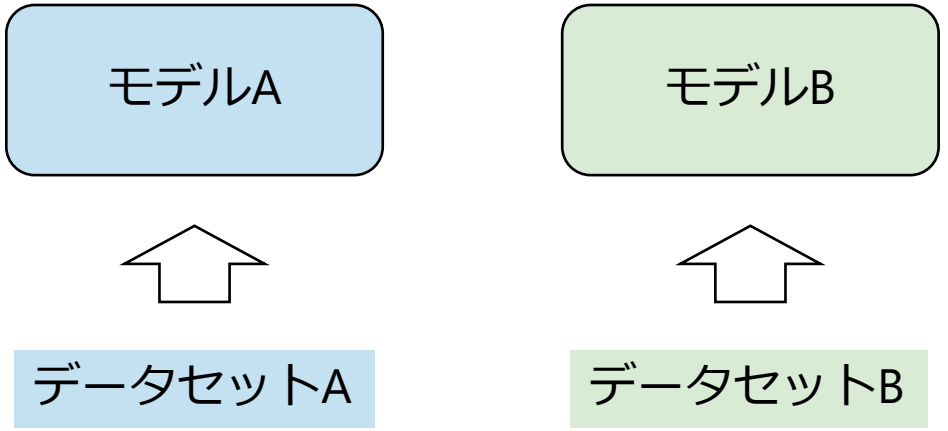
各データセットに
特化したモデル

各データセットに
特化したモデル

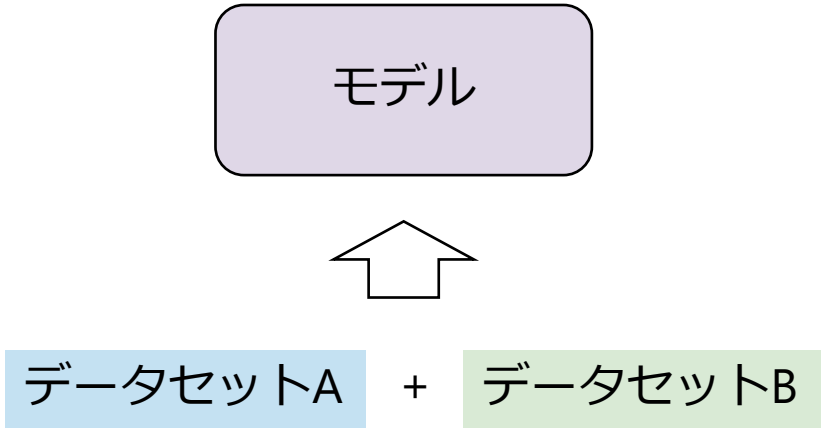
知識を統合した
モデル



データセットの知識を統合する学習



- モデル間でデータセットAとBの知識は共有されない



- AとBの知識を共有したモデルを作りたい
- テスト時にタスクの切り替えもしたい

データセットによる違い

Captionの詳細さにおける違い

りんご

3つの赤いりんごが
木目の板の上にある

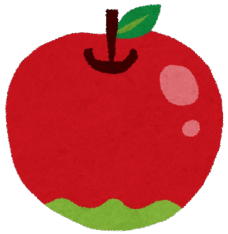
鳥

オレンジと白の鳥

猫

シャム猫が
こっちをみている

画像のドメインに関する違い



識別用データセットと説明文データセットの統合

説明文データセット

Two brown puppies on grass

A airplane flying over the shiny clouds

Beautiful purple flower is centered

✓多様な画像のドメイン
✓多様な語彙

＋

識別データセット

✓Fine-grainedなクラス
✓ラベル分布が均一

言語
エンコーダー

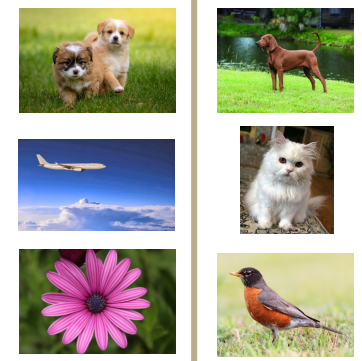
画像
エンコーダー

Prefix Conditioning Unifies Language and Label Supervision, CVPR2023

損失 (Contrastive Loss)

言語
エンコーダー

画像
エンコーダー



Two brown puppies are playing on the grass.

A airplane flying over the shiny clouds .

Beautiful purple flower centered on the image.

説明文データセット

A photo of a **redbone**

A close-up of a **persian cat**

A picture of **american robin**

Classification (識別） データセット

損失 (Contrastive Loss)

言語
エンコーダー

画像
エンコーダー

Two	brown	…	grass
a	airplane	…	clouds
beautiful	purple	…	image

Tokenize

Two brown puppies are playing on the grass.

A airplane flying over the shiny clouds .

Beautiful purple flower centered on the image.

説明文データセット

a	photo	…	redbone
a	close-up	…	cat
a	picture	…	robin

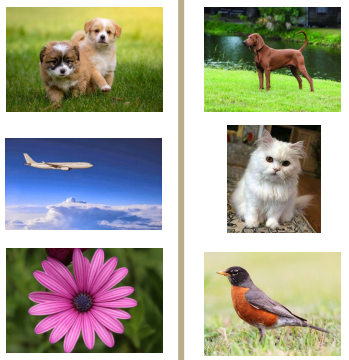
Tokenize

A photo of a **redbone**

A close-up of a **persian cat**

A picture of **american robin**

Classification (識別） データセット



損失 (Contrastive Loss)

言語
エンコーダー

画像
エンコーダー

Prompt

<説明文>	Two	brown	…	grass
<説明文>	a	airplane	…	clouds
<説明文>	beautiful	purple	…	image

Tokenize

Two brown puppies are playing on the grass.

A airplane flying over the shiny clouds .

Beautiful purple flower centered on the image.

説明文データセット

Prompt

<識別>	a	photo	…	redbone
<識別>	a	close-up	…	cat
<識別>	a	picture	…	robin

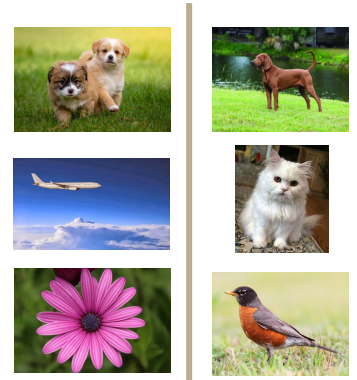
Tokenize

A photo of a **redbone**

A close-up of a **persian cat**

A picture of **american robin**

Classification (識別） データセット



損失 (Contrastive Loss)

・効率よく知識統合できる!
・推論時にスイッチングできる！

言語
エンコーダー

画像
エンコーダー

Prompt

<説明文>	Two	brown	…	grass
<説明文>	a	airplane	…	clouds
<説明文>	beautiful	purple	…	image

Tokenize

Two brown puppies are playing on the grass.

A airplane flying over the shiny clouds .

Beautiful purple flower centered on the image.

説明文データセット

Prompt

<識別>	a	photo	…	redbone
<識別>	a	close-up	…	cat
<識別>	a	picture	…	robin

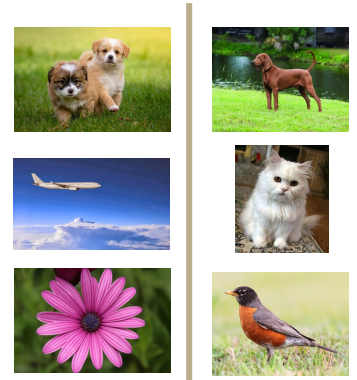
Tokenize

A photo of a **redbone**

A close-up of a **persian cat**

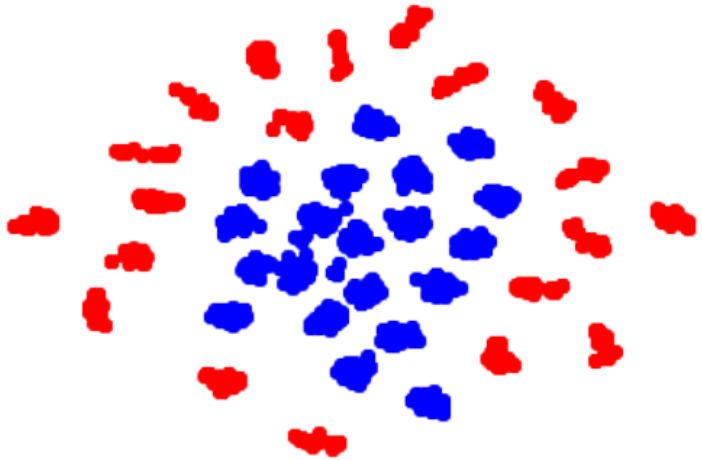
A picture of **american robin**

Classification (識別） データセット

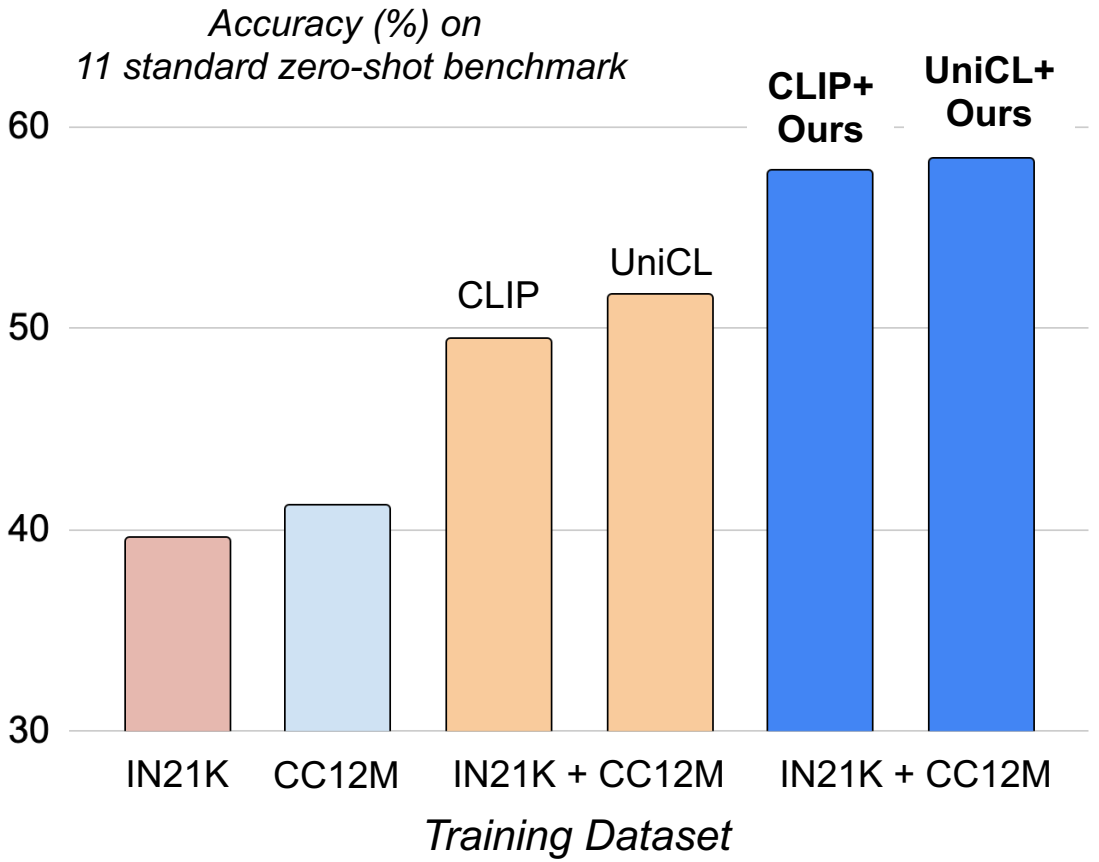


知識統合のメリット+タスクの切り替えを観測

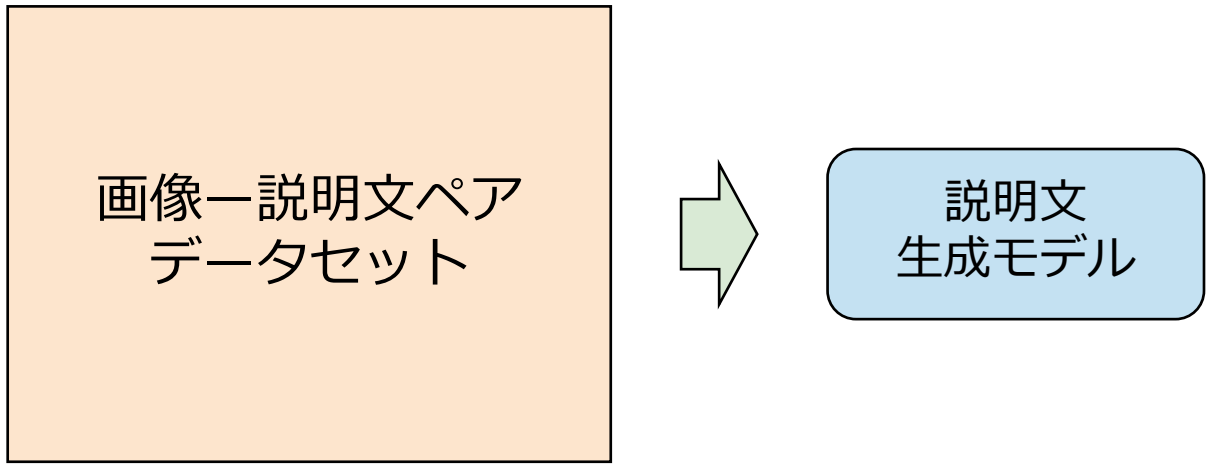
言語Embeddingの分布
(プロンプトのみをスイッチ)



赤：Classification データセットプロンプト
青：Caption データセットプロンプト



説明文の質を考慮する学習

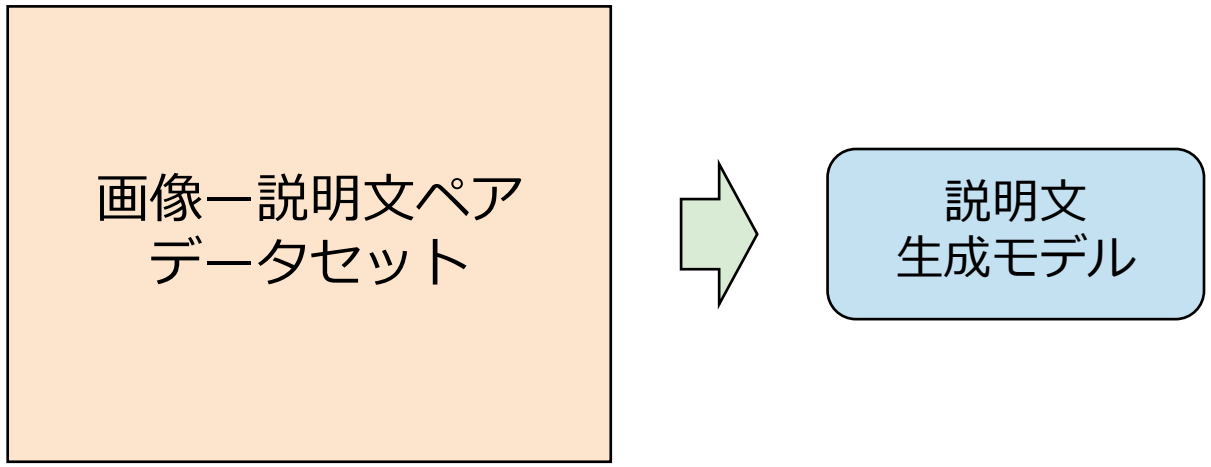


Noise-aware Learning from Web-crawled Image-Text Data for Image Captioning, ICCV 2023

説明文の質を考慮する学習

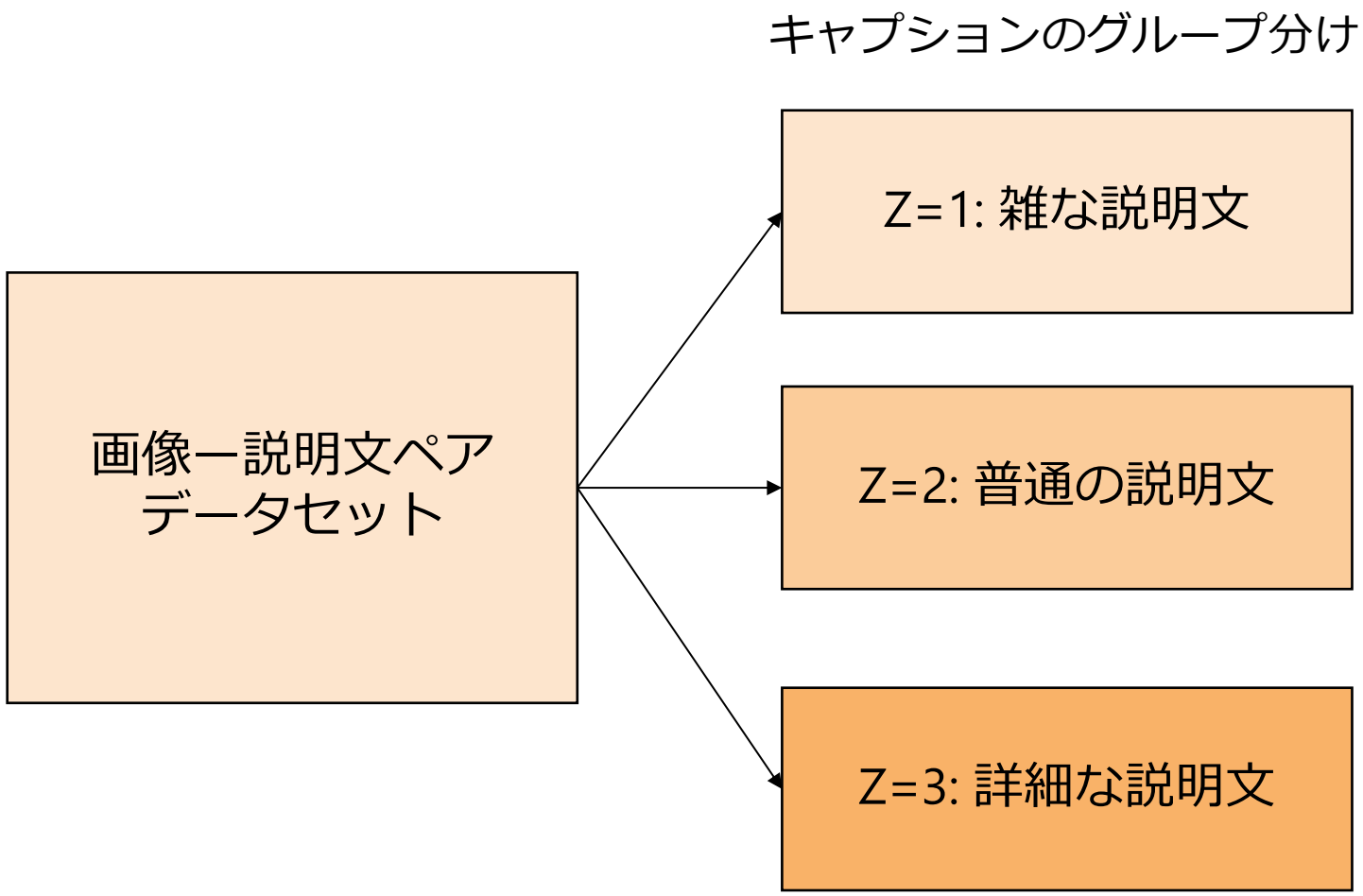
問題点

詳細、普通, 雑な説明文、全てがデータセットに含まれる。
=> 生成される文の質をうまくコントロールできない。

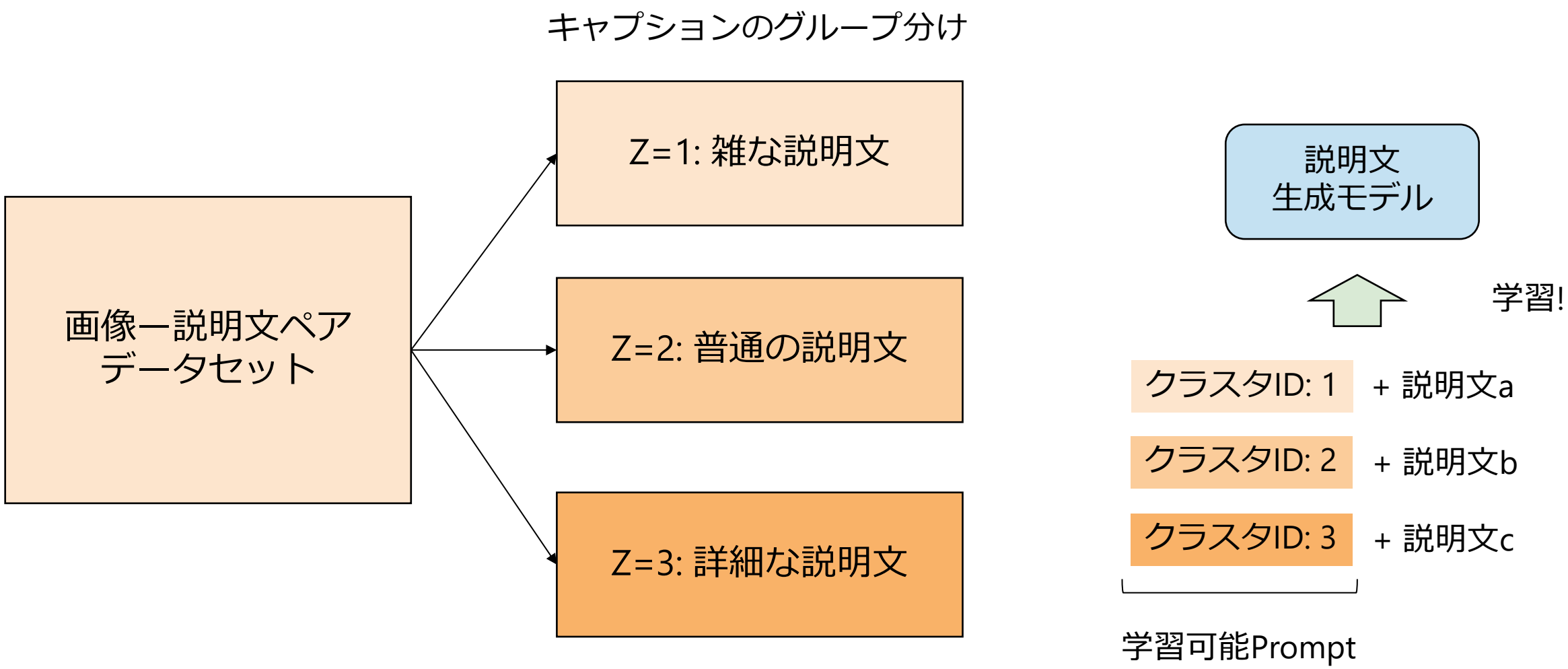


Noise-aware Learning from Web-crawled Image-Text Data for Image Captioning, ICCV 2023

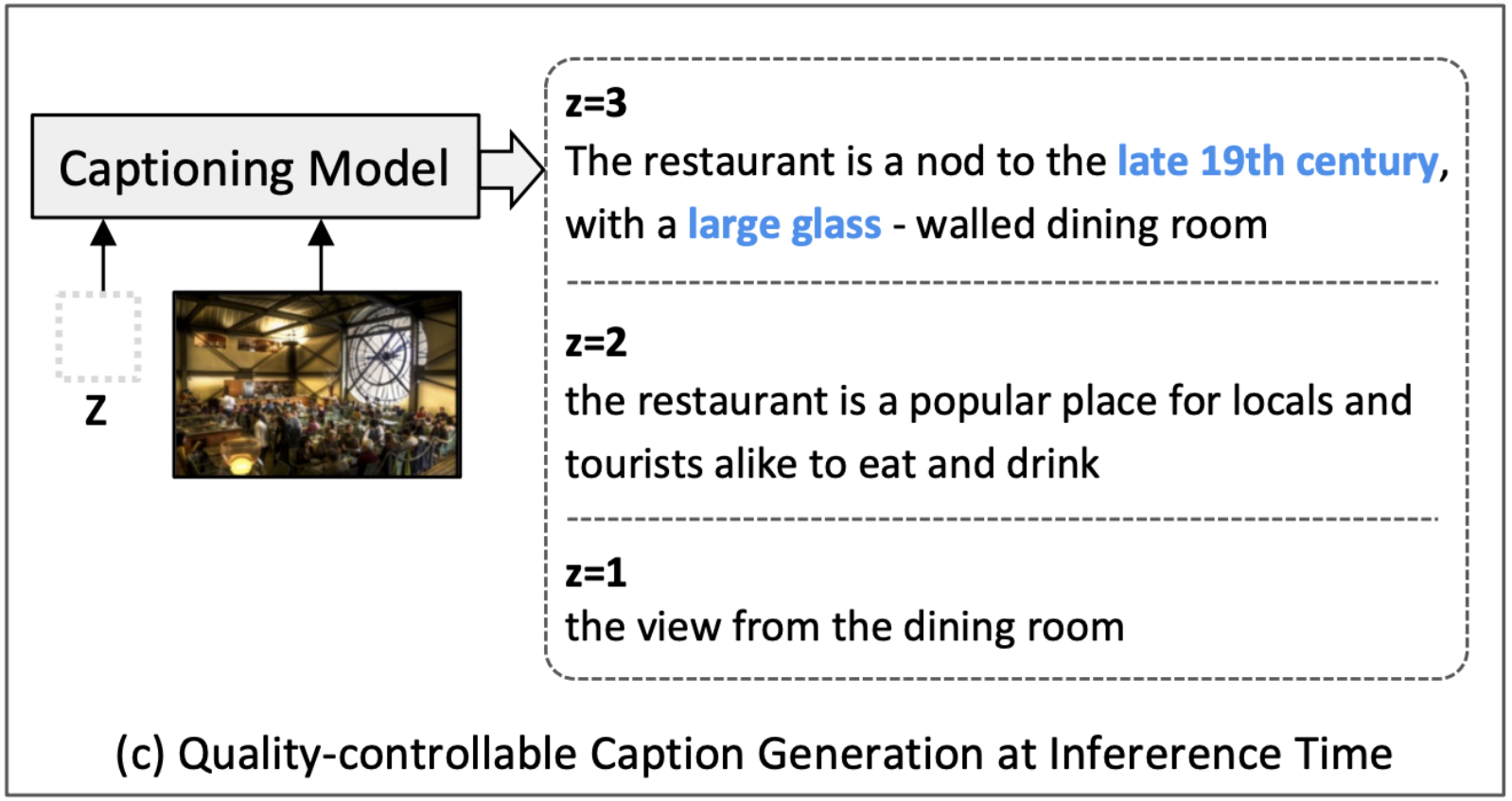
説明文の質を考慮する学習



説明文の質を考慮する学習



説明文の質を考慮する学習



Noise-aware Learning from Web-crawled Image-Text Data for Image Captioning, ICCV 2023

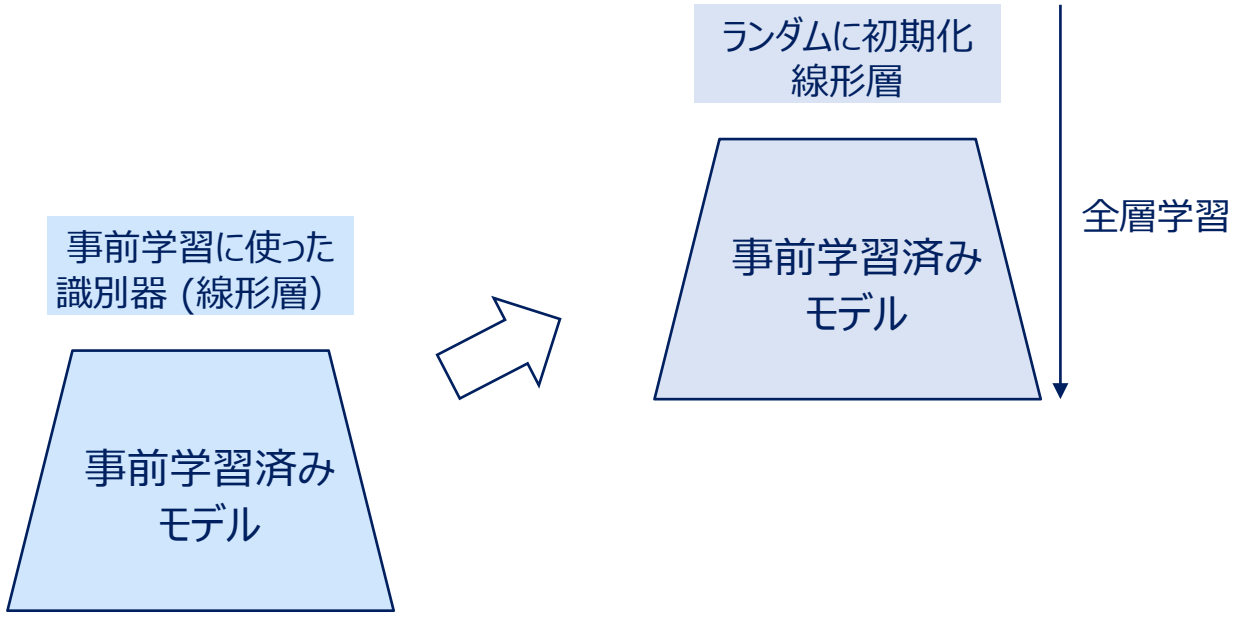
Parameter Efficientなチューニングのまとめ

- **Parameter Efficientなチューニング**
  - 精度向上はデータセット依存な部分が大きい
  - 組み合わせのSearch等
- **タスク切り替えとしてのParameter Efficientなモデル**
  - データセットの違いを考慮しつつ、知識を統合する学習

基盤モデルのチューニング : ロバストな適合

ファインチューニングはドメイン汎化能力を落とす？

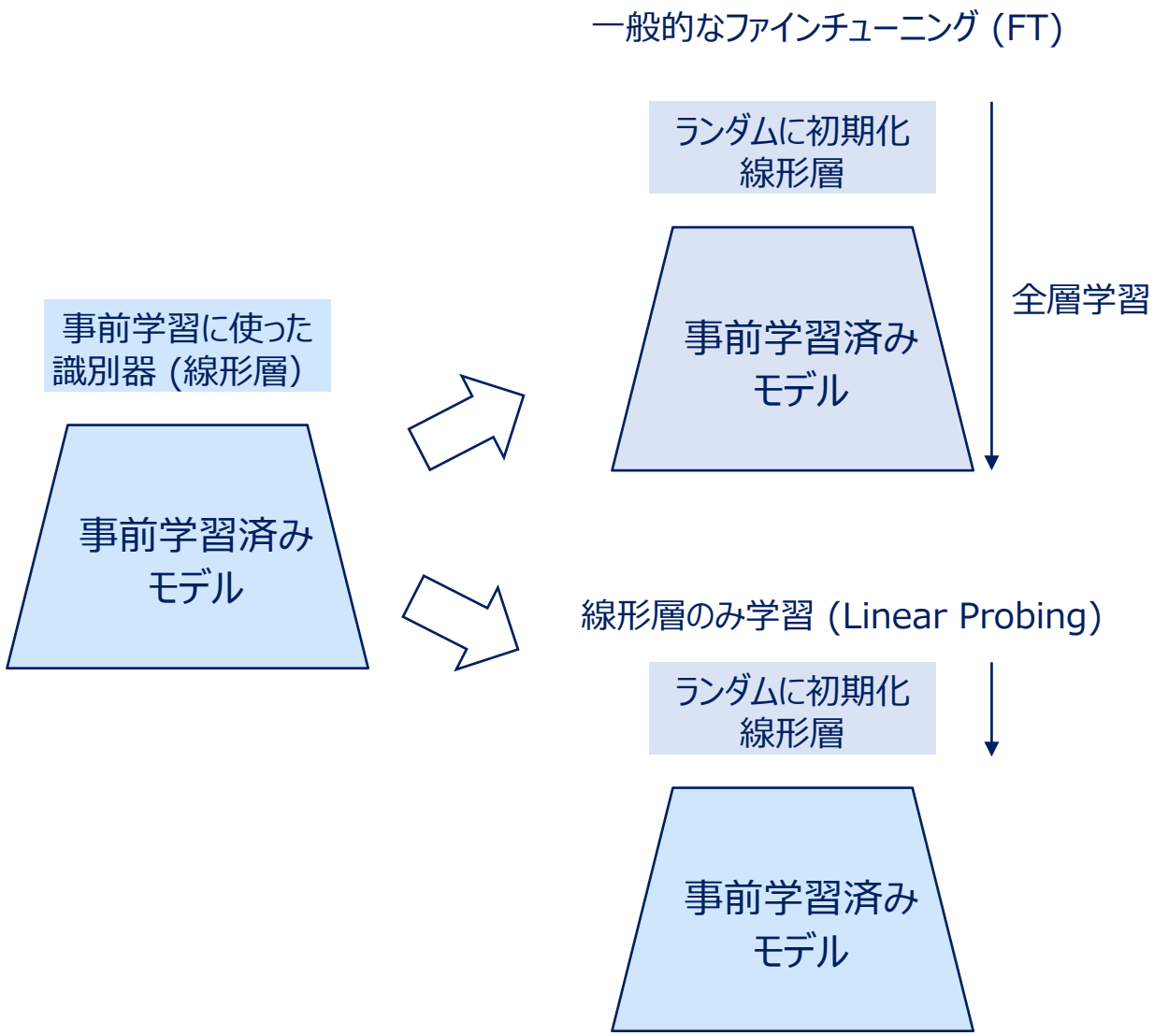
一般的なファインチューニング (FT)



- 学習データに対して、しっかりFitする。
- 事前学習済みモデルは更新されてしまう。

Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution, ICLR2022

ファインチューニングはドメイン汎化能力を落とす？

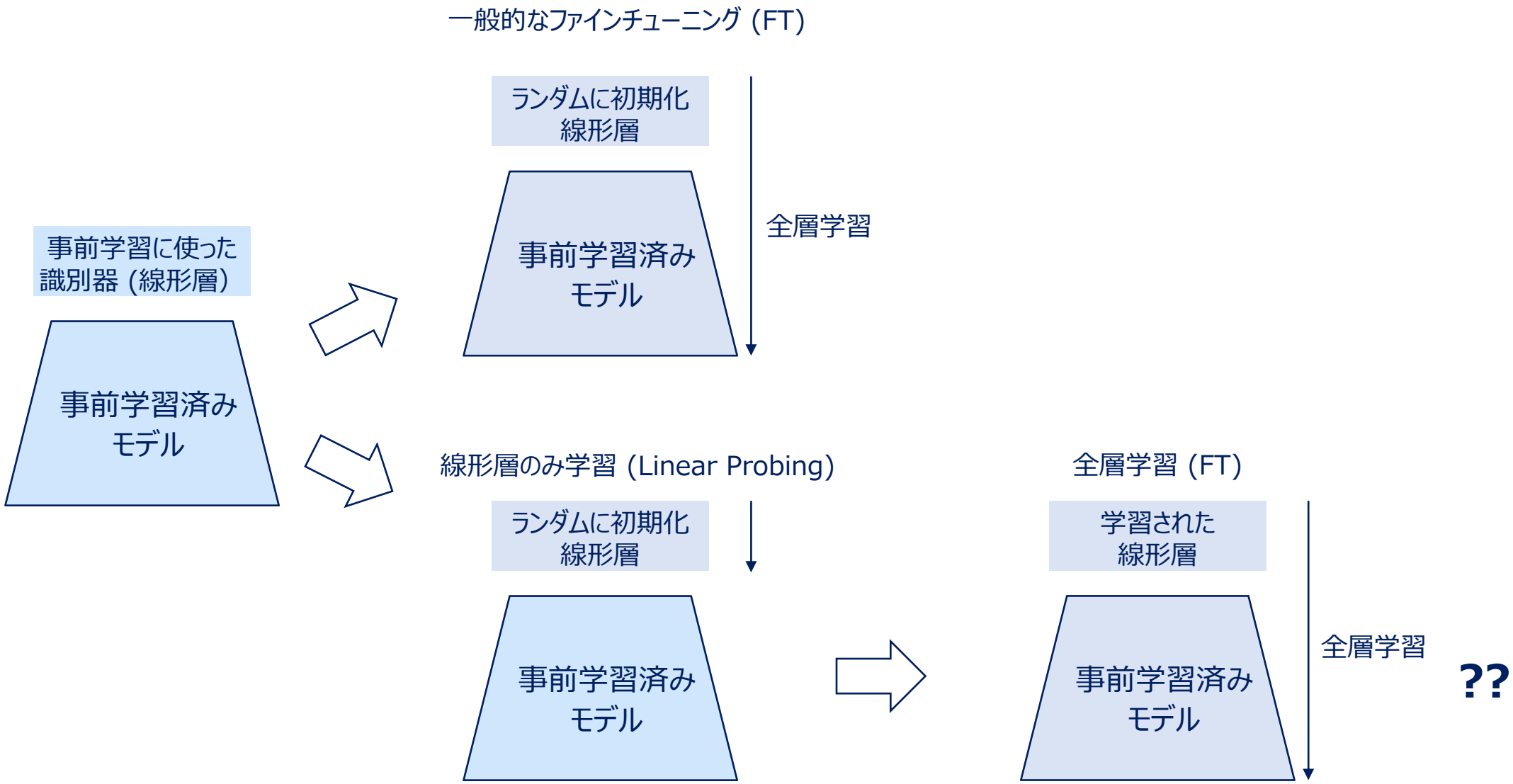


- 学習データに対して、しっかりFitする。
- 事前学習済みモデルは更新されてしまう。

- 線形層のみでは、しっかりFitできないかも。
- 事前学習済みモデルは完全に保持できる。

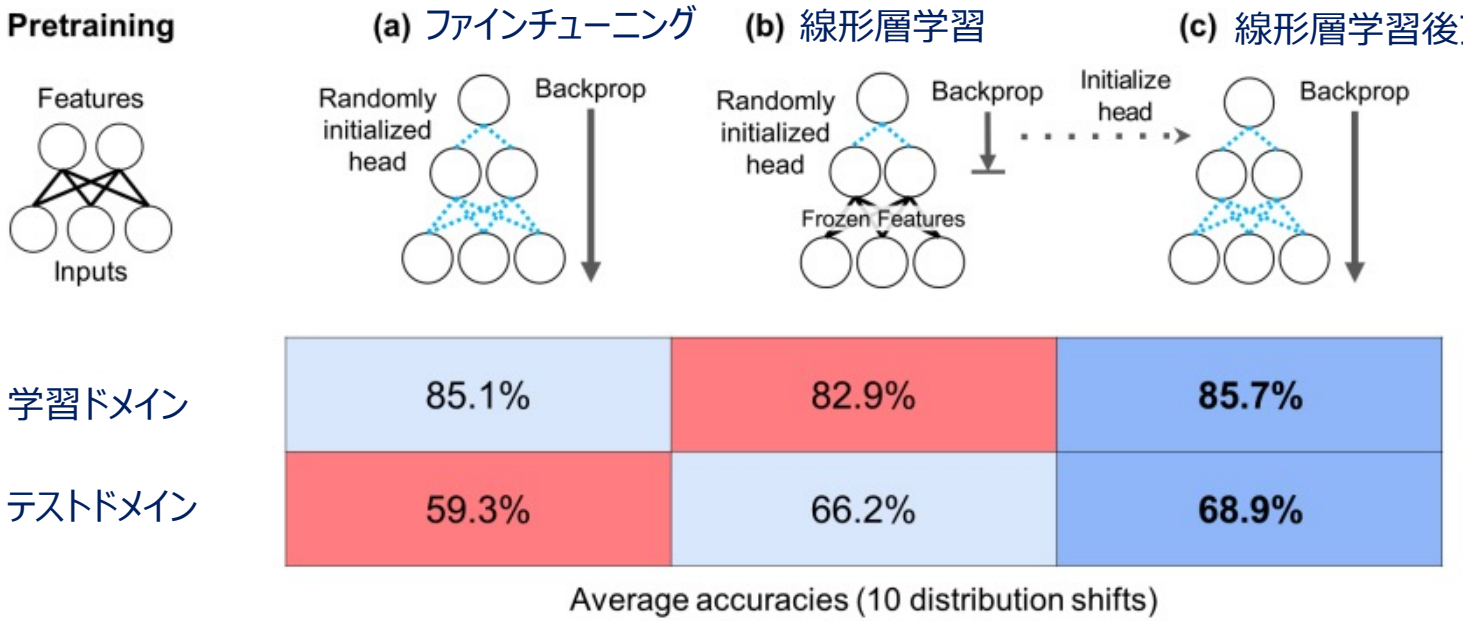
Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution, ICLR2022

ファインチューニングはドメイン汎化能力を落とす？



Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution, ICLR2022

ファインチューニングはドメイン汎化能力を落とす？

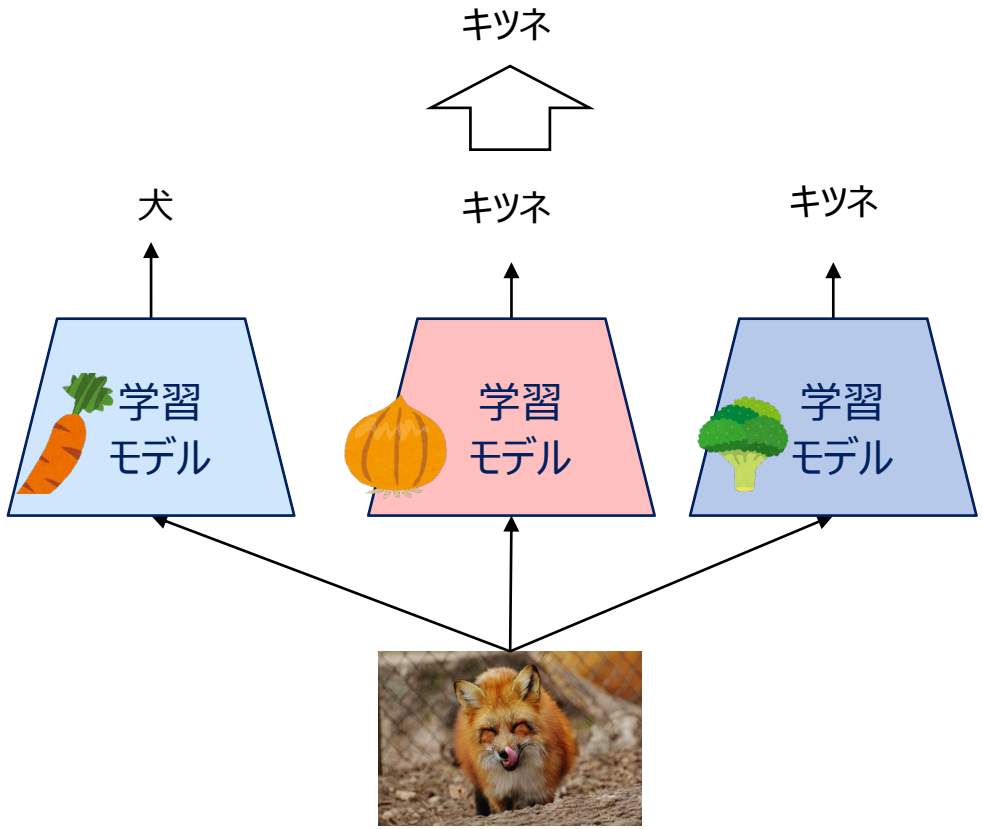


何が起こっているのかを大雑把にいうと、

線形層をチューニングせずに学習すると、
線形層をチューニングしていれば、起こらなかったアップデートが事前学習部分で起こる。
事前学習で持っていた性能を落としてしまう。

Model Soup: モデルの重みアンサンブル

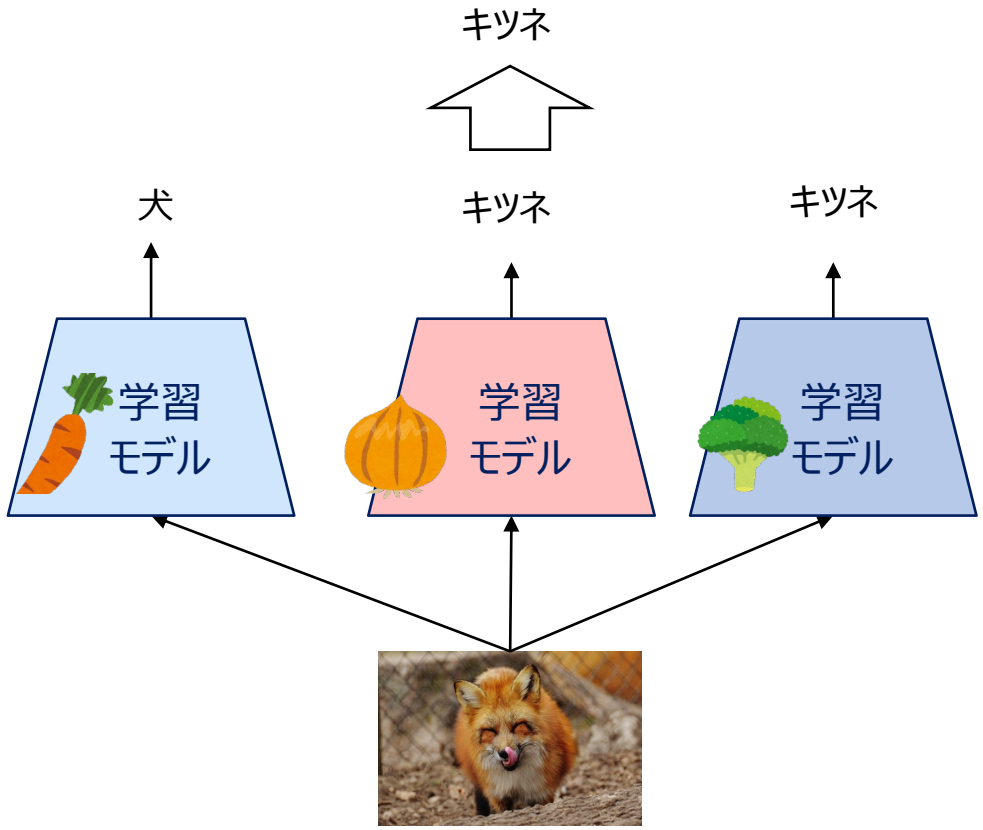
一般的なアンサンブル: 予測結果を統合する



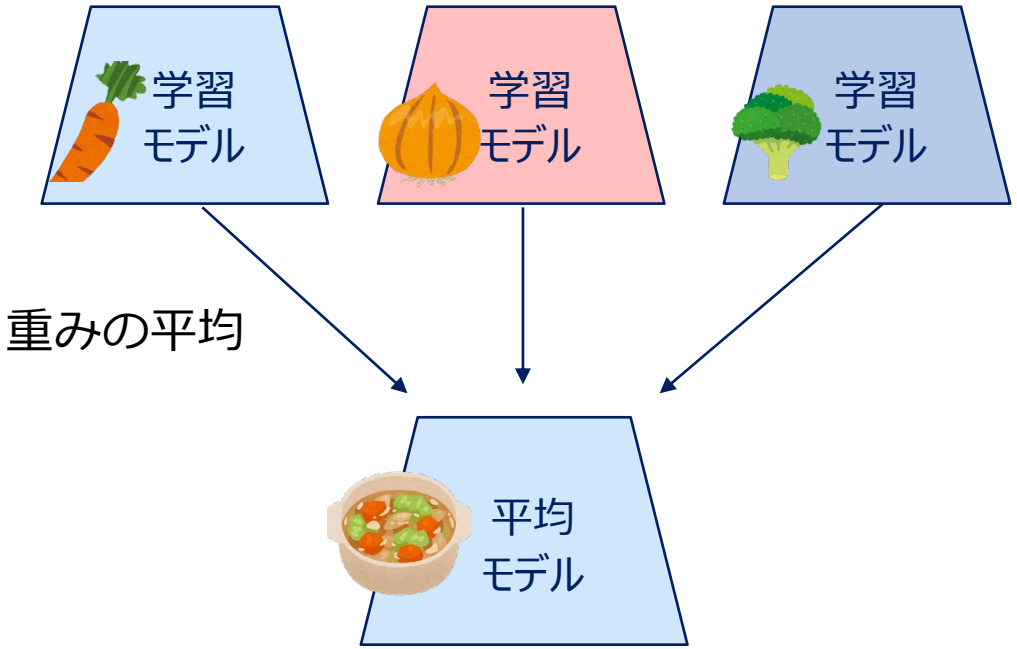
Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, ICML2022

Model Soup: モデルの重みアンサンブル

一般的なアンサンブル: 予測結果を統合する



Model Soup: 重み空間でアンサンブルをとる

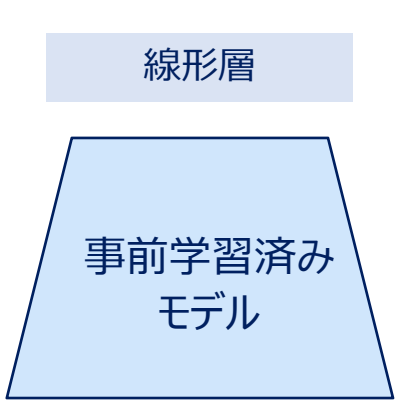


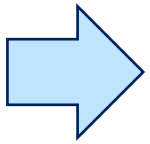
しかし，普通に平均化するのでは、ダメそう..

Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, ICML2022

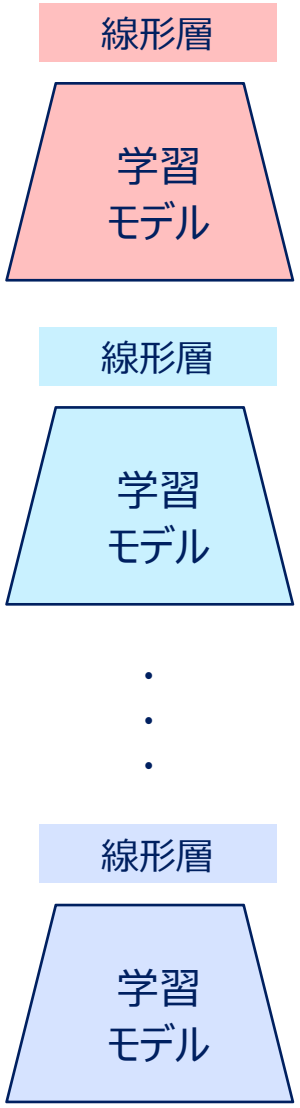
モデル初期化

線形層のみ学習 (Linear Probing)





各モデルを独立に学習



モデル初期化

線形層のみ学習 (Linear Probing)

線形層

事前学習済み
モデル

各モデルを独立に学習

線形層

学習
モデル

線形層

学習
モデル

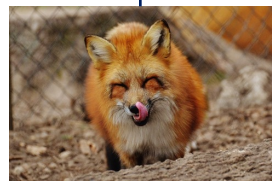
線形層

学習
モデル

重み付け平均

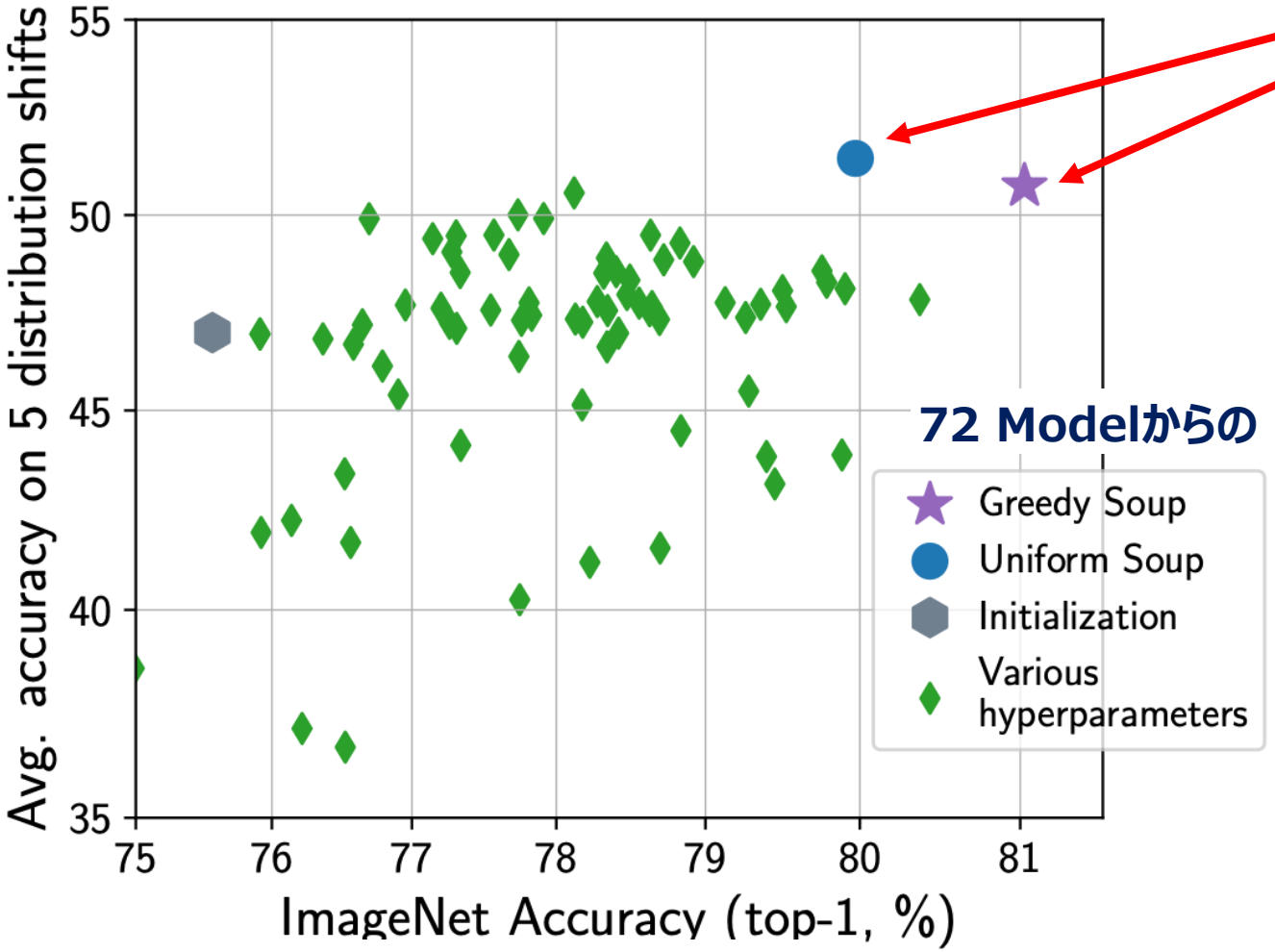
線形層

平均化
モデル



様々なドメインで汎化する!

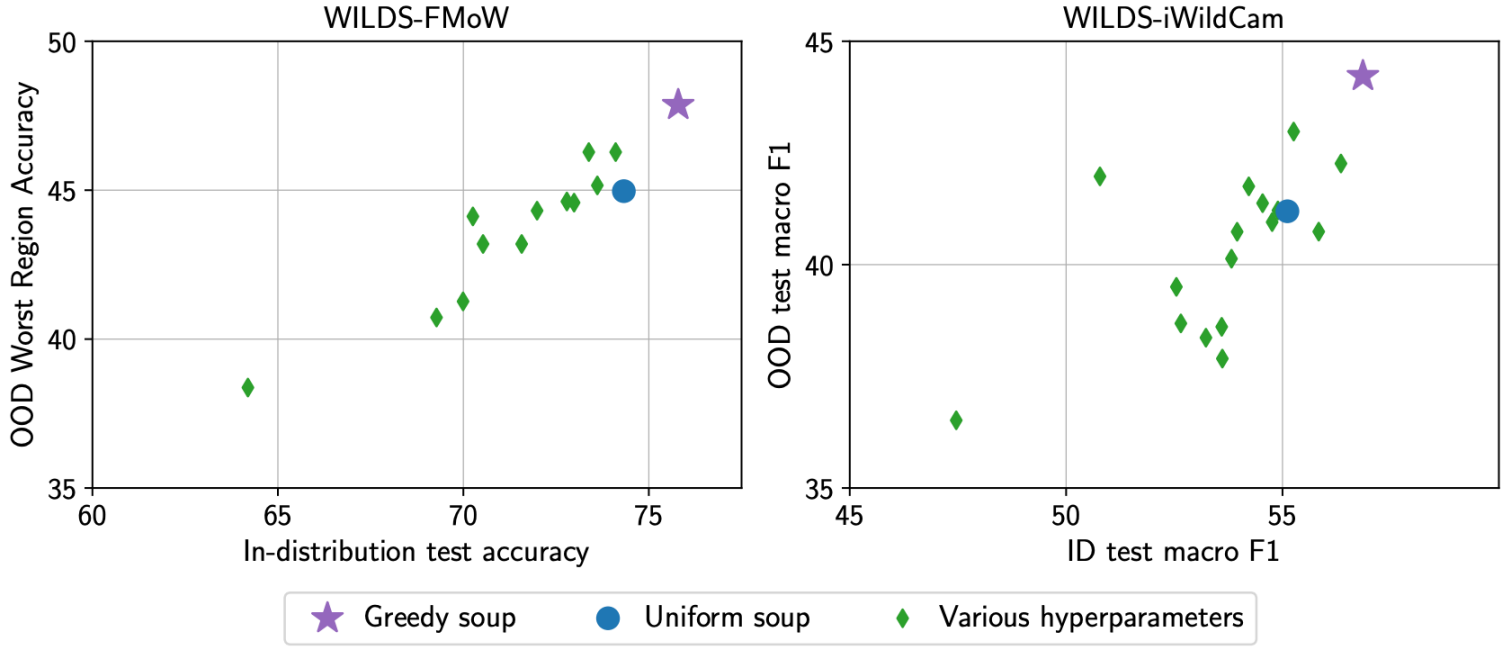
学習ドメインにおける精度（横軸）vs 異なるドメインにおける精度（縦軸）



普通に学習した場合 ♦ と比べて精度が高い

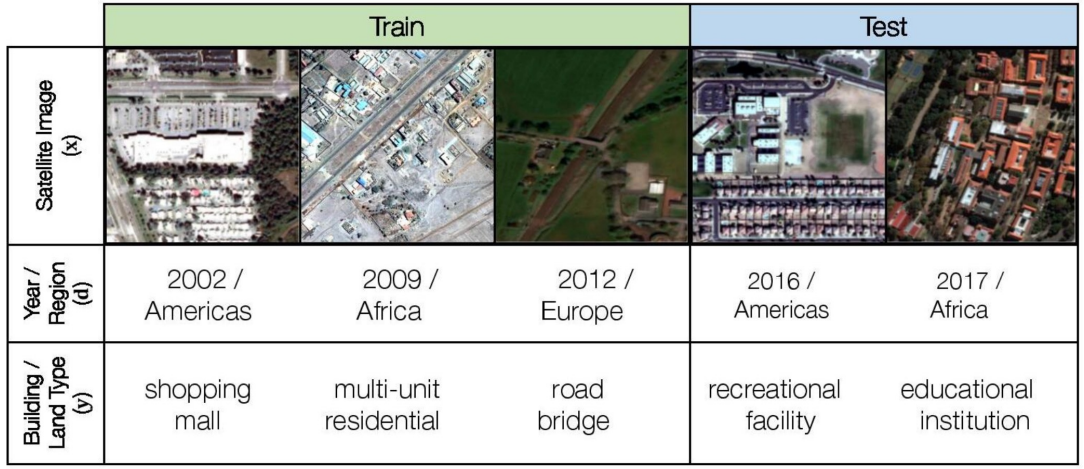
Uniform: 全モデル足し合わせる
Greedy: 足し合わせて精度が上がるかどうかで、取捨選択

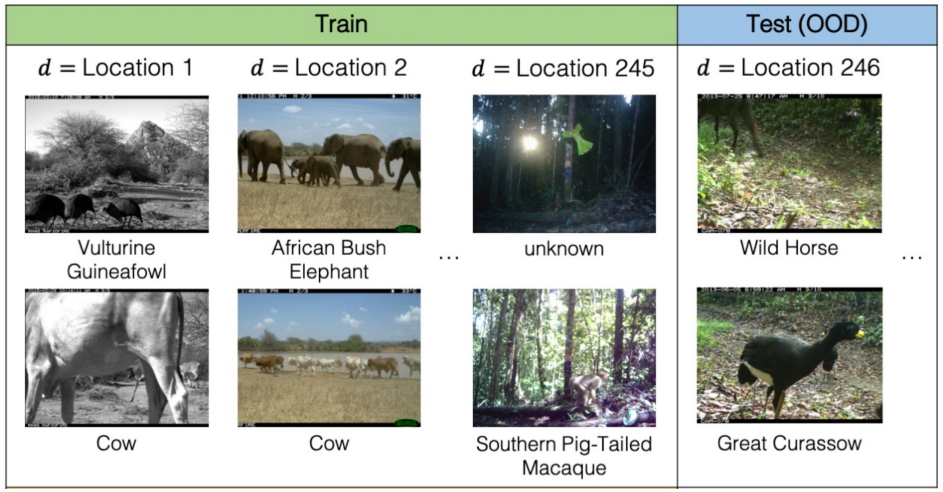
様々なドメインで汎化する!



WILDS

- 実世界のドメインギャップを評価するために提案された
- 非常にChallengingなデータセット





https://wilds.stanford.edu/datasets/

事前学習済みモデルで正則化するFine-tuning手法

・事前学習済みのモデルFine-tuningを**物体検出**で評価。
・事前学習済みモデルの重みで正則化、精度向上

正則化項

$$\Omega(w) = \sum_i \| \underbrace{w_i}_{\text{学習モデル}} - \underbrace{w_i^{pre}}_{\text{事前学習済みモデル}} \|^2.$$

事前学習済みモデルがロバストなのは、既知。
パラメータを学習する際に大きくそこから離れないようにしたい!

Mind the Backbone: Minimizing Backbone Distortion for Robust Object Detection, Arxiv 2022

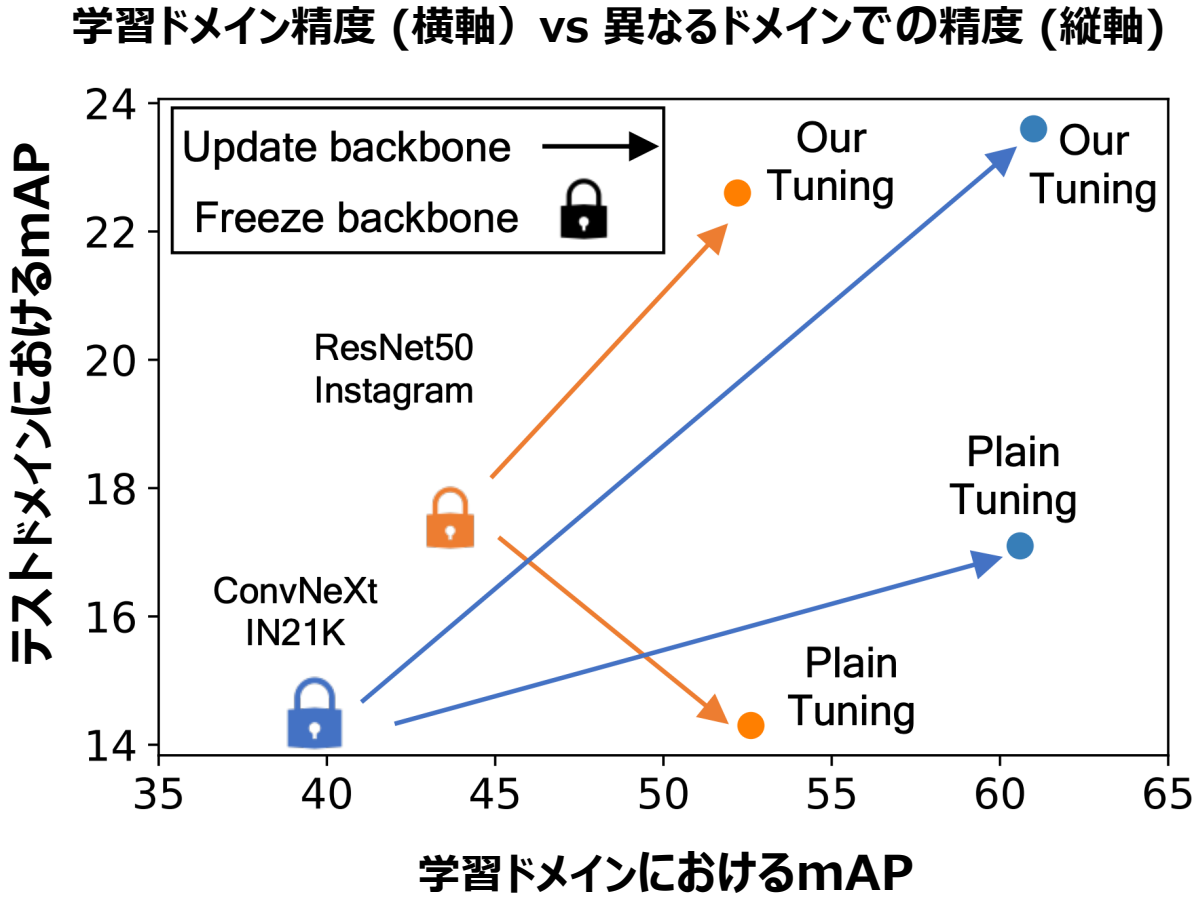
事前学習済みモデルで正則化するFine-tuning手法

- 事前学習済みのモデルFine-tuningを**物体検出**で評価。
- 事前学習済みモデルの重みで正則化、精度向上

正則化項

$$\Omega(w) = \sum_i \| \underbrace{w_i}_{\text{学習モデル}} - \underbrace{w_i^{pre}}_{\text{事前学習済みモデル}} \|^2.$$

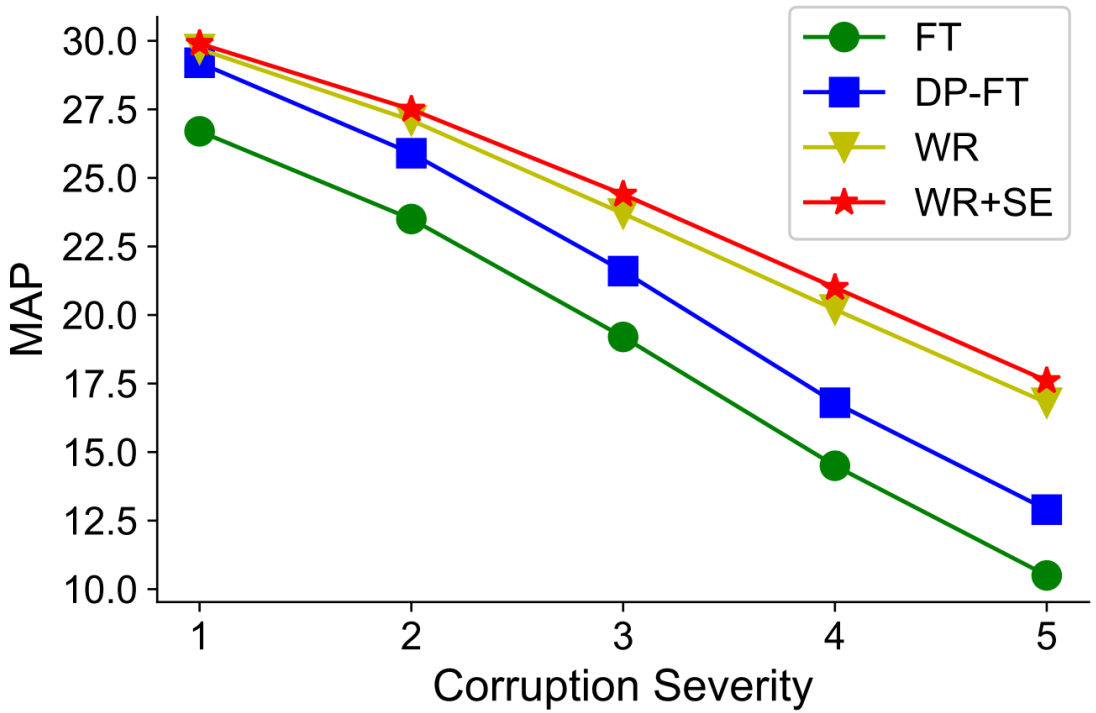
事前学習済みモデルがロバストなのは、既知。
パラメータを学習する際に大きくそこから離れないようにしたい



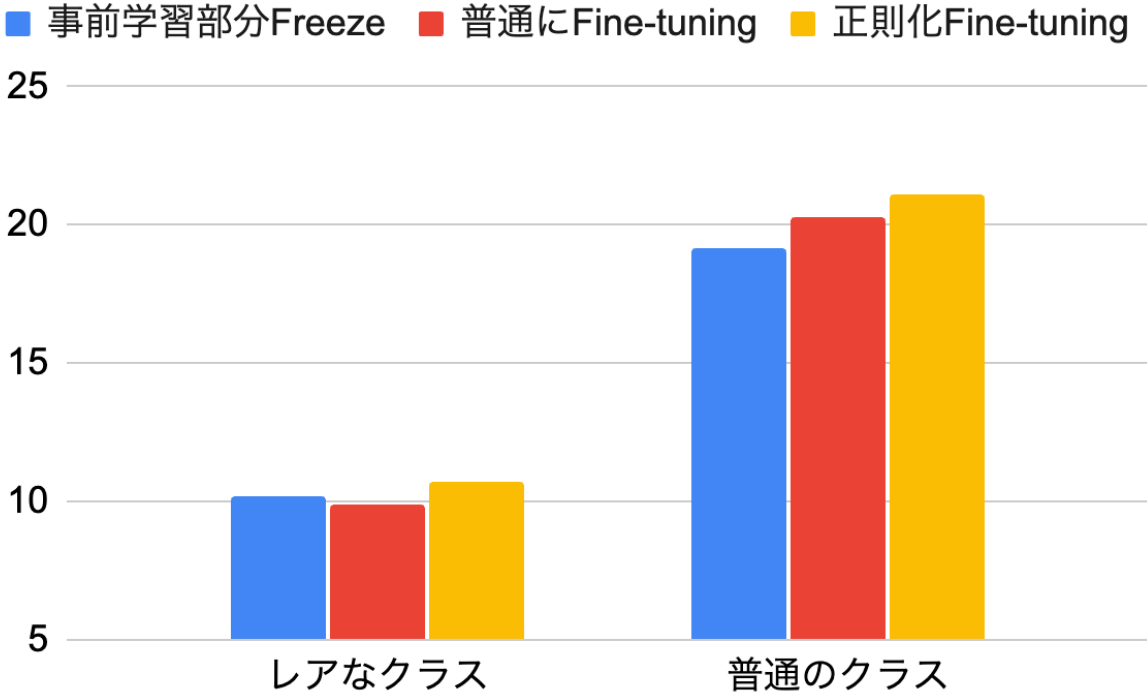
Mind the Backbone: Minimizing Backbone Distortion for Robust Object Detection, Arxiv 2022

事前学習済みモデルは様々なShiftに有効

Covariate-shift (様々なノイズ)に対するロバストさ



Target-shift (ラベルの分布) に対するロバストさ



・学習クラスをサンプル数によって分類。
・普通にTuningするとレアなクラスで精度が落ちる。
・正則化すると、精度向上。

正則化なし

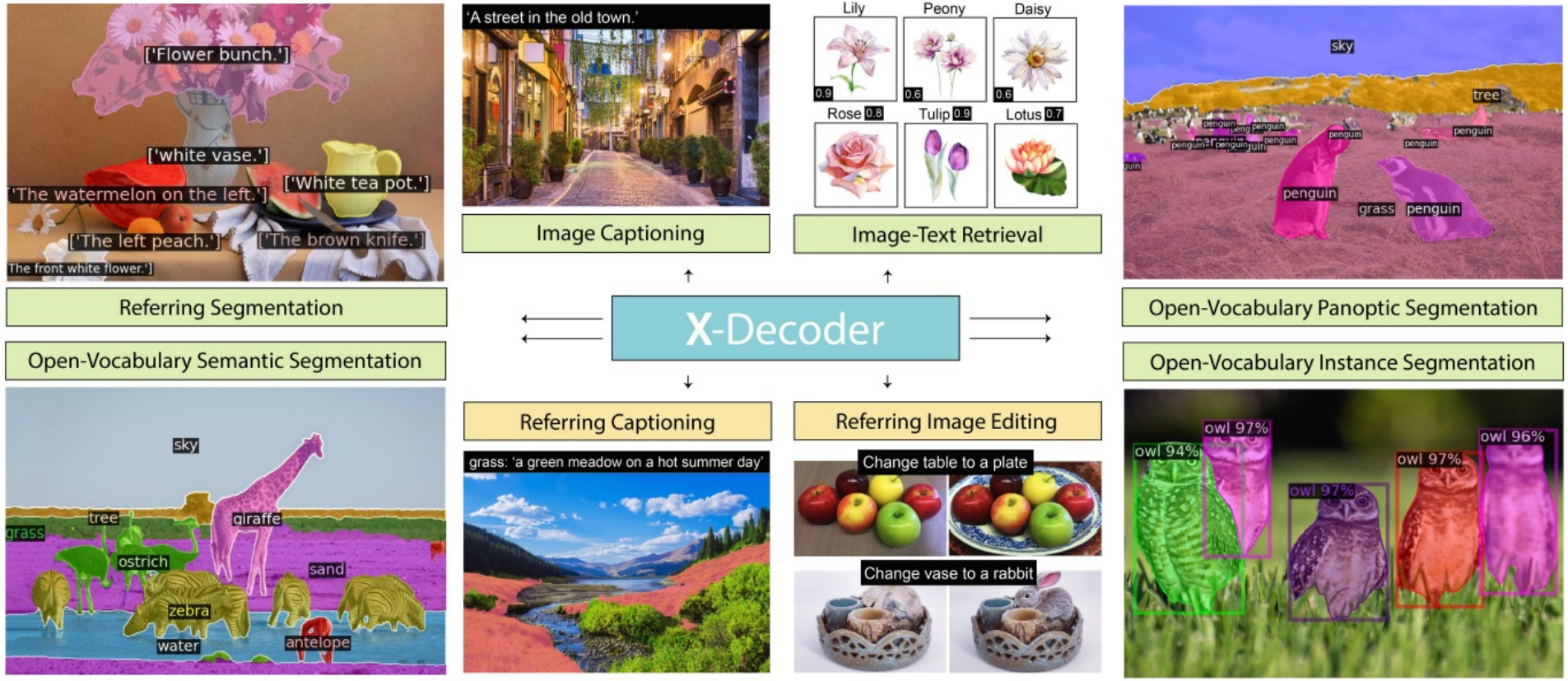
正則化あり

・COCOで学習。Augmentationは特にしていない。
・学習データには夜の画像は少ない
・葉っぱ？塵？がノイズになっている。

https://www.youtube.com/watch?v=RLxZW8wjRWM&ab_channel=TheWeatherNetwork

今後何が起こる？

- **Universalなモデルの研究は加速しそう**
  - よりFlexibleに, よりGeneralize



Generalized Decoding for Pixel, Image, and Language. CVPR2023