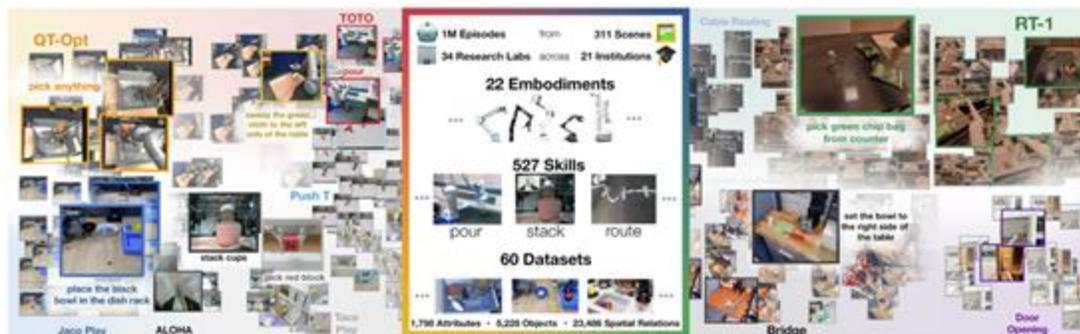


書誌情報

- Project Page : <https://www.physicalintelligence.company/blog/pi0>
 - Physical Intelligence社の論文
- 著者
 - Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, Ury Zhilinsky
- 概要
 - 新しいデータセットおよびアーキテクチャ、学習上の工夫によって SoTAなロボット基盤モデルを作成

背景

- ロボット基盤モデル
 - 異なる環境で複数のタスクへの適応が可能な汎用的なロボット制御モデル
 - 異なるロボット、異なるタスクからなるデータを大量に学習することで構築
 - 製造業など産業界のみならず、より環境やタスクが複雑な家庭内などへの進出が期待されている
 - ICRA2024では関連研究がBest Paperになるなど、近年注目を集めている
- ロボット基盤モデルの研究が進んでいる背景
 - 模倣学習の研究の発展：人がロボットを操作するデータから方策を学習できるように
 - スケーリングへの期待：大規模言語モデルと同様に、スケーリングによる性能向上・創発が期待される



<https://robotics-transformer-x.github.io/>

▶
学習

ロボット
基盤モデル

果物を掴む

机を拭く

ネジを回す

先行研究との比較

- ロボット基盤モデルの先行研究と $\pi 0$ の位置付け

- 入出力

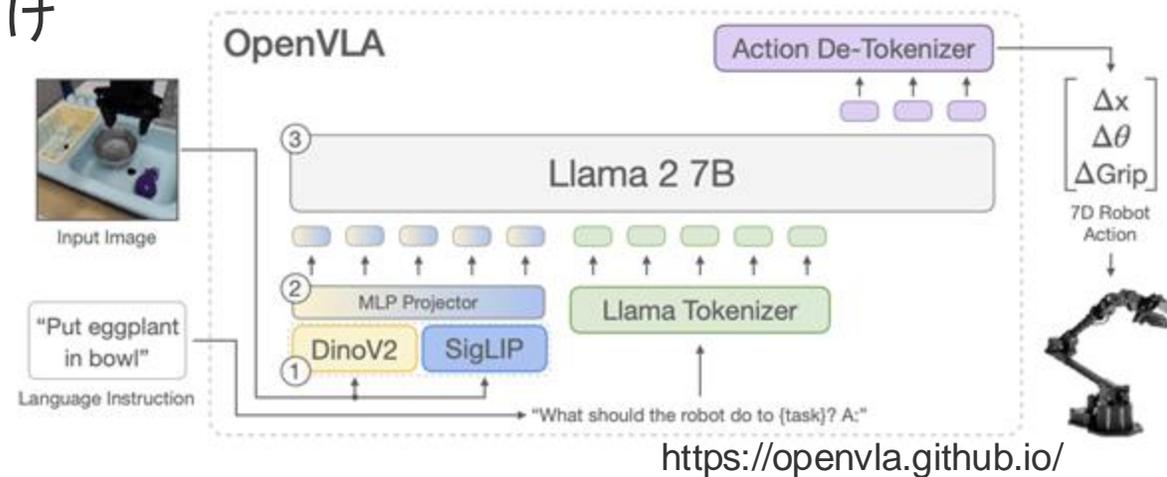
- 入力：タスク、観測画像、ロボットの状態
 - 出力：ロボットのaction

- アーキテクチャ

- VLM / Transformer Only

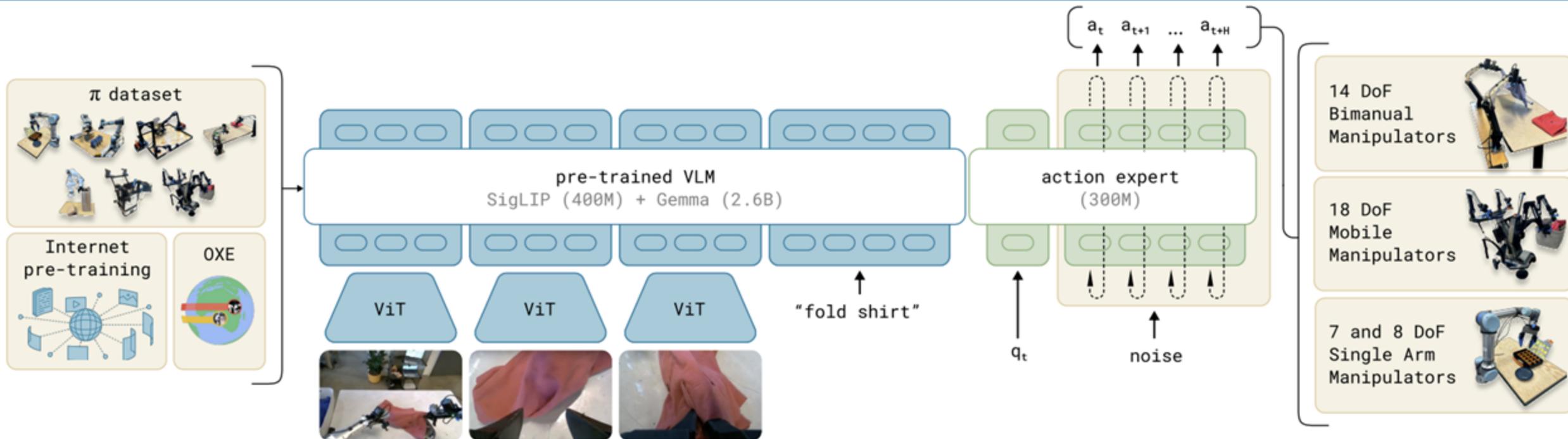
- データセット

- Open X-Embodiment(OXE)が主流



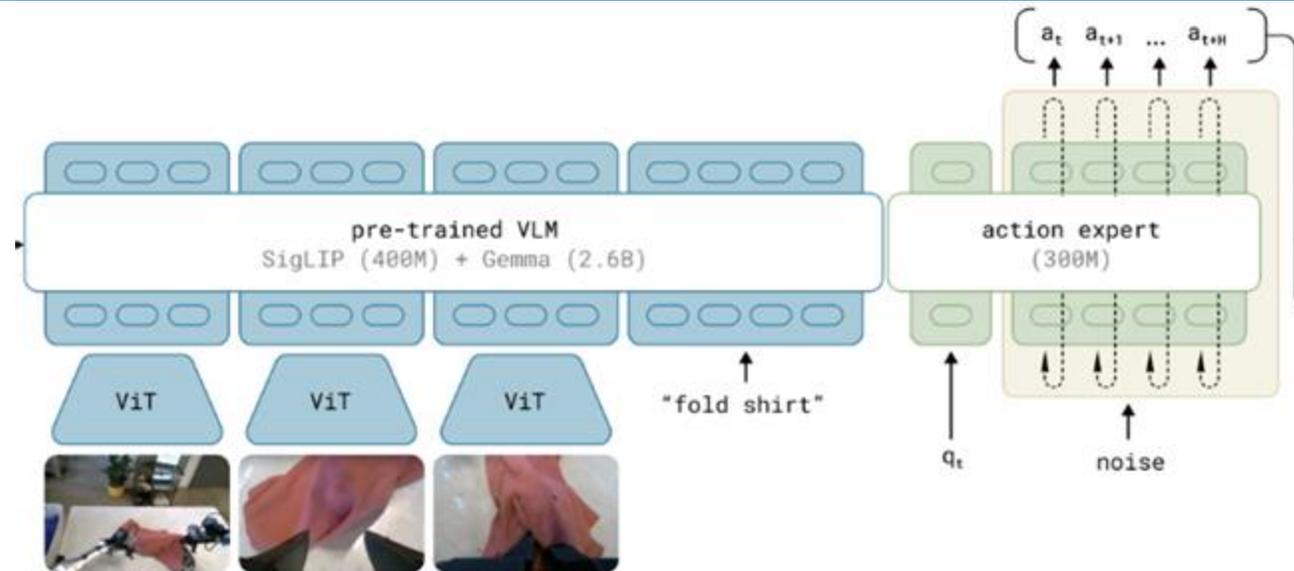
モデル名	データセット	アーキテクチャ	action生成
RT-1-X	OXE	Transformer (35M)	De-tokenize
RT-2-X	OXE	VLM (55B)	De-tokenize
Octo	OXE	Transformer (93M)	Diffusion Policy
OpenVLA	OXE	VLM (7B)	De-tokenize
$\pi 0$	π dataset(903M steps) + OXE	VLM (3.3B)	Flow Matching

モデルアーキテクチャ



- Pre-trainedされたVLM(PaliGemma)にaction生成部(action expert)を追加
 - 複数ステップの予測を行う(Action Chunking)
 - 実際には50ステップ分を一度に予測
- 異なるカメラ数や自由度を持つロボットデータを同時に学習可能
 - 次元が小さいロボットデータにはゼロパディングを行う
 - 3枚未満の画像が入力される場合はマスクを行う

モデルアーキテクチャ



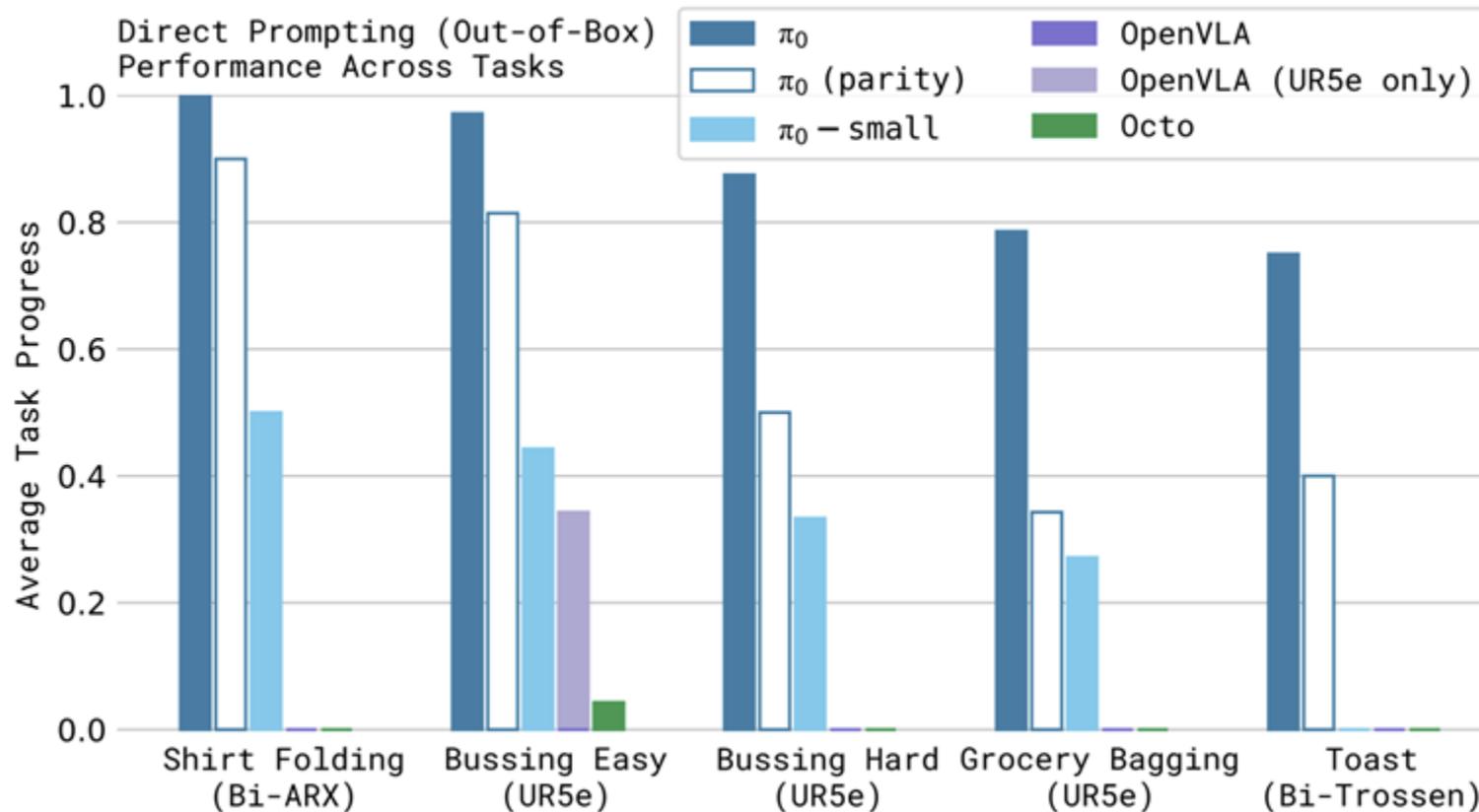
- Flow Matchingを利用したaction生成
 - 連続的なactionを生成するために、**Conditional Flow Matching**を使用
 - **高い精度と多峰性**を備え、高周波数制御が必要なdexterous(器用な)タスクに向いている
 - 実験でdiffusion policyなどと比較
- 複数の目的関数で単一モデルを学習
 - 離散データと連続データで同時に単一モデルを学習する手法の、Transfusionを参考にした
 - フローマッチングロスとクロスエントロピーロスをを用いた

データセットと学習

- 学習は**Pre-training**と**Post-training**の二段階で行う
 - Pre-trainingでは汎用的な物理的概念の獲得を目指す
 - Post-trainingでは下流タスクを器用にこなすことを目指す
 - Pre-trainingとPost-trainingでそれぞれ異なるデータセットを用意する
- **独自の π データセット**
 - 106M Stepsは単腕ロボット、797M Stepsは双腕ロボット(計 903M Steps、**10000時間**)
 - ロボットは**7種類**、タスクは**68種類**
 - 従来のロボットデータセットと異なり、「**抽象的な動詞**」で1つのタスクを表す
 - “Bussing”(片付け)タスクは「皿やカップを棚に収納」や「ゴミをゴミ箱に捨てる」を含む
- **Pre-trainingデータセット**
 - OXEデータセットと π データセットを混合して利用
 - オープンデータセット(OXE)の割合は**9.1%**
- **Post-trainingデータセット**
 - 下流タスクによってデータセットは異なり、5時間~100時間程度
- タスクによってデータ量にばらつきがあるため重み付けを行う

Pre-Trainingモデルの評価

- Pre-Trainingモデルの評価
 - ベースラインはOXE+ π で学習したOpenVLAおよびOcto
 - π_0 (parity)はOpenVLAとOctoの学習ステップ数に合わせたモデル
 - π_0 -smallは π_0 からVLMバックボーンを除外したモデル
- フォーマット、VLMアーキテクチャの有効性



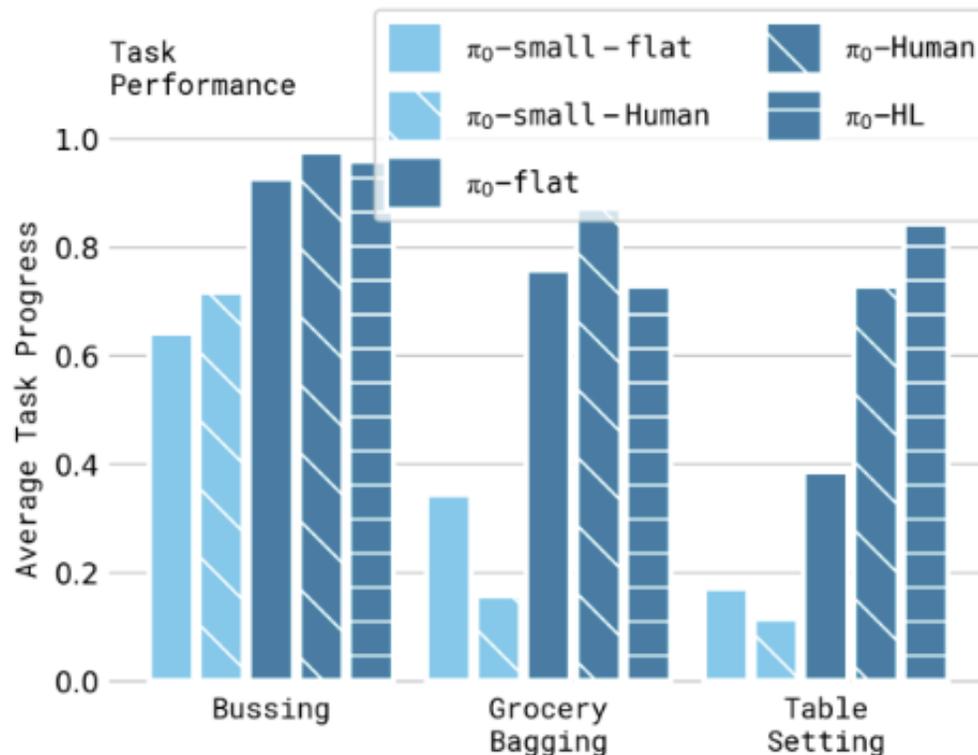
言語入力に対する性能評価

- 言語入力方法に対するタスク性能評価

- 評価タスクでPost-Trainingした π_0 を用いる

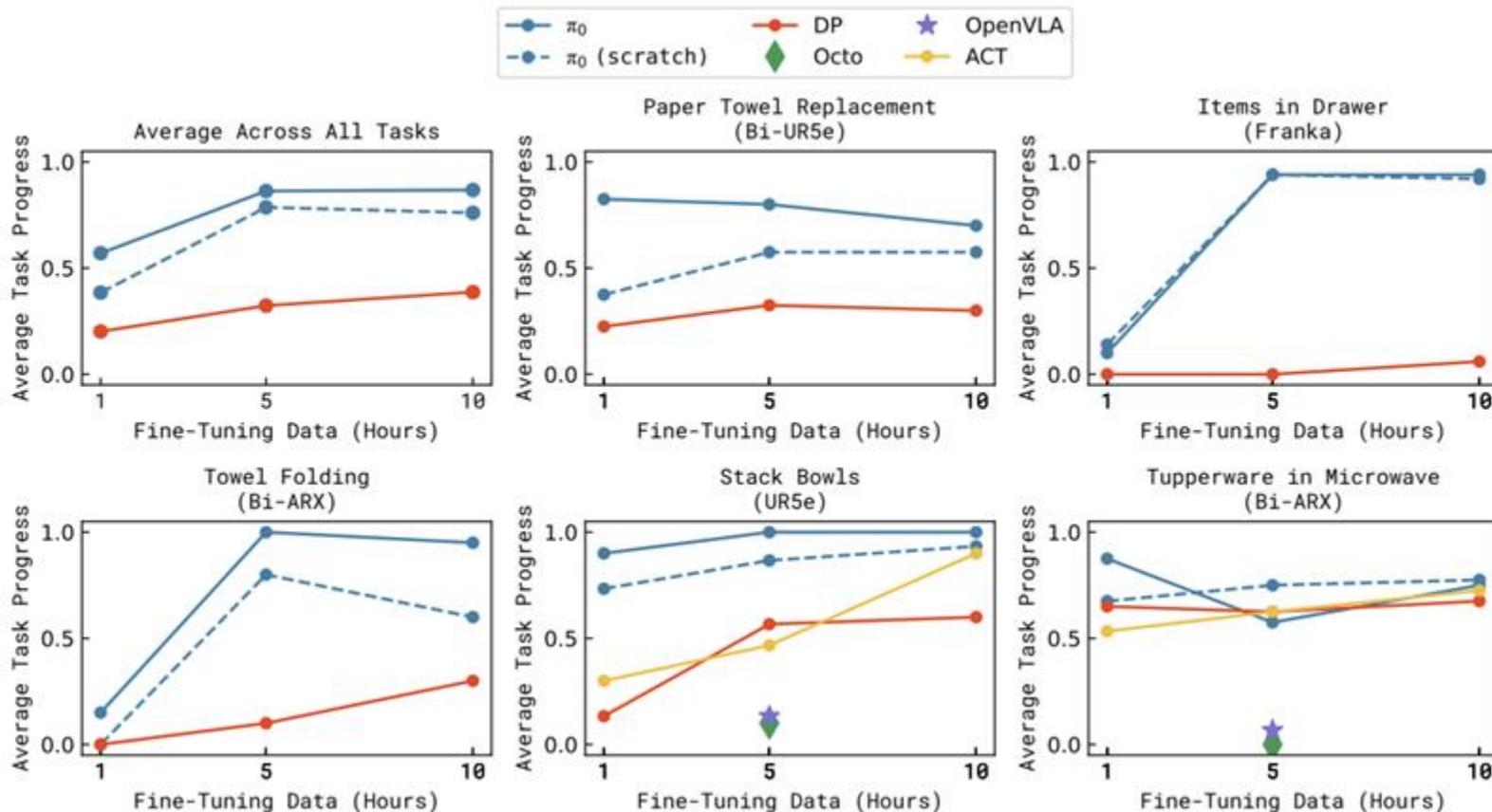
- -flat : タスクコマンドをそのまま入力
- -Human : 人がタスクを中間コマンドに分解し入力
- -HL : VLMポリシーに中間コマンドへ分解させ入力

→ π_0 では中間コマンドへの分解でタスク性能向上の傾向が見られた



器用なタスクでの評価

- 事前学習データとは異なる、器用な(dexterous)タスクでPost-Trainingした際の評価
 - π_0 (scratch)は事前学習を行わず、微調整用データセットのみで学習したモデル
 - ファインチューニングしたOpenVLA、Octoに加え、単一ポリシーのDiffusion Policy、ACTと比較
 - ベースラインより高い性能を示す傾向、事前学習によって性能向上の傾向



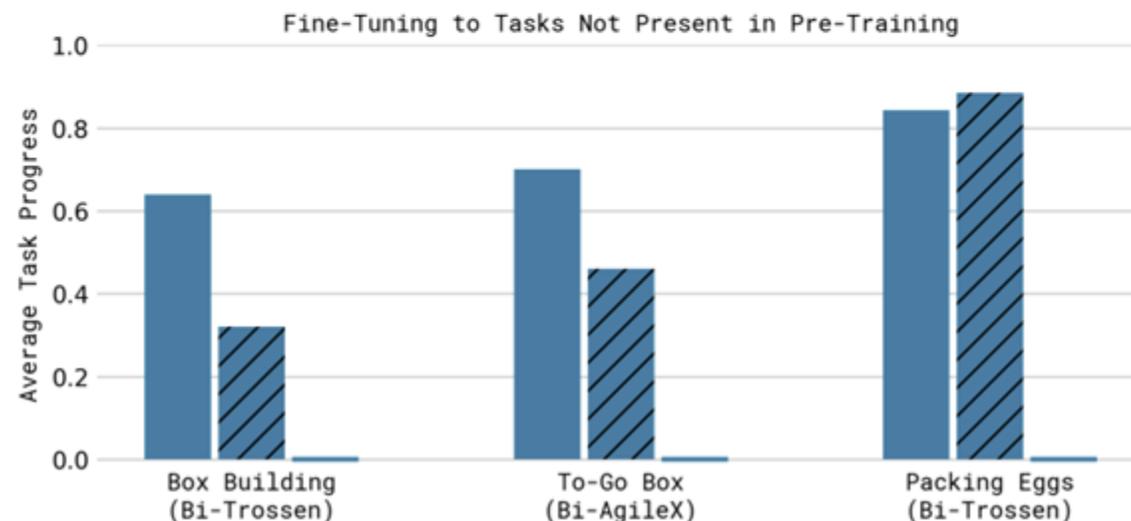
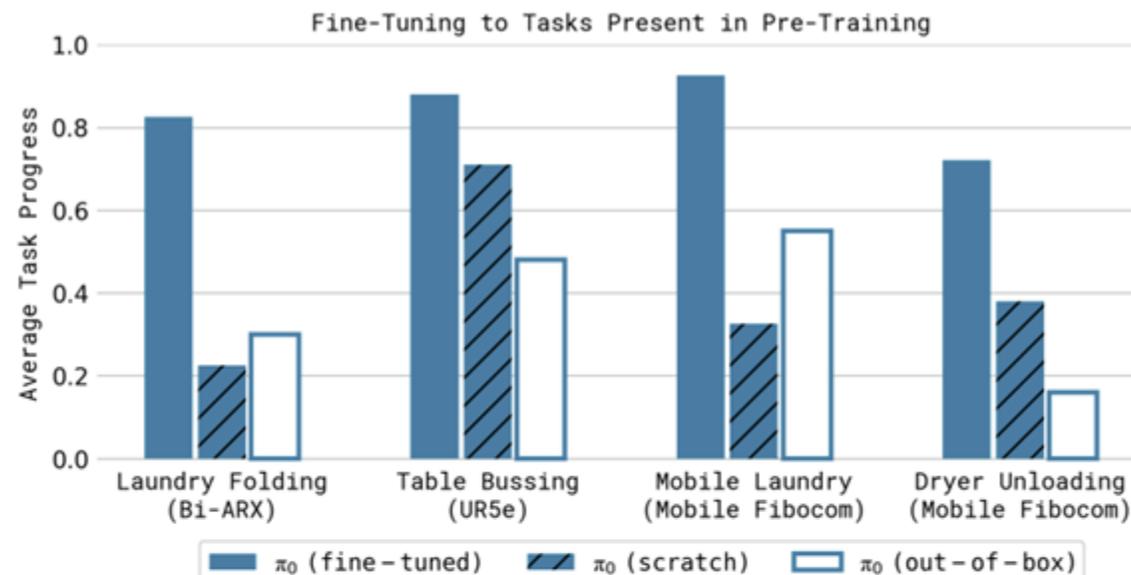
タスク詳細 (難易度昇順)

- 4つのボールを積み重ねる
- タオルを畳む
- タッパーをレンジに入れる
- ペーパータオルの交換
- ものを引き出しにしまう

複雑な多段階タスクでの評価

- よりlong-horizonで段階的なタスクでの評価
 - fine-tuned : Pre, Post-trainingあり
 - scratch : 微調整データセットのみで学習
 - out-of-box : Pre-trainingのみ
- タスク例
 - Table Bussing
 - テーブルの上のゴミや食器を片付ける
 - Box Building
 - 広がった状態の段ボール箱を組み立てる

→ 多くのタスクを実行することができる
やはり事前学習によって性能向上の傾向



Discussion / Limitation / Future work

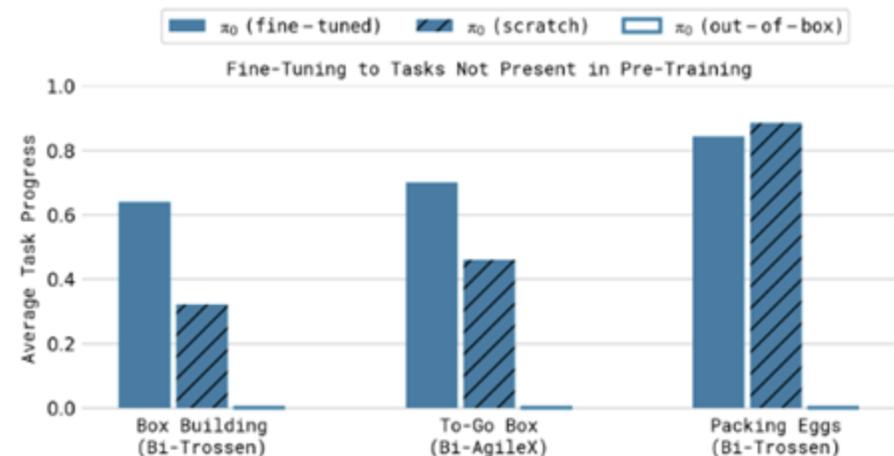
- ロボット基盤モデルにおいても**事前学習 / 事後学習が重要である可能性を示唆**
 - 事前学習のみだと、タスク実行能力が足りない
 - 高品質なデータでの事後学習のみだと、失敗から復帰できない
- **事前学習データをどのように構成すべきかまだ分かっていない**
 - あらゆる組み合わせを試したが、どのデータが有用か、どのように重み付けるかは不明
- あるタスクが解けるようになるまでに、**どのくらいの量や種類のデータが必要か分かっていない**
- 異なるロボットやデータセット間で**どれだけ転移の効果が期待できるのか不明**
 - 自動運転や脚ロボットなど、より広範なドメインに関して調査が必要

まとめ・感想

- π_0 は10000時間分もの巨大なロボットデータセットを学習したロボット基盤モデル
- VLMの利用、Flow Matchingを用いたaction生成に加えて、事前学習・事後学習プロセスの導入により、他の基盤モデルや単一ポリシーと比較し高い性能を達成

感想

- とにかくデモ動画とロボットデータ量が凄まじい
- 基盤モデルの性能は単一ポリシーと同等か劣るイメージだったが、印象が変わった
 - データの蓄積によって得られた汎用的な知識がタスク性能に寄与
- Pre-Trainingに含まれないタスクでのout-of-box性能が低いことから、Zero-shotでのタスク実行はまだまだ難しそう



(おまけ)推論tips

- 家庭向けGPUであるRTX4090で推論
 - 推論時間はネットワーク込で86ms
- Action ChunkingのHorizonは50
- Temporal Ensemblingは行っていない
 - 性能が低下してしまったため
- 推論・制御周期
 - 20Hzのロボットの場合
 - 0.8秒ごとに推論
 - 推論された50stepsの16stepsを実行
 - 50Hzのロボットの場合
 - 0.5秒ごとに推論
 - 推論された50stepsの25stepsを実行

model part	inference time
image encoders	14 ms
observation forward pass	32 ms
x10 action forward pass (flow)	27 ms
network latency (if off-board)	13 ms
total on-board inference	73 ms
total off-board inference	86 ms