

コンピュータの制御を学習するためのデータ駆動型アプローチ

ピーター・ハンフリーズ¹ デビッド・ラボソ¹ トビー・ポーレン¹ グレゴリー・ソートン¹ ラチータ・チャパリア¹ アリスター・マルダル¹
 ジョシュ・エイブラムソン¹ ペトコゲオルギエフ¹ アダム・サントロ¹ ティモシー・リリックラップ¹

概要

機械が人間と同じようにコンピュータを使用して、私たちの日常業務を支援できるようになれば便利です。これは、大規模な専門家のデモンストレーションとインタラクティブな行動の人間の判断を活用する可能性もある設定です。これらは、最近の AI の成功を後押しした 2 つの要素です。ここに投資します

自然言語で目標を指定し、キーボードとマウスを使用してコンピュータ制御の設定を行います。手作業で設計されたカリキュラムと特殊な行動空間に焦点を当てる代わりに、実際の人間とコンピュータの相互作用によって通知された行動の事前情報と組み合わせた強化学習を中心としたスケラブルな方法の開発に焦点を当てています。MiniWob++ ベンチマーク内のすべてのタスクで最先端の人間レベルの平均パフォーマンスを達成し、コンピュータ制御の問題の挑戦的なスイートを達成し、クロスタスク転送の強力な証拠を見つけました。これらの結果は、コンピュータを使用するようにマシンをトレーニングする際に、統合されたヒューマン エージェントインターフェイスの有用性を示しています。全体として、私たちの結果は、MiniWob++ を超えて、一般的に人間が行うようにコンピュータを制御する能力を達成するための公式を示唆しています。

1. はじめに

自然言語(Brown et al., 2020; Rae et al., 2021)、コード生成(Chen et al., 2021)、および 3D シミュレートされた世界におけるマルチモーダルインタラクティブ動作(Deep Mind Interactive Agents Team, 2021)に関する最近の研究顕著な表現力、文脈認識、および一般的な知識を備えたモデルを作成しました。この研究は、機械と人間の間で一致する豊かで構成的な出力空間、

¹ DeepMind、ロンドン、イギリス。連絡先: Peter Humphreys <peterhumphreys@deepmind.com>, Timothy Lillicrap <countzero@deepmind.com>.

また、大量の人間のデータと判断が機械の動作に影響を与えます。

これら 2 つの要素を備えているが、比較的注目されていない 1 つのドメインは、デジタル デバイス制御です(Shi et al., 2017; Liu et al., 2018; Nakano et al., 2021; Chen et al., 2021; Shvo et al., 2021; Li et al., 2020; Toyama et al., 2021; Gur et al., 2021)は、無数の本質的に有用なタスクを達成するためのデジタル デバイスの使用を含んでいます。デジタル情報をほぼ排他的に使用するため、このドメインはデータ取得と制御の並列化に関してうまく拡張できます(たとえば、ロボット工学や核融合炉と比較して)。また、多様でマルチモーダルな入力、表現力豊かで構成可能で人間と互換性のあるアフォーダンスと組み合わせます。この作業では、キーボードとマウスを使用したコンピュータ制御に焦点を当て、ピクセルとドキュメント オブジェクト モデル (DOM) を観察します。

コンピュータ制御の初期調査に役立つベンチマークは、MiniWob++ タスク スイート(Shi et al., 2017; Liu et al., 2018)です。これは、クリック、タイピング、フォーム入力などを必要とするタスクに続く一連の命令で構成されています。このような基本的なコンピュータの相互作用(図 1b)。MiniWob++ はさらに、プログラムで定義された報酬を提供します。これらのタスクは、人間が自然言語を使用してタスクを指定し、その後のパフォーマンスに関する判断を提供する、よりオープンエンドな人間とエージェントの相互作用への第一歩です(Li et al., 2016; Ammanabrolu et al., 2020; DeepMind Interactive Agentsチーム、2021年)。

私たちは、これらのタスクを解決するためのエージェントのトレーニングに焦点を当てています。これらの方法は、デジタル デバイスで実行することが望まれるすべてのタスクに原則として適用でき、望ましいデータおよびコンピューティングのスケールリング プロパティを備えています。したがって、強化学習 (RL) (Sutton and Barto, 2018) と行動クローニング (BC) (Pomerleau, 1989)の単純な組み合わせに目を向けます。後者は、人間とエージェントの行動空間(つまり、キーボードとマウス)。

これは、MiniWob の構想(Shi et al., 2017)で提案された組み合わせですが、その時点では、高スコアのエージェントを生成することはありませんでした。その後の作業では、エージェントに DOM 固有のアクション(Liu et al., 2018; Gur et al., 2018; Jia et al., 2019)、カリキュラム(Gur et al., 2018)、制約のある探索技術を通じて、

精選されたガイダンスを使用して、各ステップで利用可能なアクションの数 (Liu et al., 2018)、模倣と強化学習のシンプルでスケラブルな組み合わせを再検討すると、高いパフォーマンスを達成する上で欠落している主な要因は、単純に行動クローニングに使用される人間の軌跡データセットのサイズであることがわかります。パフォーマンスは、人間のデータの量が増えるにつれて確実に向上し、以前の研究で使用されたサイズの最大400倍のサイズのデータセットで継続的な改善が観察されました。このデータを使用した結果は、以前の最先端のパフォーマンスを大幅に上回り、タスクスイート全体で人間レベルの平均パフォーマンスを達成することさえできます。

2. 方法

2.1. ミニウォブ++

MiniWob++は、Liuらによって導入されたWebブラウザベースのタスクのスイートです。(2018) (以前のMiniWobタスクスイートの拡張(Shi et al., 2017))。タスクは、単純なボタンのクリックから、特定の指示が与えられたときにフライトを予約するなどの複雑なフォーム入力までさまざまです(図1a)。

各タスクにはプログラムによる報酬が用意されており、標準的な強化学習手法を使用できます。

MiniWob++に関する以前の作業では、DOM固有のアクションにアクセスできるアーキテクチャを考慮しており(Liu et al., 2018; Gur et al., 2018; Jia et al., 2019)、エージェントがDOM要素と直接対話できるようにします(マウスやキーボードを使用してナビゲートする必要はありません)。代わりに、マウスとキーボードベースのアクションのみを使用することを選択します。これにより、人間の行動データの使用が簡素化され、対話するコンパクトなDOM(および非Webブラウザベースのタスク)がなくても、このインターフェイスがコンピュータ制御タスクにうまく移行するという仮説がさらに立てられます。最後に、多くのMiniWob++タスクでは、DOM要素ベースのアクションでは実現できないクリックまたはドラッグアクションが必要です(図1bの例を参照)。

以前のMiniWob++研究(Liu et al., 2018; Gur et al., 2018; Jia et al., 2019)と同様に、特定のサーバーに入力する必要があるテキスト文字列の環境提供辞書へのアクセスをエージェントに与えます。タスクの入力フィールド(例については、付録の図9を参照してください)。これは、まばらなRL入力のみから生成テキストモデルを学習するという探索の問題を回避するのに役立ちます。興味深いことに、セクションで説明するように、3.4では、エージェントはこの入力がない場合でも最先端のパフォーマンスを実現できます。

MiniWob環境はリアルタイムであり、技術的およびアルゴリズム的な複雑さが生じます。たとえば、アクションが一時的にどのように観察につながるかについての保証はありません。観察は、コンピュータのリソースをめぐる競争による遅延の影響を受ける可能性があります。デモンストレーションデータを提供する人間の参加者のために、環境を実行しているサーバーに十分なリソースを確保します。

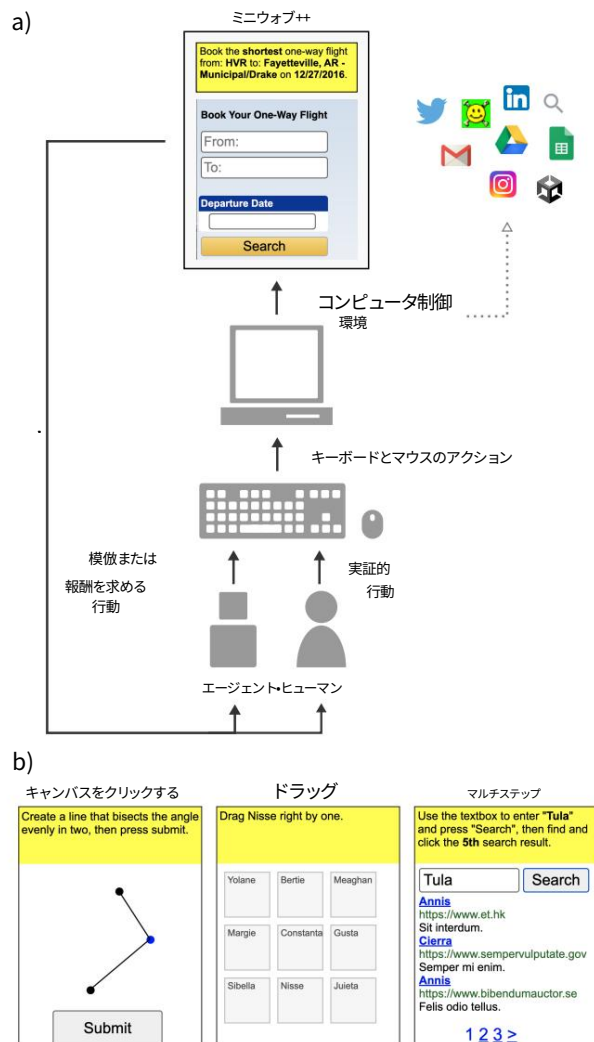


図 1. Mini Wob++ を実行するコンピューター制御環境。人間とエージェントの両方がキーボードとマウスを使用してコンピューターを制御し、人間は行動のクローン作成に使用される実証的な動作を提供し、エージェントはこの動作を模倣したり、報酬を求めて行動したりするように訓練されます。人間とエージェントは、MiniWob++タスクスイートの解決を試みます。このタスクスイートは、クリック、タイピング、ドラッグ、フォーム入力、およびその他の基本的な操作を必要とする指示に従うタスクで構成されています(「book-flight」タスクが描かれています)。b) さまざまな形式の対話を必要とする MiniWob タスクの例。

観測とアクションを 30Hz で同期するローカルマシンでの一時的なジッターを最小限に抑えます。エージェントにとって、一時的なジッターは、人間が経験するものとは確実に異なります。それにもかかわらず、ほとんどの MiniWob++タスクは比較的時間に敏感ではなく、私たちの結果は、不一致が実際には問題にならないことを示しています。さらに、デモデータを積極的に操作してアクションのないステップ(タイミングの違いを誇張するだけの操作)を削除すると、最良の結果が得られることがわかりました。以前の研究と同様に、人間とエージェントのスコア

完了までの時間による割引はありません。

2.2.環境インターフェース

エージェントが人間と同じようにコンピューターを使用する場合、エージェントは観察とアクションを送受信するための適切なインターフェースを必要とします。元の MiniWob++タスク スイートには、Selenium ベースのインターフェイスが付属しています(Liu et al., 2018)。Web ブラウザーで実行できるあらゆるタスクを柔軟にサポートするように設計された代替環境スタックを実装することにしました。このインターフェイスは、セキュリティ、機能、およびパフォーマンスのためにゼロから最適化されています(図 1a)。

安全。 Sandbox2 (GitHub リポジトリ)コンテナでWeb ブラウザー(この場合は Google Chrome)を実行します。これにより、chroot 監獄が提供され、ホスト システムの侵害に使用される可能性のあるシステム コールへのアクセスが制限されます。さらに、TCP プロキシ サーバーを使用して、すべてのネットワーク トラフィックをホスト側のローカル ソケットに再ルーティングします。これにより、ブラウザ内からアクセスできるコンテンツを正確に制御できます。これらのセキュリティ機能は、安全なエージェントと環境の対話にとって重要であるだけでなく、Amazon Mechanical Turk などの公開されているシステムでのデモ データの記録も簡素化します。ネットワーク リソース。

元の MiniWob++環境の実装は、内部ブラウザの状態にアクセスし、Selenium を介して制御コマンドを発行します。代わりに、Chrome DevTools Protocol (CDP) と直接やり取りして、DOM ツリーなどのブラウザの内部情報を取得し、実行します。

ページ上の開発者提供の JavaScript コード。起動時にブラウザに渡されるファイル記述子を介して CDP にアクセスします。これは、攻撃面を最小限に抑え、Web ドライバーの作成者が提供するセキュリティに関する推奨事項 (ChromeDriver のセキュリティに関する考慮事項) に従うためです。

特徴。人間のデスクトップ ユーザーが使用するのと同じコントロールを介して、エージェントが標準の Web ブラウザーとやり取りできるようにしたいと考えています。これを実現するために、私たちの環境はX11 サーバーに直接接続して、マウスとキーボードの制御を入力し、現在のフレームバッファを取得します。この最小値は、元の Seleniumインターフェースの課題である、人間とエージェント環境の相互作用の間のドメイン シフトを模倣します。たとえば、マウスのドラッグ操作はSelenium で実装するのが困難です。

X11 を使用することには、他にも重要な利点があります。(1) エージェントは完全なブラウザ (タブとアドレスバーを含む) とやり取りできる可能性がある。(2) 人間とエージェントの環境がほぼ同等であるため、一人称デモデータを大規模に記録するのは簡単です。、および (3) 状況依存システムのマウス カーソルをレンダリングできます。

環境スタック全体をC++で enに実装しました

低レイテンシの対話が可能です。これは、デモンストレーションを記録する場合に特に重要です。合計入力レイテンシが高すぎると、正確なマウス カーソルの移動が困難になるからです。

2.3.エージェントのアーキテクチャ

最終的に、たとえばグラフネットに基づいた特殊な DOM 処理アーキテクチャを実装する必要がないことがわかりました(Jia et al., 2019)。代わりに、マルチモーダルアーキテクチャに関する最近の研究に動機付けられて、以下と図2で説明するように、関連情報に柔軟に対応するためにマルチモーダルトランスフォーマーに主に依存して、最小限のモダリティ固有の処理を適用しました。

感知。エージェントは視覚入力 (165x220 RGB ピクセル) と言語入力 (入力の例を付録の図9 に示します) を受け取ります。ピクセル入力は、3x3カーネル、2.2.2.2 のストライド、および増加する出力チャンネル数 (32,128,256,512) で、一連の4つの ResNet ブロックを通過します。これにより、14x11の特徴ベクトルが生成され、154 個のトークンのリストにフラット化されます。

3種類の言語入力 - タスク命令、DOM、およびタスク フィールド (最後はポリシーにのみ使用される) - は同じモジュールを使用して処理されます。各テキスト文字列はトークンに分割され、各トークンはサイズ 64 の埋め込みにマップされます。370語の語彙を使用して、学習可能な埋め込みテーブルにインデックスを付けます。語彙外の入力単語用に、追加の1000 のインデックス可能な埋め込みを予約します。

これらの単語のインデックスは、64 ビットのハッシュ関数を使用して計算され、出力は370 ~ 1369の整数に縮小されます。4つのヘッドを持つ1層のトランスフォーマー (「言語トランスフォーマー」) を使用して、相互注意を使用して単一の512次元の埋め込みを生成する個々の文字列-トークンの埋め込みを使用してキーと値を生成し、追加の学習可能な埋め込みをクエリとして使用します。追加の学習可能な埋め込みにより、入力に依存しないコンポーネントが注目を集めます。これは、BERT で使用される特別な「CLS」トークン (Devlin et al, 2018) に類似しており、その出力は変換器の集約出力として直接使用できます。

マルチモーダル統合と記憶。ビジュアル入力埋め込み、DOMおよびタスク命令から生成された言語埋め込み、および2つの追加学習埋め込みが、8層、8ヘッド、および512次元埋め込みを備えたマルチモーダル トランスフォーマーに供給されます。

追加の埋め込み ding に対応する処理済みの出力は、残りの出力と前のアクションの埋め込みにわたる機能ごとの平均プーリング操作の結果と連結されます。結果として得られる1536次元のベクトルは、レイヤーごとに512個の隠れユニットを持つデュアルレイヤー LSTM に入力されます。残りの接続は、各LSTM レイヤーをバイパスします。

コンピュータの制御を学習するためのデータ駆動型アプローチ

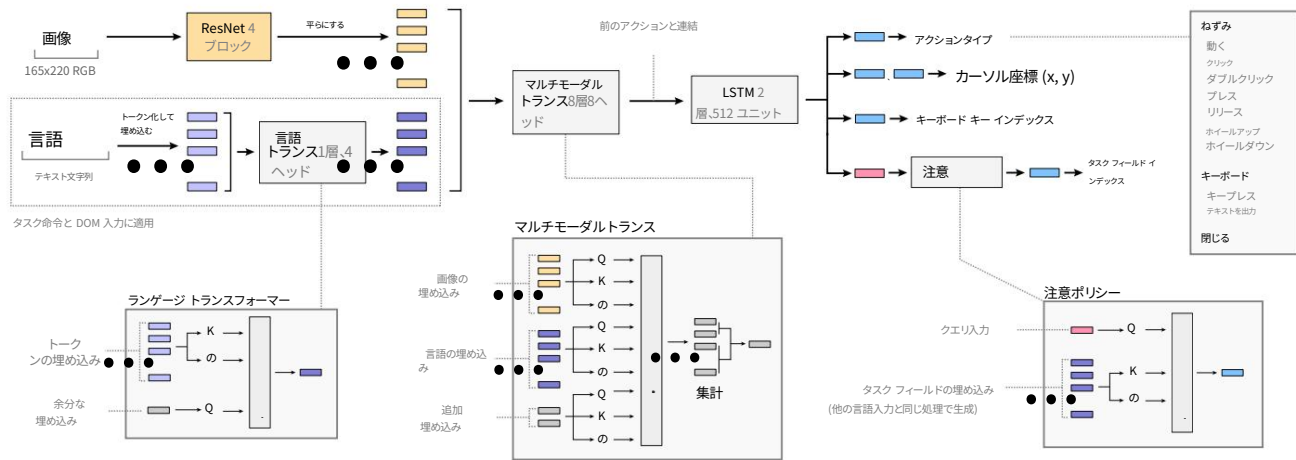


図 2. コンピュータ制御エージェントのアーキテクチャ (CC-Net)。ピクセル入力は 4 つの ResNet ブロックによってエンコードされます。言語入力 (タスク命令と DOM) はクロスアテンションによってエンコードされます。8 層のマルチモーダル トランスフォーマーは、ピクセルと言語の埋め込みを組み合わせ、その出力は前のアクションの埋め込みと連結され、2 つの LSTM を通過します。エージェントは、アクション タイプ、カーソル座標、キーボード キー インデックス、およびタスク フィールド インデックス (エージェントが出力するタスク フィールド文字列の 1 つを選択できるようにする) の 4 つの出力を生成します。すべての出力は、アテンション ベースのポリシーを介してロジットが生成されるタスク フィールド インデックスを除いて、線形変換を介して生成されます。

最終的な構成では LSTM アブレーションを実行していませんが、以前のチューニングでは、デュアルレイヤー LSTM がシングルレイヤー LSTM またはフィードフォワードアーキテクチャに比べてパフォーマンスが向上することが示されました。これは、チューニングでこれを制御しなかったため、デュアルレイヤー LSTM に関連するパラメータ数の増加が原因である可能性があります。移動要素を伴うタスクなど、メモリが特に役立つタスクがいくつかあることに注意してください。

ポリシー。エージェントのポリシーは、アクションタイプ、カーソル座標、キーボード キー インデックス、およびタスク フィールドインデックスの 4 つの出力で構成されます。各出力は、各次元を 51 個のピンに分割する 2 つの離散分布 (高さや幅の座標) によってモデル化されるカーソル座標を除いて、単一の離散確率分布によってモデル化されます。これは、高さの座標が幅の座標を条件とする、自己帰帰の唯一のポリシー コンポーネントです。

アクション タイプは、ノーオペレーション (アクションなしを示す)、7 つのマウスアクション (移動、クリック、ダブルクリック、プレス、リリース、ホイール アップ、ホイール ダウン)、および 2 つのキーボードを含む 10 の可能なアクションのセットから選択されます。アクション (キーを押す、テキストを出力する)。キーを押すアクションは、キーボードのキーまたは一連の小さなマクロ (CTRL+C など、付録表 2 の完全なリスト) の 1 つを発行するために使用されます。エミット テキスト アクションを使用して、MiniWob の「タスク フィールド」観測の 1 つによって指定された文字列をエミットできます。

残りのポリシー出力のサンプリングは、選択されたアクション タイプによって異なります。つまり、マウスの移動が選択された場合、カーソル座標がサンプリングされます。エミット テキストが選択された場合は、タスク フィールド インデックスがサンプリングされます (エミットするタスクフィールド文字列を決定するため)。key の場合、キーボード キー インデックスがサンプリングされます。

press が選択されました (発行するキーを決定するため)。

アクション タイプ、カーソル座標、およびキーボード キー インデックス分布のロジットは線形変換によって生成されますが、タスク フィールド インデックスのロジットは内積注意によって生成されます。このアテンション ポリシーは次のように機能します。まず、線形出力を使用してクエリを生成します。使用可能なタスク フィールドの埋め込みからキーを生成するために、別の線形変換が使用されます。

ロジットは、クエリとキーの間の内積とそれに続くソフトマックスから得られるアテンション ウェイトです。

セクション 3.4 のアブレーション実験では、エージェントが DOM で直接動作できるようにする 2 つの代替ネイティブ アクション タイプを使用します。特定の DOM 要素でクリックをトリガーするための DOM クリックと、タスクを直接書き込むための DOM エミット テキストです。フィールド文字列を指定された DOM 要素に変換します (つまり、このマクロ アクションでは、DOM 要素がフォーカスされ、そこにテキストが出力されます)。これらのアクションでは、クリックする DOM 要素を決定するために、追加のポリシー出力 (DOM 要素インデックス) が必要です。これは、上記の注意ポリシーを使用して作成されましたが、DOM 要素の埋め込みには注意が必要です。DOM エミット テキスト アクションでは、タスク フィールド インデックス出力を再利用して、エミットするテキスト文字列を決定しました。

2.4. ヒトデータ収集

エージェントと人間が同じインターフェースを使用することを考えると、元の MiniWob 出版物 (Shi et al., 2017) では、人間のデモンストレーションの使用が考慮されました。ただし、その研究のために収集された人間のデータはわずか 17 時間でした (純粋な行動クローニング ポリシーでは、5% 多いタスクしか解決できませんでした)。

コンピュータの制御を学習するためのデータ駆動型アプローチ

ランダムなポリシーと比較して)後の MiniWob++ 論文 (Liu et al., 2018) で使用されている少数のタスクについて、タスクごとにさらに1000のデモンストレーションがあります。データスケールがパフォーマンスに与える影響を示す最近の傾向(Deep Mind Interactive Agents Team, 2021; Kaplan et al., 2020 など) を考えると、人間のデモンストレーションを大量に使用することを再検討するのは自然なことです。

合計77人の人間の参加者から104のMiniWob++タスクの 240 万以上のデモンストレーションを収集しました。これは約6300時間に相当します。参加者は、クラウドソーシング データ収集プラットフォームを介して募集され、一定の時給が支払われました。人間のデータ収集は、倫理審査プロセスの対象でした。

人間は、練習しなくても MiniWob++タスクの多くをうまく実行できます。ただし、サブセットには、練習、専門知識の学習 (POSIX 端末コマンドなど)、テンプレート化された目標の意図におけるわずかなあいまいさの理解、またはタイミングやラグへの調整が必要です。

したがって、練習するにつれて人間のパフォーマンスが向上することを観察し (データは示していません) 、以下で説明するように、失敗した軌跡はデータセットから除外されました。

2.5. トレーニング

私たちは、模倣学習 (行動クローニング (BC) (Pomerleau, 1989))と強化学習 (RL) (VMPOアルゴリズム(Song et al., 2019)を使用) の単純な組み合わせを使用してエージェントをトレーニングしました。多くの挑戦的なドメインで大成功を収めています(Silver et al., 2016; Vinyals et al., 2019; Stiennon et al., 2020; Liu et al., 2018; Shi et al., 2017)。トレーニング ハイパー パラメーターは、付録の表1に記載されています。

模倣と強化学習を組み合わせるには、次のことができます。(2) BC を使用して事前にトレーニングし、その後 RL で調整します。発散ペナルティを使用して、または使用せずに、BCポリシーに戻します(Vinyals et al., 2019, eg)。私たちの研究では、BC と RL の損失に等しい重みで前者を考慮していますが、BC を使用した事前トレーニングが学習の効率を向上させる可能性があるという初期の証拠があります (データは示していません)。

BC に使用される前に、人間のデモンストレーション データはトレーニングセットとテストセット (それぞれ 220 万エピソードと 31 万エピソード) に分割されました。これらのエピソードは、成功によってさらにフィルタリングされました。つまり、最終的な報酬が0.5未満のタスクは削除されました(約 5%)。さらに、デモンストレーションから、人間が行動しなかったステップである「no op」ステップを、最大10連続ステップまで切断します。エージェントはこのデータで無差別にトレーニングされました。すべてのタスクは、デモンストレーション全体で均一なサンプリングを使用して共同トレーニングされました。これにより、収集されたデータ、フィルタリング手順、およびデモンストレーションの長さに応じて、タスク表現全体にわずかな非対称性が導入される可能性があります。

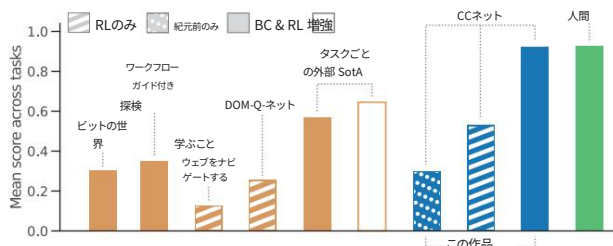


図3. MiniWob++ タスク全体の平均パフォーマンス。示されているのは、以前に公開された MiniWob++の結果(Shi et al., 2017; Liu et al., 2018; Gur et al., 2018; Jia et al., 2019) であり、それぞれがタスクのサブセットのパフォーマンスのみを報告しています (これらの結果について)。報告されていないタスクのパフォーマンスはゼロに設定され、使用されるすべてのスコアは付録の表3 に示されています。公正な比較を行うために、各タスクの最良の外部結果を集計された外部 SotA スコアに結合し、追加の拡張の有無にかかわらず得られた結果を報告します (対応する結果から、外部スコアのない 104 のタスクのうち16 をフィルター処理します)。私たちのエージェントである CC-Net は、これらの結果を大幅に上回り、平均的な人間のパフォーマンスに匹敵します。環境拡張を使用せずにこれを実現します。代わりに、BC と RL の組み合わせが CC-Net のパフォーマンスにとって重要であることがわかりました。BC および RL のみのトレーニングはあまり効果的ではありません。

2つの理由から、すべてのタスクで共同トレーニングを選択します。まず、共同トレーニングで見られる各タスクのフレームごとのトレーニングがより効率的になり、有意な転送効果があることがわかります(セクション3.2を参照)。第二に、私たちの最終的な目的は、一般的に有用なエージェントであるため、できるだけ多くの機能を備えた1つのエージェントが必要で

最後に、エージェントの計算要件により、限られた数の実験に基づいて手動で調整が行われました。さらに、この論文で報告する条件ごとに1つのシードしか実行できませんでした。

3. 結果

3.1. MiniWob++ での人間レベルのパフォーマンス

通常、論文は MiniWob++タスクのサブセットのみに取り組んでいるため、パフォーマンスを以前の文献と直接比較することは困難です。したがって、個々のタスクごとに公開されている最高のパフォーマンスを取得し、この集計されたパフォーマンスをエージェントの比較として使用します(カリキュラムやその他の拡張機能がある場合とない場合の両方で個別の集計を実行します)。私たちのエージェントは、この SotA ベンチマークのパフォーマンスを大幅に上回っています(図3、外部 SotA とのタスクごとの比較は、付録の図11 に報告されています)。さらに、一連の MiniWob++タスク全体で、エージェントが人間レベルの平均パフォーマンスを達成することがわかりました。このパフォーマンスは、BC と RL の共同トレーニングの組み合わせによって可能になります。

図6は、成功したエージェントの軌道のフレームを示しています。

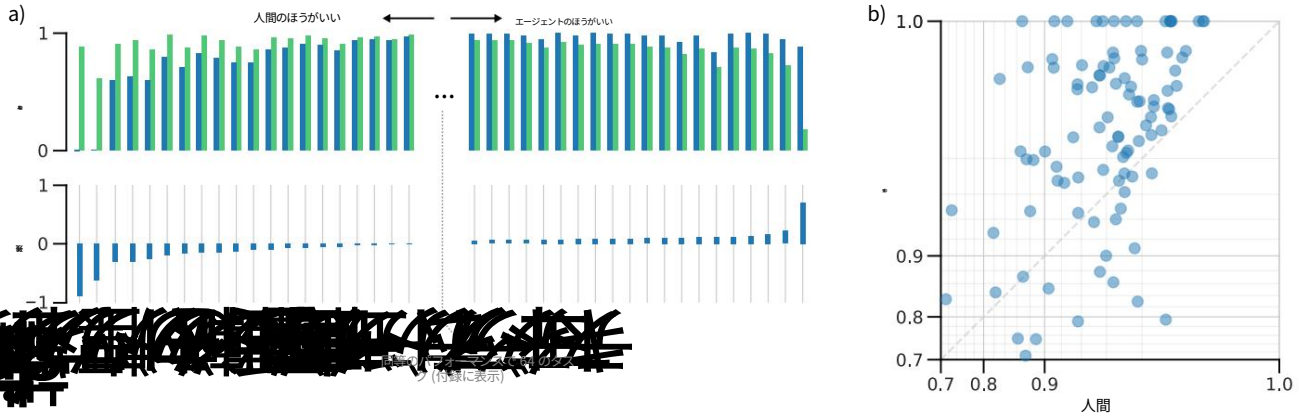


図4. CC-Netと人間のパフォーマンスのタスクごとの比較。a) 少数のタスクでは、人間(緑)がエージェント(青)よりも優れたパフォーマンスを発揮し、その逆も同様です。大多数の場合、人間とエージェントの両方がタスクをほぼ完全に解決するため、パフォーマンスの違いは最小限です(すべてのタスクについては、付録の図10を参照してください)。b) 右側の散布図に描かれているのは、3つの外れ値(人間またはエージェントのスコアのいずれかが < 0.7 である棒グラフの極値)を除いた104タスクすべてのパフォーマンススコアです。

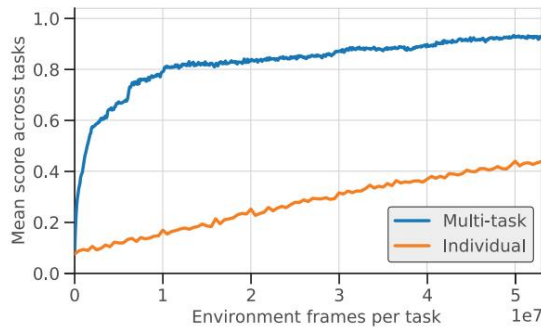


図5. シングルタスクとマルチタスクトレーニングの比較。104個のMiniWobタスクすべてを共同トレーニングすると、各タスクを個別にトレーニングするよりもデータ効率性が大幅に向上し、タスク間の転送の証拠が得られます。

挑戦的なタスクブックフライト。

タスクスイート全体のパフォーマンスを詳細に調べると、平均スコアの一致は人間のパフォーマンスを意味しますが、人間がエージェントよりも大幅に優れたパフォーマンスを発揮するタスクがあることがわかります(図4)。特に、エージェントはsimon-saysとterminalの2つのタスクでスコアを取得できません。simon-saysは、人間にとっても挑戦的です。ボタンを押すランダムなシーケンスを覚えてから、このシーケンスを繰り返す必要があります。

私たちのエージェントは約2Hzで環境を観察します。つまり、シーケンス内の少なくとも1つのボタンの表示を頻繁に見落とします。ターミナルでは、Unixターミナルを使用してファイルを検索および削除します。人間の参加者は、ほとんどがUnix端末の経験がなかったため、高いパフォーマンスを達成する前にこのタスクを実行するためのトレーニングが必要でした。エージェントがこのタスクでスコアを取得できなかった理由を理解するには、さらに調査する必要があります。

text-transform、simple-arithmetic、enter-text-2などの関連するキー押下タスク(text-transformの100エピソードのキー押下のヒストグラムについては、付録の図12を参照)。エージェントが平均的な人間のパフォーマンスを大幅に上回るタスクがいくつかあります。最大のパフォーマンスギャップは、ネットワーク関連の制御遅延が原因である可能性が最も高い、人間の参加者がほとんど成功しないアイテムの移動タスクです。

3.2. タスク転送

タスクごとのトレーニングステップの予算が固定されている場合、104個のMiniWob++タスクすべてで1つのエージェントをトレーニングすると、各タスクで個別のエージェントをトレーニングする場合と比較して、パフォーマンスが大幅に向上することがわかりました(図5)。これらのトレーニング体制を比較するために、個々のタスクで消費されるフレームの関数として表される各タスクのパフォーマンスを測定しました。プロットは、すべてのタスクの平均を示しています。これらの結果は、このような統合された制御インターフェイスが、さまざまな範囲の人間のタスクにわたる一般化をサポートする可能性について有望なヒントを提供します。同様の結果が、Jia et al.で小規模(同時に9つのタスクのトレーニング)で見つかりました。(2019)。

3.3. スケーリング

人間の軌跡データセットのサイズは、エージェントのパフォーマンスにおける重要な要素です(図7)。データセットの1/1000(およそ6時間分のデータに相当)を使用すると、オーバーフィッティングが急速に進み、RLのみのパフォーマンスよりも大幅な向上はありません。このベースラインからのデータ量を3桁以上、最大でデータセットサイズまで増やすと、継続的にパフォーマンスが向上することがわかります。これらのデータセットサイズでより高いパフォーマンスが可能になる可能性があることに注意してください。



図 6.ブック フライトで成功したエージェントの軌跡から選択された手順。このタスクには19億を超える可能な順列があり、純粋な強化学習の重要な探索課題を表しています。BC と RL の組み合わせで問題を解決できることがわかり、人間のデータを使用して行動の事前確率を形成することの有用性が実証されました。

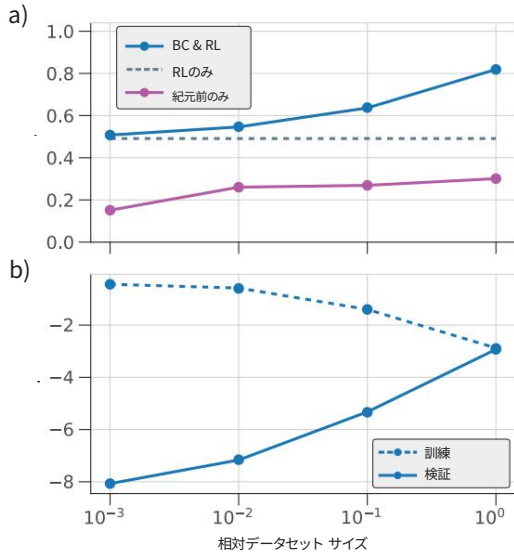


図 7.さまざまなデータセット サイズに対するエージェントのパフォーマンス:
a) さまざまなデータセット サイズを使用した BC と RL の共同トレーニングの 20 億フレームでのパフォーマンス。また、最大 BC のみのトレーニングパフォーマンスも示されています。b) BC と RL の共同トレーニング下での、列車 (実線) および検証 (破線) の軌道における人間の行動のエージェント予測の対応するログ確率。ご覧のとおり、データセットのサイズが小さいと、大幅な過適合が発生します。

アルゴリズムまたはアーキテクチャの変更に伴い。たとえば、データセットのサイズを縮小したとき、実際には代わりに KL ダイバージェンスを使用の方が効果的である場合に、BC 損失の相対重み (相対サイズの平方根でスケール) を単純に減らしてオーバーフィッティングを遅くしました。検証損失が最小になるようにトレーニングされた、事前トレーニング済みの BC のみのポリシーに対するペナルティ。とはいえ、低データ体制で効率を向上させる手段を探るために必要な研究と実験のコストは、より多くのデータを収集する単純さとバランスを取る必要があります。

3.4. インプットアブレーションとアウトプットアブレーション

私たちのエージェントは、ピクセルと DOM 情報の両方を消費し、さまざまな可能な範囲をサポートするように構成できます。

行動。これらの異なるアーキテクチャの選択の重要性を理解するためにアブレーションを行います。

最初に、さまざまなエージェント入力を除去します (図8a)。現在のエージェント構成は DOM 情報に大きく依存しており、この入力が増加されるとパフォーマンスが 75% 低下します。対照的に、エージェントは視覚情報なしでの操作にはあまり敏感ではありません。これは、形状や線を含む「キャンバス」の処理が必要なタスクにも当てはまります。

DOM を調べると、MiniWob++ の場合、このキャンバスは DOM 内できれいに表現され、ピクセル情報がなくてもエージェントがそのような情報を利用できることがわかります。これは、より一般的なコンピューター制御タスクには当てはまらず、エージェントのピクセルのみのパフォーマンスを改善することの重要性を示しています。

図 8b では、環境によって与えられたテキスト入力の選択肢 (タスクフィールド) を使用するエージェントの能力を削除するアブレーションを示します。興味深いことに、このようなエージェントはフォーム入力を含むタスクを解決できますが、人間の軌跡から、テキストを強調表示し、これに関連するテキスト ボックスにドラッグすることでこれを行うことを学習します。特に、このドラッグアクションは、元の Selenium バージョンの環境でエージェントが実行するのは簡単ではありません。

さらに、エージェントが指定された DOM要素 (セクション2.3 で説明) と対話する代替アクションを使用するアブレーションを図8b に示します。これは、キャンバス内の特定の場所でのクリック、ドラッグ、または強調表示を含むタスクをエージェントが解決できないことを意味します。このエージェント アーキテクチャを最適化していないため、記録された人間の軌跡とエージェントが利用できるアクションとの間には大きな不一致があります。模倣学習中、私たちは単にこれらのアクションを無視しますが、それにもかかわらず、エージェントがDOM ベースのアクションにとって必ずしも最適ではない軌道に向かって動かされることにつながります。追加の実験では、元のアクション セットを DOM ベースのアクションで拡張しても、ベースラインを超えるパフォーマンスの向上にはつながりませんでした (データは示していません)。

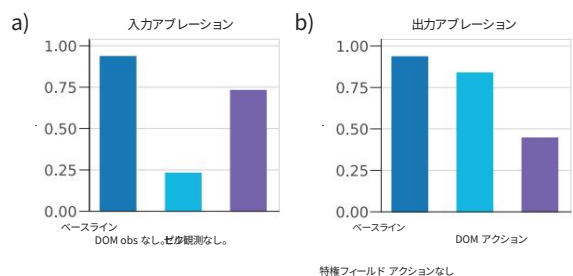


図 8. アブレーション。 a) エージェント入力アブレーション: DOM 入力観察を削除すると、パフォーマンスが大幅に低下しますが、ピクセル観察を削除しても害は少なくなります。 b) 出力アブレーション: エージェントは、環境によって提供されるテキスト入力の選択肢 (タスク フィールド) がなくても、驚くほど効果的です。これは、マウスを使用してテキストをプロンプトから必要な入力フィールドにコピーすることを学習できるためです。エージェントが指定された DOM 要素と直接対話するアクション スペースを使用することは、ベースラインのマウスおよびキーボード アクションよりも効果的ではありません。

4. 議論、制限、および関連作業

デバイスの支援。デジタル デバイスで私たちが効果的に支援するエージェントを作成するには、(1)人間の意図、言語、および判断を理解すること、および (2)言語で指定された目標をデバイス上でアクションに変える能力が必要です。MiniWob++ベンチマーク(Liu et al., 2018)は、2 番目の要件を研究するための出発点として機能します。エージェントは、言語の目標を最新の Web ブラウザーのアクションに変換する必要があります。人間のデータで強化された深層強化学習がこの問題を解決するのに適していることを示していますが、MiniWob++ベンチマークは最初の要件の複雑さを隠しています。(1)と(2)を協調して取り組み、デバイス上で有用なエージェントを作成する作業が残っています。

MiniWob++は、コードによって結果に報酬を与えるように正確にマッピングされた自然な (ただしスクリプト化された) 言語を介して目標を規定し、人間の意図、言語、および好みを扱う難しさを脇に置きます。したがって、報酬情報はノイズが少なく、遅延がなく、大量に利用できます。最近の研究では、強化学習エージェントを複雑な人間の好みや人間の言語に合わせる方法が研究されています。たとえば、Christiano ら。(2017) 強化学習を介してエージェントをトレーニングし、人間のフィードバックに基づいた報酬を使用して、Atari をプレイし、運動制御タスクを実行しました。

人間のアラインメントへの関心は加速し(Wirth et al., 2017; Ibarz et al., 2018)、対話エージェントを含むさまざまな領域に広がっています(Jaques et al., 2019)。並行して、Transformer アーキテクチャ上に構築された大規模な言語モデル(Vaswani et al., 2017)は、自然言語を消費および生成するエージェントを構築する能力に革命をもたらしました。GPT-3 (Brown et al., 2020)および関連研究(Shoeybi et al., 2019; Rae et al., 2021)は、膨大な量の人間が生成したテキストの事前トレーニングにより、高速で柔軟な学習者であるモデルが作成されることを示しました。めまいがする配列

言語理解タスクの。Codex (Chen et al., 2021)と呼ばれる最近の言語モデルは、これらの「基礎モデル」(Bommasani et al., 2021)が公開されているコードでモデルを微調整することにより、自然言語の意図をコンピュータと結び付ける可能性を示しています。

これらの断片を結び付ける一連の研究が始まりました。(2020)クラウドソーシングによるファンタジーテキストゲームで、目標を追求するために行動し、コミュニケーションをとるエージェントを構築します。大規模な言語モデルと人間のデモンストレーションを組み込んで、効果的に目標を追求するために行動し、話すことができるエージェントを導き出します。DeepMind Interactive Agents Team (2021)は、人間のデモンストレーションと判断を使用して、3D ゲーム環境で人間の言語と意図に反応するエージェントを構築します。ここで最も重要なのは、WebGPT (Nakano et al., 2021)が大規模な言語モデリングの結果に基づいて構築され、人間のフィードバックによって強化されたブラウザ支援の質問応答エージェントを作成することです。このエージェントは、検索、リンクの追跡、ソースの引用を可能にするスポーク テキスト インターフェイスを使用してインターネット上で動作します。インターネットを使用することで、エージェントは人間の参加者によって判断される質問応答能力を向上させます。

データ。人間レベルのパフォーマンスを達成するには、最初に収集されたデータ(Shi et al., 2017)の約400 倍のデータが必要です。これは最初のコレクションに比べてかなりの量ですが、1 回限りのコストでもあり、たとえば言語モデリングに使用される最近のデータセットよりも桁違いに小さくなっています。さらに、少量のデータで適切に機能する新しいアルゴリズムとアーキテクチャの実験に関連するコストは、単純により多くのデータを収集するために必要なコストをすぐに上回ってしまう可能性があります。

私たちのデータセットが MiniWob++タスク プレゼンテーションのすべての可能な順序を使い果たし、タスクを「解決」するタスクを単純な記憶に変えてしまうのではないかと心配する人もいかもしれません。ただし、タスクの提示の可能な順序は、収集したデータの量を大幅に超えるため、これは真実とはほど遠い。さらに、マウス カーソルの初期位置など、環境に組み込まれた自然な変動があります。最後に、個別トレーニングとマルチタスク トレーニングの大きな違いは、他のタスクから移転可能なスキルを得ることができる多くのタスクが存在することを示唆しています。

パフォーマンス。パフォーマンスを向上させる可能性のある未踏のアプローチが多数あります。たとえば、より困難なタスクが過剰に表現されるように、行動クローン データの表示を歪めることができます。または、RL と BC の目的を調整することもできます。おそらく、事前に訓練された BC と減衰する KL ペナルティを使用して、RL中に BC ポリシーに戻します(Vinyals et al., 2019)。それにもかかわらず、この作業の目標の一部は、簡単な方法の有効性を示すことでした。

最小限のオプションで実行されるテクニック。

私たちのアプローチは、アルゴリズムの目新しさの最先端にあるわけではありませんが、元の MiniWob 作業で導入された具体的な仮説に解決策を提供します。つまり、キーボードとマウスを含むコンピューター上のタスクを解決するための深層強化学習法の有効性です。実際、この分野の文献を調べると、従来のアプローチでは不十分であり、より専門的な技術が必要であることが示唆されます。標準的なディープ RL 手法を強化する際のデモンストレーション データの役割を理解する限り、これが当てはまらないことを示しました。

結論。人間は、毎日何十億時間もデジタル デバイスを使用しています。これらのタスクのほんの一部でも支援できるエージェントを開発できれば、エージェント支援の好循環に入り、その後失敗に対する人間のフィードバックが続き、エージェントの改善と新しい機能につながることを期待できます。

5. 貢献

AS, TP, TL, PCH がこのプロジェクトを考案しました。AG がプロジェクトを管理しました。GT, AS, TP, AM, PCH, PG, RC, および JA は、環境、データ収集、実験、およびエージェント インフラストラクチャを開発しました。AS, DR, および PCH は、エージェントアーキテクチャを開発し、実験を実行し、データを分析しました。AS, DR, TL, PCH が原稿を書きました。

6. 謝辞

DeepMind Interactive Agents Team ([DeepMind Interactive Agents Team](#), 2021 年) には、インフラストラクチャを実験のベースとして使用することを手伝ってくれたことに感謝します。また、人間のデータ収集パイプラインと環境に貢献してくれた DeepMind Platform Team と Crowd Compute にも感謝します。

参考文献

Sandbox2 GitHub リポジトリ。 https://github.com/google/sandboxed-api/tree/main/sandboxed_api/sandbox2. [オンライン; 2022 年 1 月 20 日にアクセス]。

Chrome ウェブドライバー 安全 考慮事項。 <https://chromedriver.chromium.org/security-considerations>. [オンライン; 2022 年 1 月 20 日にアクセス]。

P. Ammanabrolu, J. Urbanek, M. Li, A. Szlam, T. Rocktaschel, および J. Weston. ドラゴンをやる気にさせる方法: 目標主導のエージェントに、ファンタジーの世界で話したり行動したりするように教えます。 arXiv プレプリント arXiv:2010.00685, 2020.

R. ボンマサニ, DA ハドソン, E. アデリ, R. アルトマン,

S. Arora, S. von Arx, MS Bernstein, J. Bohg, A. Bosse lut, E. Brunskill, 他. 基礎モデルの機会とリスクについて。 arXiv プレプリント arXiv:2108.07258, 2021.

TB Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, 他言語モデルは数ショット学習器です。 arXiv プレプリント arXiv:2005.14165, 2020.

M. Chen, J. Tworek, H. Jun, Q. Yuan, HP d. O. ピント, J. カプラン, H. エドワーズ, Y. ブルダ, N. ジョセフ, G. ブロックマン, 他コードでトレーニングされた大規模な言語モデルの評価。 arXiv プレプリント arXiv:2107.03374, 2021.

P. Christiano, J. Leike, TB Brown, M. Martic, S. Legg, および D. Amodei. 人間の好みからの深層強化学習。 arXiv プレプリント arXiv:1706.03741, 2017.

DeepMind インタラクティブ エージェント チーム. 模倣と自己教師あり学習によるマルチモーダル インタラクティブ エージェントの作成。 arXiv プレプリント arXiv:2112.03763, 2021.

I. Gur, U. Rueckert, A. Faust, および D. Hakkani-Tur. ウェブをナビゲートすることを学びます。 arXiv プレプリント arXiv:1812.09195, 2018.

I. Gur, N. Jaques, Y. Miao, J. Choi, M. Tiwari, H. Lee, および A. Faust. ゼロショット構成強化学習の環境生成. 神経情報処理システムの進歩, 34, 2021.

B. Ibarz, J. Leike, T. Pohlen, G. Irving, S. Legg, および D. Amodei. 人間の好みやアタリでのデモンストレーションから学ぶことに報いる。 arXiv プレプリント arXiv:1811.06521, 2018.

N. Jaques, A. Ghandharioun, JH Shen, C. Ferguson, A. Lapedriza, N. Jones, S. Gu, および R. Picard. 対話における暗黙の人間の好みのポリシーからかけ離れたバッチ深層強化学習。 arXiv プレプリント arXiv:1907.00456, 2019.

S. Jia, J. Kiros, および J. Ba. Dom-q-net: 構造化言語に基づいた rl. arXiv プレプリント arXiv:1902.07257, 2019.

J. カプラン, S. マキャンドリッシュ, T. ヘニハン, TB ブラウン, B. チェス, R. チャイルド, S. グレイ, A. ラドフォード, J. ウー, D. アモデイ. ニューラル言語モデルのスケールアップ則。 arXiv プレプリント arXiv:2001.08361, 2020.

DP Kingma と J. Ba. Adam: 確率的最適化の手法です。 arXiv プレプリント arXiv:1412.6980, 2014.

J. Li, AH Miller, S. Chopra, M. Ranzato, J. Weston. ヒューマン・イン・ザ・ループによる対話学習。 arXiv プレプリント arXiv:1611.09823, 2016.

コンピュータの制御を学習するためのデータ駆動型アプローチ

- Y. Li, J. He, X. Zhou, Y. Zhang, および J. Baldridge. ping の自然言語命令をモバイル UI アクションシーケンスにマッピングします。arXiv プレプリント arXiv:2005.03776, 2020.
- EZ Liu, K. Guu, P. Pasupat, T. Shi, および P. Liang. ワークフローに沿った探索を使用して、Web インターフェイスでの学習を強化します。arXiv プレプリント arXiv:1802.08802, 2018.
- R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, 他私たちは、人間のフィードバックによるブラウザ支援の質問応答をbgptします。arXiv プレプリント arXiv:2112.09332, 2021.
- DAボメルロー。Alvin: 自律型陸上車両ニューラルネットワークで、神経情報の進歩
- 処理システム、305 ~ 313 ページ、1989 年。
- JW Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, R. Ring, S. Young, 他言語モデルのスケールアップ: Gopher のトレーニングからの方法、分析、洞察。arXiv プレプリント arXiv:2112.11446, 2021.
- T. Shi, A. Karpathy, L. Fan, J. Hernandez, および P. Liang. ビットの世界: Web ベースのエージェント向けのオープン ドメイン プラットフォーム。機械学習に関する国際会議、3135 ~ 3144 ページ。PMLR、2017年。
- M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, および B. Catanzaro. Megatron-LM: モデルの並列処理を使用して、数十億のパラメータ言語モデルをトレーニングします。arXiv プレプリント arXiv:1909.08053, 2019.
- M. Shvo, Z. Hu, R. T. Cartwright, I. Mohamed, A. Jepsen, および S. McIlraith. Appbuddy: 強化学習を介してモバイル アプリでタスクを達成することを学習します。arXiv プレプリント arXiv:2106.00133, 2021.
- D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, 他ディープ ニューラル ネットワークとツリーサーチを使用して囲碁をマスターします。自然、529 (7587) :484, 2016。
- HF ソング, A. アブドルマレキ, JT スプリンベルグ, A. クラーク, H. ソイヤー, JW レイ, S. ヌーリー, A. アフジャ, S. リュー, D. テイルマラ, 他V-mpo: 離散および連続制御のためのオンポリシー最大事後ポリシー最適化。arXiv プレプリント arXiv:1909.12238, 2019.
- N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, および P. Christiano. 人間のフィードバックから要約することを学びます。arXiv プレプリント arXiv:2009.01325, 2020.
- RS サットンと AG バルト。強化学習: An Introduction. MIT Press, 2018年。
- D. Toyama, P. Hamel, A. Gergely, G. Comanici, A. Glaese, Z. Ahmed, T. Jackson, S. Mourad, および D. Precup. droidenv: Android 向けの強化学習プラットフォーム。arXiv プレプリント arXiv:2105.13231, 2021.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, 他 Transformer: 必要なのは注意だけです。神経情報処理システムの進歩、ページ 5998-6008, 2017 年。
- O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, 他マルチエージェント強化学習を使用したスタークラフト II のグランドマスターレベル。ネイチャー、575(7782): 350-354, 2019。
- C. Wirth, R. Akrouf, G. Neumann, J. Furnkranz, 他好みに基づく強化学習法の調査。機械学習研究ジャーナル、18(136): 1-46, 2017。

コンピュータの制御を学習するためのデータ駆動型アプローチ

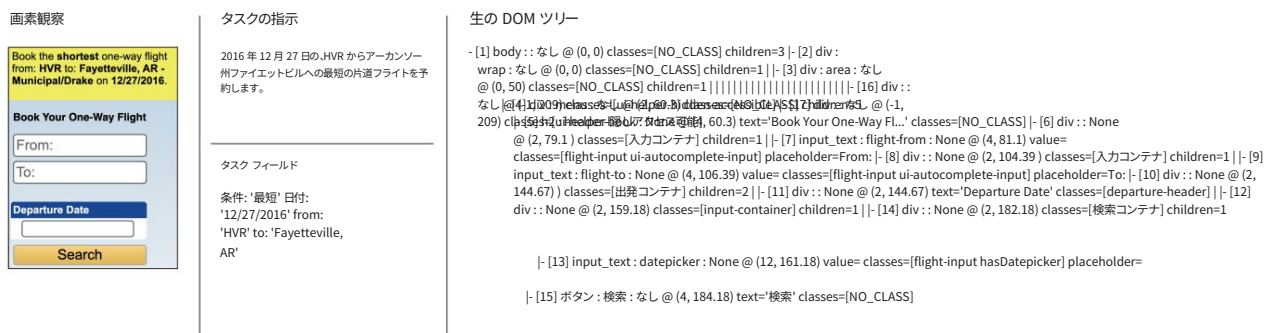


図 9.ブック フライトタスクの MiniWob 観測の例。生の DOM ツリーは、DOM要素のリストに処理されます。このリストに含まれる各 DOM 要素の情報には、整数参照、そのテキスト、任意の入力テキスト、クラス、ラジオ ボタンなどの状態、その位置、フォーカスされているかどうか、エピソードで操作されたかどうかが含まれます。こここのところ。

表 1. トレーニングで使用されるハイパーパラメーター。ソングらを参照してください。(2019) VMPO ハイパーパラメーターの説明。

パラメータ	値Adam
最適化	(Kingma and Ba, 2014) 1st-4
学習率	0.9 0.999 1st-1 1.0 1.0 0.1 0.2
Adam b1パラメータ	0.9 256 64 50
Adam b2パラメータ	
減量 (バイアスを除く)	
VMPO減量	
BC減量 (ベースライン)	
VMPO α	
VMPO σ	
代理店割引 γ	
バッチサイズ	
軌道展開長	
ターゲット ネットワークの更新期間 τ	
エピソードごとの最大ステップ数	300

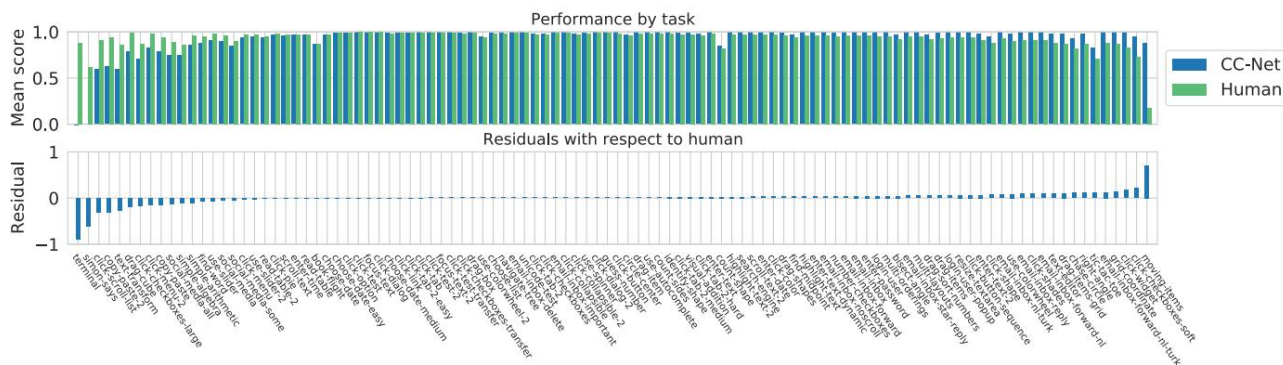


図 10.人間の参加者とのタスクごとのパフォーマンスの比較。図4aのプロットの拡大版。

表 2. キーを押すアクションでエージェントが発行できるキーボード キーとマクロのリスト。

入力	桁6	KeyG	ControlRight+KeyV
ページアップ	桁7	KeyH	Shift+KeyA
ページダウン	桁8	キー I	Shift+KeyB
バックスペース	桁9	KeyJ	ShiftLeft+KeyC
消去	桁0	キーK	Shift+KeyD
タブ	テンキー0	キーL	Shift左+KeyE
空	テンキー1	キーM	ShiftLeft+KeyF
上矢印	テンキー2	キーN	Shift+KeyG
ArrowRight Numpad3		キー O	ShiftLeft+KeyH
ArrowDown Numpad4		キーP	Shift+KeyI
左矢印	テンキー5	KeyQ	ShiftLeft+KeyJ
ブラケット左テンキー 6		キーR	Shift+KeyK
ブラケット右テンキー 7		キーS	Shift左+KeyL
マイナス	テンキー8	キーT	Shift+KeyM
同等	テンキー9	KeyU	Shift+KeyN
セミコロン	テンキー追加	KeyV	ShiftLeft+KeyO
引用	テンキー乗算キーW		Shift+KeyP
バックslash	テンキー減算 KeyX		ShiftLeft+KeyQ
コンマ	テンキー分割キーY		Shift左+KeyR
期間	NumpadDecimal KeyZ		ShiftLeft+KeyS
slash	テンキーEnter	ControlLeft+KeyA	ShiftLeft+KeyT
バッククオート	キーA	ControlRight+KeyA	ShiftLeft+KeyU
桁1	キーB	ControlLeft+KeyC	ShiftLeft+KeyV
桁2	キーC	ControlRight+KeyC	ShiftLeft+KeyW
桁3	キーD	ControlLeft+KeyX	ShiftLeft+KeyX
桁4	キーE	ControlRight+KeyX	ShiftLeft+KeyY
桁5	KeyF	ControlLeft+KeyV	ShiftLeft+KeyZ

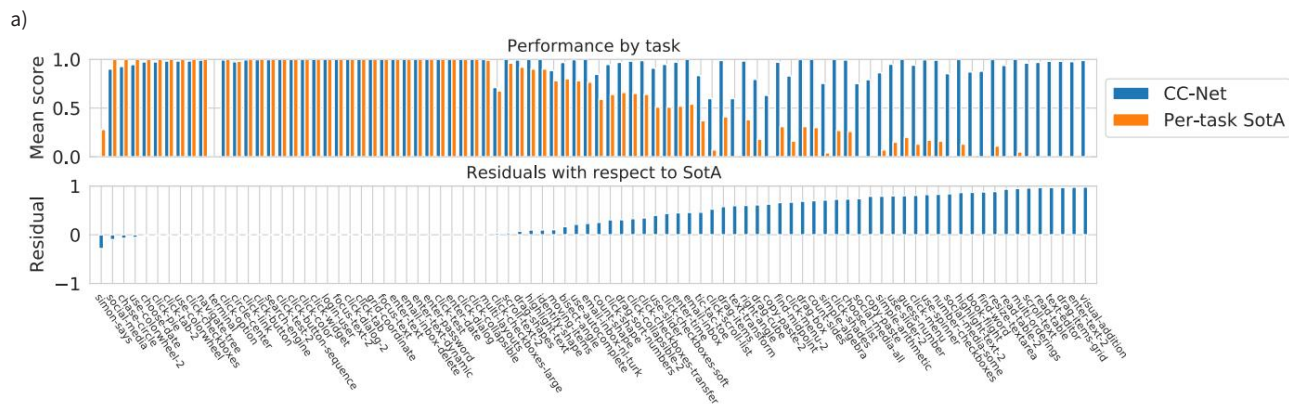


図 11. タスクごとのパフォーマンスと、公開されている各タスクの外部 SotA の最良の結果との比較。これらの SotA スコアは、表3 に示す集計スコアに対応しています。

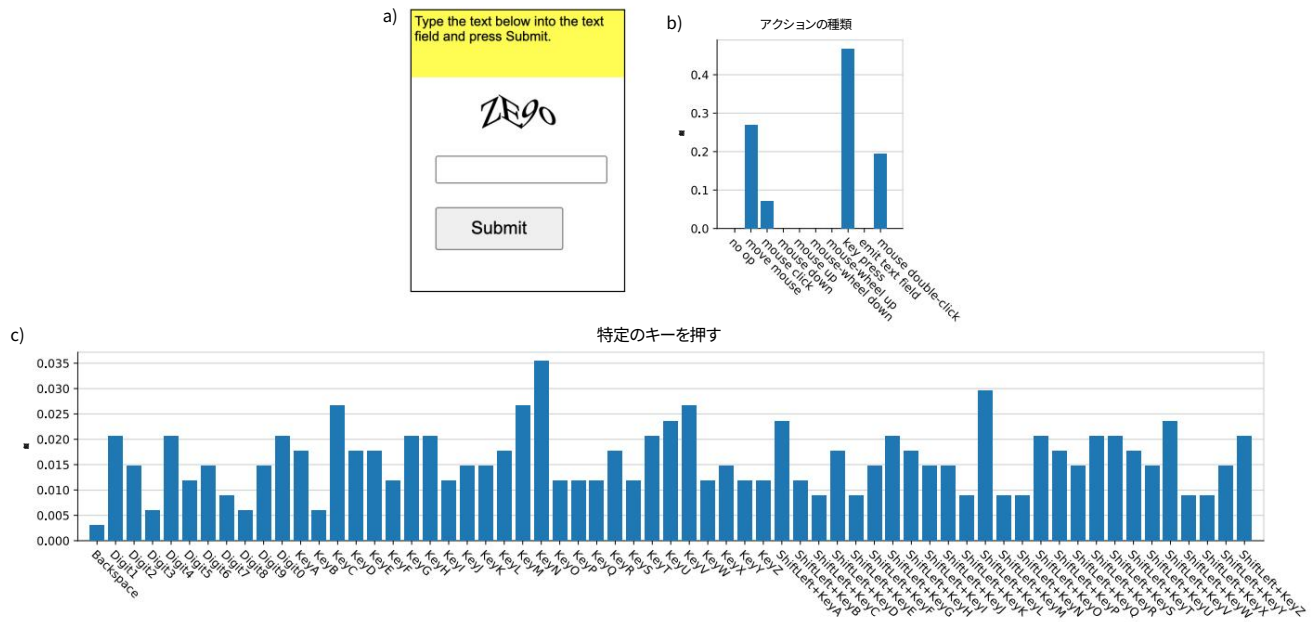


図 12. a) テキスト変換の例のエピソードの最初のフレーム。入力するテキスト (キャプチャ) も DOM に存在するため、言語入力としてエージェントに渡されることに注意してください。 b) テキスト変換の100エピソードのランダム サンプルに対してエージェントによって選択されたアクション タイプの分布。このタスクでは、エージェントはキーの押下を出力することを最も頻繁に選択します。 c) b) に含まれるエピソードの特定のキー押下の選択肢の配布。エージェントは、キャプチャを完了するために、すべての文字 (az、小文字と大文字) およびすべての数字 (0-9) に対して個別のキー押下を生成することを学習します。

コンピュータの制御を学習するためのデータ駆動型アプローチ

表3: 人間の参加者、エージェント (CC-Net)、および外部研究から収集されたスコアのタスクごとの平均スコア(Shi et al., 2017; Liu et al., 2018; Gur et al., 2018; Jia et al., 2019)。これらの研究の一部のスコアは、図形式でしか入手できなかったため、可能な限り正確に推定されました。最後の2つの列は、BCとRL、またはBC、RLと拡張を使用して、各タスクの外部から報告された最高のスコアを示します。

タスク	人間	CCネット	CCネット 紀元前のみ	ビットの世界 (紀元前 &RL)	ワークフロー 誘導探 索 (BC & RL) n/ a 0.00 n/ an/an/a	ナビを学ぶ ウェブのゲ ート (RL)	ドム Qネット (RL)	ワークフロー 誘導探 索 (拡 張) n/a 0.00 n/ an/an/a	学ぶ アビヘ Web の統合 (拡 張) n/a 1.00 n/an/	集計 下 (紀元前と RL)	集計 下 (8月)									
bisect-angle	0.92	0.97	0.29	0.80	0.00 0.16	該当	n/	0.00 0.16	an/an/a	0.80	0.80									
book-flight 追	0.87	0.87	0.00	0.00	n/a 0.99	なし	a/a/	n/a 1.00	0.26 n/an/	0.00	1.00									
跡サークル	0.82	0.93	0.80	1.00	1.00 0.68	0.26	a/a/	1.00 0.84	a 1.00 n/	1.00	1.00									
choose-date-easy	0.99	0.99	0.42	N /	0.51 0.64	該当	a	0.94 0.64	an/an/an/	該当な	該当な									
choose-date-medium	0.98	0.99	0.26	A / A	0.98 0.65	なし	1.00	1.00 0.99	an/a 1.00	し:	し:A									
choose-date choose-	0.97	0.97	0.12	\$ 0.25	1.00 1.00	1.00	a/a/	1.00 1.00	n/an/a	1.00	1,5,26									
list サークルセンター	0.98	0.99	0.19	0.98	1.00 1.00	該当	a	1.00 1.00	1.00 1.00	0.26	0.00									
	0.96	0.97	0.36	0.25	1.00 n/an/	なし	1.00	1.00 n/	n/an/ an/	0.00	1.00									
クリックボタンシーケンス	0.94	1.00	0.47	0.27	a 1.00 0.32	1.00	1.00	an/a 1.00	a 1.00 n/	1.00	1.00									
	0.98	1.00	0.78	0.25	n/a 0.22	1.00	a/a/	0.32 n/a	an/an/an/	1.00	0.00									
クリック-チェックボックス-ラ	0.87	0.71	0.00	0.25	n/ 0.64 an/	該当な	a/a	0.99 n/	an/an/an/	0.00	0.00									
ージ-クリック-チェックボックス	0.73	0.95	0.04	0.13	an/a 0.64	し	1.00	an 0. /an/	a 1.00 n/	0.00	0.00									
ス-ソフト-クリック-チェックボックス	0.98	0.99	0.36	0.83	0.55 1.00	1.00	a/a/	a 0.98	an/an/an/	0.00	0.00									
ストランスファー-クリック-チェ	0.97	0.98	0.32	0.93	n/a 1.00	該当な	a/a/	1.00 1.00	an/an/an/	0.59	0.00									
クボックス-クリック-折りたたみ可	0.97	0.98	0.17	0.00	0.93 n/an/	し/	a	n/a 1.00	an/an/an/	0.59	0.00									
能-2-クリック-折りたたみ可能-クリ	0.99	1.00	0.81	0.00	a 0.59 n/	an/	1.00	0.93 n/	an/an/an/	0.59	0.30									
ック-カラー-クリック-ダイアログ-2	0.97	1.00	0.82	0.99	an/an/an/	an/	1.00	an/a 0.76	an/an/an/	0.301	0.30									
クリック-ダイアログ-クリック-リン	0.99	1.00	0.88	0.99	an/an/an/	an/	a/a/	n/an/an/	an/an/an/	0.18	0.30									
ク-クリック-メニュー-2-クリック-メ	1.00	1.00	0.95	0.98	an/an/an/	an/	a	an/an/	an/an/an/	該当	0.31									
ニュー-クリック-オプション click-	0.99	0.99	0.59	0.8	an/an/an/	an/a	1.00	an/an/	an/an/a	なし	0.18									
pie click-scroll-list click-shades	0.98	0.83	0.52	0.291	a 0.77 n/	1.00	a/a/	an/an/	1.00 1.00	0.01	該当な									
click-shape click-tab-2-easy	0.97	0.94	0.22	0.0	an /an/a	n/an/	a/ an/	an/an/	n/a 1.00 n/	0.41	し									
click-tab-2-hard click-tab-2-	0.99	0.99	0.21	0.600	0.43 0.00	an/	an/	an/a 0.93	an/an/an/	0.92	0.01									
medium click-tab-2 click-tab	0.98	0.97	0.15	0.26	0.99 n/a	an/	an/	n/an/	an/an/an/	0.66	0.41									
click-test-2 click -test-transfer	0.91	0.60	0.01	0.26	1.00 1.00	an/	an/a	0.99 0.96	an/an/an/	1.00	0.92									
click-test click-widget copy-	0.91	1.00	0.04	0.26	0.52 n/an/	an/	1.00	1.00 該当	an/an /a	1.00	0.66									
paste-2 copy-paste count-	0.88	0.95	0.11	0.96	a 1.00 1.00	an/	1.00	なし 1.00	1.00 n/an/	1.00	該当な									
shape count-sides drag-box	0.99	0.99	0.61	N /	1.00 0.00	an/	1.00	1.00 0.90	an/a 1.00	1.00	し									
drag-cube drag-item drag-	0.96	0.98	0.19	A /	n/an/a	an/	n/a	該当なし		0.00	1.00 /									
items-grid drag-items drag-	0.97	0.99	0.54	A / A	0.90 n/a	an/	1.00	1.00 1.00		0.00	an/									
shapes drag-sort-numbers 電子	0.97	0.98	0.27	0.00	0.99 n/a	an/	1.00	1.00 0.00		0.00	an/a									
メール-inbox-delete email-	0.99	1.00	0.95	0.78	0.99 0.05	an/	n/an/	該当なし		0.00	0.93									
inbox-forward-nl-turk email-	0.99	1.00	0.95	N / A	0.99	an/	an/	1.00 該当		0.00	n/an/									
inbox-forward-nl email-inbox-	0.99	1.00	0.94	0.20		an/	an/	なし 1.00		0.00	an/a									
forward email-inbox-important	1.00	1.00	1.00			an/	an/	該当なし		0.99	0.99									
email-inbox-nl-turk email-	0.83	1.00	0.56			an/	an/	1.00 1.00		0.05	1.00									
inbox-noscroll email-inbox-	0.94		0.01			an/	an/	0.99		1.50	1.00									
reply email-inbox -star-reply	0.94		0.04			an /	an/				0.00									
email-inbox enter-date enter-	0.82		0.21			an/	an/				1.00									
password enter-text-2 enter-	0.98		0.74			an/	an/				1.00									
text-dynamic enter-text enter-	0.99		0.61			an/	an/a				0.90									
time find-midpoint find-word	0.99		0.23			an/	1.00				0.31									
focus-text-2 focus-text グリッド	0.98		0.61			an/a	n/an/				0.00									
座標 推測番号 ハイライト-text-2	0.87		0.05			1.00	an/				1.00									
ハイライト テキスト 識別形状 ロ	0.93		0.13			1.00	an/				1.00									
グイン ユーザー ポップアップ ロ	0.96		0.26			n/a	an /				1.00									
グイン ユーザー 移動アイテム マ	0.92		0.11			1.00	an/				0.20									
ルチレイアウト ム-iti-orderings	0.99		0.22			n/an/	an/				0.13									
ナビゲート ツリー	0.88		0.00			an/	an/a				0.90									
	0.91		0.00			an/	0.54				1.00									
	0.96		0.01			an/	n/a				n/a									
	0.99		0.30			an/	1.00				1.00									
	0.93		0.05			an/	n/a				0.78									
	0.96		0.13			an/	1.00				1.00									
	0.91		0.00			an/	1.00				1.00									
	0.95		0.11			an/	n/an/				1.00									
	0.96		0.09			an/	an/a													
	0.97		0.02			an/a	1.00													
	0.96		0.02			1.00	1.00													
	0.91		0.04			n/an/	n/an/													
	0.97		0.39			and/	an/													
	0.98		0.35			または	an/													
	0.98		0.04			1.00	an/													
	0.94		0.35				an/a													
	0.96		0.05				1.00													
	0.99		0.96				n/an/													
	1.00		0.99				an/a													
	0.87		0.66				1.00													
	0.99		0.21																	
	0.97		0.40																	
	0.97		0.51																	
	0.98		0.68																	
	0.94		0.02																	
	0.96		0.00																	
	0.50.9	0.90	0.99	0.90	0.99	0.90	0.99	1.00	0.98	1.00	0.88	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.08	1.9

コンピュータの制御を学習するためのデータ駆動型アプローチ

number-checkboxes read-	0.96	0.99	0.00	0.16	n/	n/a/	n/	n/	n/	0.16	0.16
table-2 read-table resize-	0.95	0.94	0.00	0.00	an/	a/a/	an/	an/	an/	0.00	0.00
textarea	0.97	0.97	0.01	0.00	an/	a/a/	an/	an/	an/	0.00	0.00
	0.94	1.00	0.27	0.11	an/	a/a/	an/	an/	an/	0.11	0.11
直角スクロール	0.87	0.98	0.26	0.38	an/	a/a/	an/	an/	an/	0.38	0.38
テキスト 2 スクロ	0.97	1.00	0.88	0.96	an/	a/a/	an/	an/	an/	0.96	0.96
ールテキスト	0.97	0.96	0.04	0.00	an/	a/a/	an/	an/	an/	0.00	0.00
search-engine	0.97	1.00	0.15	0.00	a 0.26	a/a/	a 1.00	a 0.99	an/	1.00	1.00
simon-says simple-	0.62	-0.00	0.02	0.28	n/an/	a/a/	n/an/	n/an/	an/	0.28	0.28
algebra simple-	0.86	0.75	0.03	0.04	an/a	a/a/	an/	an/a	an/	0.04	0.04
arithmetic social-media-	0.96	0.86	0.38	0.07	0.01	a/a/	an/	0.01	an/	0.07	0.07
all social-media-some	0.89	0.75	0.00	該当な	0.01	a/a	an/a	0.42	a 1.00	0.01	1.00
social-media terminal	0.91	0.85	0.01	し	0.39	およ	1.00	1.00	n/an/	0.01	0.42
text-editor text-transform	0.96	0.90	0.03	0.23	n/an/	び/	n/an/	n/an/	an/	1.00	1.00
tic-tac-toe unicode-test	0.88	-0.01	0.00	0.00	an/a	また	an/	an/a	an/	0.00	0.00
use-autocomplete use-	0.88	0.98	0.11	0.01	0.37		an/	0.47	an/	0.01	0.01
colorwheel-2 use-	0.86	0.60	0.19	0.00	n/a		an/	n/a	an/	0.00	0.00
colorwheel use-slider-2	0.71	0.83	0.32	0.34	0.78		an/	0.98	an/	0.37	0.47
use-slider use-spinner ビ	0.99	1.00	0.86	該当な	n/an/		an/	n/an/	an/	該当な	該当な
ジュアル追加	0.98	1.00	0.07	し	an/		an/	an/	an/	し	し
	0.94	0.95	0.38	0.00	an/a		an/	an/a	an/	0.78	0.98
	0.90	0.98	0.68	1.00	0.04		an/	0.04	an/	1.00	1.00
	0.97	0.95	0.03	1.00	n/ a		an/お	n/ a	an/	1.00	1.00
	0.98	0.91	0.18	0.15			よび/		an/お	0.15	0.15
	0.98	1.90	0.67	0.51			または		よび/	0.51	0.51
	0.98		0.47	0.17 0.01					または	0.17 0.01	0.17 0.01

コンピュータの制御を学習するためのデータ駆動型アプローチ

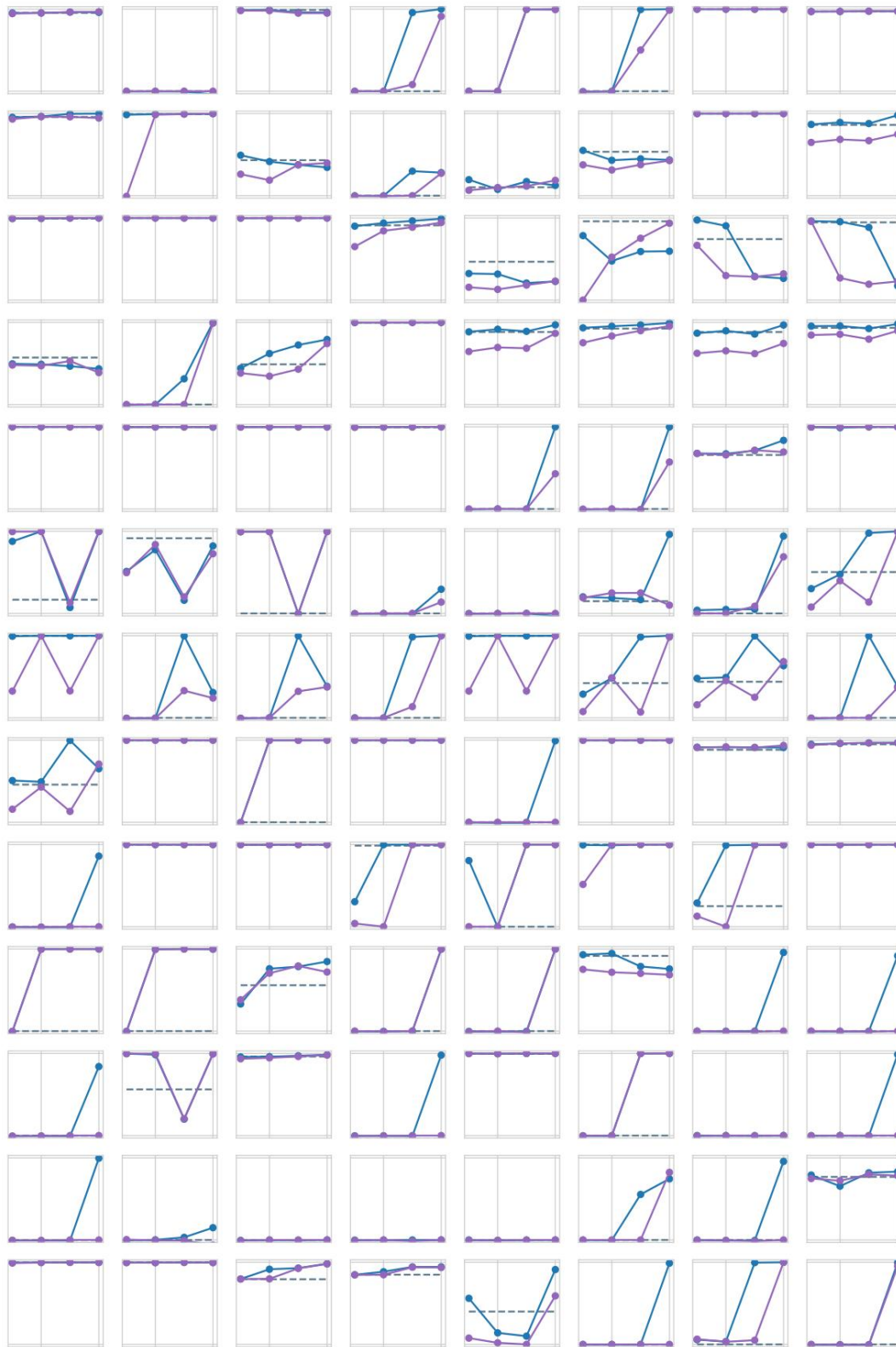


図 13. 図7の結果をタスクごとに分割。実験ごとにシードが1つしかないため、個々のタスクの結果には大きなばらつきがあることに注意してください。青は BC&RL の性能、紫は BC のみ、破線は RL のみの性能を示します。

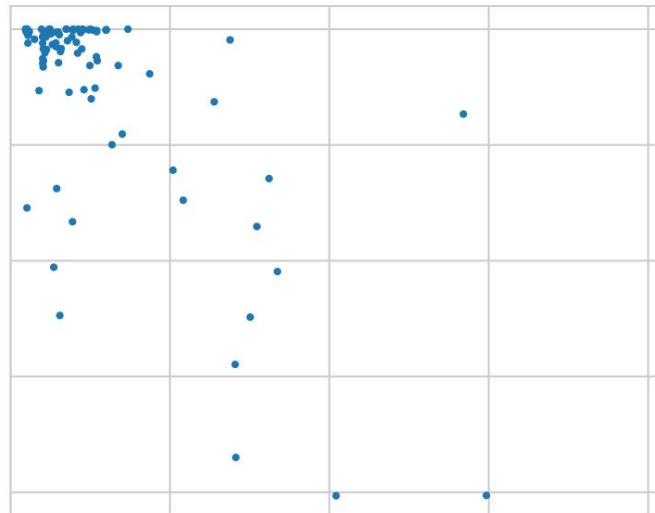


図 14. エピソードごとの平均環境ステップ数とさまざまな MiniWob タスクの平均報酬を示す散布図。