

大きな言語でHTMLを理解する モデル

Izzeddin Gur, Ofir Nachum, Yingjie Miao, Mustafa Safdari, Austin Huang, Aakanksha Chowdhery, Sharan Narang, Noah Fiedel, Aleksandra Faust コム

概要

大規模言語モデル (LLM) は、さまざまな自然言語タスクで卓越したパフォーマンスを示しています。しかし、HTML を理解するための機能 (つまり、Web ページの生の HTML を解析し、Web ベースのタスクの自動化、クローリング、およびブラウザー支援検索へのアプリケーションを使用する) は、十分に調査されていません。私たちは、HTML 理解モデル (微調整された LLM) と、(i) HTML 要素のセマンティック分類、(ii) HTML 入力の記述生成、および (iii) 3 つのタスクの下での機能の詳細な分析に貢献します。

HTML ページの自律的な Web ナビゲーション。以前の研究では、HTML 理解のための専用アーキテクチャとトレーニング手順が開発されましたが、標準的な自然言語コーパスで事前トレーニングされた LLM は、HTML 理解タスクに非常にうまく移行することが示されています。たとえば、微調整された LLM は、タスク データセットで排他的にトレーニングされたモデルと比較して、セマンティック分類で 12% 正確です。さらに、MiniWoB ベンチマークのデータを微調整すると、LLM は、以前の最良の教師ありモデルと比較して、192 分の 1 のデータを使用して 50% 多くのタスクを正常に完了することができます。評価した LLM のうち、T5 ベースのモデルが双方向のエンコーダー/デコーダー アーキテクチャにより理想的であるという証拠を示します。HTML を理解するための LLM に関するさらなる研究を促進するために、CommonCrawl.1 から抽出され、自動ラベル付けされた大規模な HTML データセットを作成してオープンソース化します。

1 はじめに

Web クローリング (Olston et al., 2010)、フォーム入力 (Diaz et al., 2013; Gur et al., 2021)、または情報取得 Web エージェント (Nogueira & Cho, 2016) は、ユーザーの自動化と支援の両方にとって重要です。Web ベースのタスクで。これらのアプリケーションや類似のアプリケーションは、Web ページ上の特定のコンテンツやコントロールを検索したり、Web サイトを自律的にナビゲートしたりできるモデルに依存しています。生の形式の Web ページは HTML ベースのテキスト シーケンスとして表現されるため、Web ベースのタスクのモデルの成功は、HTML のセマンティクス、構造、および埋め込まれた相互作用を理解する能力に依存しています。

Web の自動化と HTML の理解への主なアプローチは、特殊なモデルをトレーニングすることです。つまり、アプリケーション固有のデータセットを収集し、ニューラル ネットワーク (NN) アーキテクチャを設計して、HTML の構造の帰納的パイアスを活用します。たとえば、Liu らを参照してください。 (2018); トヤマら。 (2021); ゴール等。 (2021); ハンフリーズ等。 (2022)。ただし、データセットの収集とニューラル アーキテクチャの設計はどちらも費用と時間がかかり、高度に専門化されたドメイン固有の知識が必要です。

一方、自然言語処理 (NLP) の文献では、大規模言語モデル (LLM) が、データセットの収集と特殊な NN 設計の難しさに対する解決策として登場しました (Kaplan et al., 2020; Bommasani et al., 2021)。NLP で一般的なパラダイムは、教師なしでタスクに依存しない学習目標を介して大規模なテキスト コーパスで事前トレーニングされた既製の LLM を使用し、小さなタスク固有のデータセットで LLM を微調整またはプロンプトすることです。このパラダイムは、さまざまな NLP タスクで卓越したパフォーマンスを示しています (Xue et al., 2020; Brown et al., 2020; Austin et al., 2021)。LLM を HTML の理解に適用できるかどうか (特に、コンテキストとシーケンスの長さがはるかに大きいことを考えると) は、未調査の問題のままです。

¹<https://sites.google.com/view/llm4html/home> で結果の視覚化を参照してください。

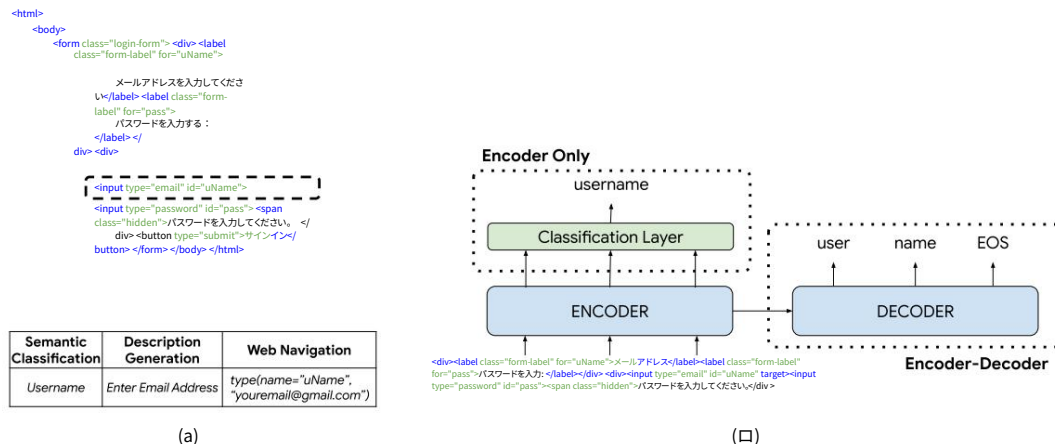


図 1: a) 重要な要素である重要な要素 (破線のボックス) が強調表示された HTML の例のページ。すべての正規タスクは、一連のカテゴリの 1 つとして分類するか、その目的のテキスト説明を生成するか、複数ページの Web サイトのシーケンシャルナビゲーションの一部としてアクションを適用することにより、この要素との明確な相互作用を評価します。b) LLM アーキテクチャの概要。破線のボックスは、エンコーダーのみまたはエンコーダー/デコーダー モデルのいずれかに固有のサブモジュールを示します。エンコーダーのみのモデルの場合、追加の分類レイヤーを追加します。デコーダーのみのモデル (図にはありません) は、エンコーダー/デコーダー モデルに似ています。主な違いは、HTML スニペットがデコーダーに供給され、左から右に処理されることです。

このホワイトペーパーでは、カスタム NN アーキテクチャ設計を必要とせずに、LLM を HTML 理解に適用して、より優れたパフォーマンスとサンプル効率の高い HTML 理解モデルを作成できるかどうかを調査します。そのために、これらのアプリケーションの本質を捉え、構造とコンテンツの両方を理解する必要がある HTML を理解するための 3 つのベンチマークタスクのスイートを提示します。まず、セマンティック分類を、特定の HTML 要素をアドレス、電子メール、パスワードなどの一連のカテゴリの 1 つに分類するモデルを必要とするタスクとして考案し、自動フォーム入力に適用します。次に、モデルに HTML スニペットが与えられ、自然言語の説明を生成するように求められるラベル抽出タスクである説明の生成を提示します。たとえば、メールフィールドの場合、説明は「メールアドレスを入力してください」のようになります。大部分の Web ページでは、入力要素と説明コンテンツの間のこの接続は生の HTML コードでのみ暗黙的に示され、そのようなリンクを推測することは、より高いレベルのナビゲーション目的の前提条件であることに注意してください。3 つ目のタスクは、Autonomous Web Navigation (Shi et al., 2017) です。モデルには、自然言語コマンドとペアになった HTML ページが表示され、コマンドを満たすために一連の HTML ページに適切なアクションを適用する必要があります。これらのタスクの簡単な例については、図 1a を参照してください。

これらのベンチマークタスクを使用して、アーキテクチャ (エンコーダーのみ、エンコーダー/デコーダー、またはデコーダーのみ)、モデルサイズ (24.6M から 62B パラメーター) が異なる、さまざまな事前トレーニング済み LLM (表 1) の転送機能の評価を行います。)、およびトレーニングデータコーパス (トレーニング前の NLP および HTML コーパスを含む、または除外する)。以前の研究では、モデルへの入力として HTML を普遍的に事前解析していましたが (Gur et al., 2021; Liu et al., 2018; Nakano et al., 2021)、私たちの研究は、私たちの知る限り、最初の研究です。生の未処理の HTML を使用します。私たちの結果は、LLM がすべてのタスクで HTML を非常によく理解していることを示しており、ゼロからトレーニングされたモデルよりもサンプル効率が最大 192 倍高く、MiniWoB ベンチマークスイートで教師あり学習の新しい SoTA を達成していることを示しています (Shi et al., 2017)。双方向のアテンションを備えたエンコーダー/デコーダーアーキテクチャは、事前トレーニングに HTML が含まれていない場合でも、全体的に最高のパフォーマンスを示します。さらに、パフォーマンスがモデルのサイズに応じてサブリニアにスケールアップすることを示します。

要約すると、私たちの論文は、HTML の理解と Web 自動化のための機械学習に関する新しい視点を提供し、事前トレーニング済みの LLM がそのようなタスクで大きなパフォーマンスを達成できることを示し、特殊なアーキテクチャとトレーニングプロトコルの必要性を回避します。この方向でのさらなる研究を促進するために、記述生成用のデータセットをオープンソース化する予定です。

2 関連作品

HTML の理解自律的な Web ナビゲーションは、ニューラル ネットワーク モデルの一般的なアプリケーションであり、さまざまな研究で、Web ベースのエージェントをトレーニングするためのシミュレートされた Web サイトが提案されており、タスクの遂行に適用されます (Yao et al., 2022; Gur et al., 2021; Burns et al., 2022; Mazumder & Riva, 2020; Shi et al., 2017; Liu et al., 2018) および情報検索または質問応答 (Adolphs et al., 2021; Nogueira & Cho, 2016))。シミュレートされた Web サイトは、モデルをオンラインで評価する簡単な方法を提供します。このため、Web ナビゲーション設定には既存の MiniWoB ベンチマーク (Shi et al., 2017) を使用します。ただし、さまざまな実世界の Web サイトでモデルを評価するためのメカニズムを持つことは依然として重要です。これが、説明生成タスク用の独自のデータセットを生成する主な動機でした。これは、CommonCrawl から抽出されて自動ラベル付けされ、私たちの論文の重要な貢献です。

これらのベンチマークに加えて、多くの研究が Web ナビゲーションと関連するサブタスクのモデルを開発しました (Pasupat et al., 2018; Bommasani et al., 2021; He et al., 2021; Humphreys et al., 2022; Liu et al., 2018; Jia et al., 2019)。これらの作業は、多くの場合、HTML 構造の帰納的バイアスを利用する特殊なニューラル ネットワーク アーキテクチャや、モデルへの入力を容易にする HTML の前処理に依存しています。対照的に、私たちの仕事は最小限のアプローチを取り、生のテキスト形式で HTML を提供し、広く採用されている変換ネットワークを使用します。

LLM と HTML LLM と HTML の共通点を探求する作品は、一般に 2 つのカテゴリに分類されます。最初のカテゴリは、LLM を使用して Web ナビゲーションを支援し (Nakano et al., 2021; Yao et al., 2022)、通常はカスタムの前処理に依存して、Web ページのコンテキストと構造を自然言語にマッピングします。モデルが解析できる HTML ページ。2 番目のカテゴリは、HTML テキストの大規模なコーパスで LLM を事前トレーニングします (Aghajanyan et al., 2021)。ただし、これらの作業は通常、モデル評価を標準の NLP タスク (要約や質問/応答など) に制限し、HTML の理解や Web 自動化に関連するタスクとは対照的です。

私たちの仕事は逆に考えることができます: LLM の事前トレーニングを変更せずに維持し、事前トレーニングされた LLM を HTML 関連のタスクに転送するメカニズムに焦点を当てています。

3 半構造化テキストデータとしての HTML に関する簡単な背景

HTML は、Web ページの構造とコンテンツを整理するために使用されるマークアップ言語です。図 1a の HTML ページの例を考えてみましょう。この Web ページには、隣接する 2 つの入力要素 (1 つは電子メール用、もう 1 つはパスワード用) が含まれており、ページの別の分岐に対応するラベルが付いています。

これらの入力とラベルは、HTML ビルディング ブロックとして機能する多くの可能な要素の 1 つです。

各要素には、スタイルや人間が読めるテキストなど、要素の内容を説明する一連の属性 (キーと値のペア) があります。ブラウザでレンダリングされる場合、これらの属性は、要素の表示方法と配置場所を担当します。図 1a の例では、最初の入力に `tag="input", type="email", id="uName"` の 3 つの属性があり、要素を識別子 (「uName」) を持つ電子メール入力として識別します。プログラムでアクセスできます。

HTML を理解するための 4 つの標準タスク

HTML の理解能力を研究するために、3 つの標準的なタスクを考案します。

自律的な Web ナビゲーション。このタスクでは、モデルが複数ページの Web サイトをどの程度うまくナビゲートするかを、一連の意思決定の問題として評価します (Shi et al., 2017; Liu et al., 2018)。エピソードの開始時に、エージェントには自然言語による指示が与えられます。たとえば、ユーザー名「lyda」とパスワード「N22t」をテキスト フィールドに入力し、ログインを押します。エージェントは一連の HTML ページにアクションを適用します。各アクションは関数 (セレクター、テキスト) の形式です。関数はクリックまたはタイプ of のいずれかであり、セレクターは要素を一意に識別する整数ポインターであり、テキストはタイプ機能がアクティブ化されている場合に入力するテキストです。エピソードは、ページが最終状態に達するか (「サインイン」ボタンがクリックされるなど)、または最大ステップ数に達すると終了します。

セマンティック分類。多くの HTML 理解アプリケーションでは、HTML 要素を標準化されたカテゴリに分類できるモデルが必要です。たとえば、自動化されたフォーム入力 (Diaz et al., 2013; Gur et al., 2021) では、多くの Web サイト (たとえば、ショッピング、

タスク	データセット	サイズ	モデル		
			入力	アーキテクチャの出力	タスクの出力
自律的な Web ナビゲーション	MiniWoB デモ (Shi et al., 2017)	12K	ページ	En-Dec 12月	文章 辞書
意味分類	注釈付きショッピング ウェブページ (Gur et al., 2021) 28K スニペット			全て	文章 カテゴリー
説明 世代	CommonCrawl (新規)	85K スニペット		En-Dec 12月	文章 文章

表 1:タスク、データセット、およびモデルの概要。すべてのモデルは生の HTML を受け取ります。Autonomous Web Navigation は HTML 全体を受け取りますが、他のタスクは特定の顕著な要素から抽出された HTML スニペットを受け取ります。

フライト予約、ユーティリティ アプリケーションなどをさまざまなボタン表現 (位置、色、テキストなど) で表示します。したがって、要素を役割カテゴリに分類することとして意味分類を定式化します。2つの入力要素と送信ボタンを含む図 1a の HTML の例を見てみましょう。最初の入力を、システムによって分類される対象の要素 (顕著な要素とも呼ばれます) として選びましょう。この要素はログイン ページに表示され、通常はフォーム入力アプリケーションでユーザー名に関連付けられる電子メール アドレスのラベルがあるため、システムはこの要素をユーザー名として分類する必要があります。これを解決するために、システムはページ内の複数のソースから情報を集約できます。電子メール アドレスを入力してくださいというラベル、入力属性 (type="email" および id="uName")、または「パスワード」や「サインイン」などのページ。このタスクを適切に解決するには、モデルがページの構造とコンテンツの理解を構築し、この理解を使用して正確な予測を行う必要があります。

説明 世代。アクセシビリティ志向の Web ブラウザー制御 (Jorgensen & Binsted, 2005) のアプリケーションに動機付けられて、HTML 内の要素のテキスト記述を見つけて出力として生成することを目標とする抽出問題として記述生成を定式化します。たとえば、図 1a の顕著な要素の説明は、電子メール アドレスの入力です。レンダリングされると、このラベルは「メール」入力フィールドの上に表示されます。HTML は大きな柔軟性を提供するため、一般に、レンダリング時に特定の要素の横に表示される説明テキストは、HTML プレーンテキストを見ると、その要素から非常に離れている可能性があります。したがって、このタスクは、レンダリングされた Web ページに直接アクセスできないにもかかわらず、ユーザーに表示される HTML の構造を理解するモデルの能力を評価します。

5つのデータセット

各標準タスクには個別のデータセットが必要です。説明生成タスクでは、CommonCrawl に基づいて新しく提供され、自動ラベル付けされたデータセットを使用します。

自律的な Web ナビゲーション。公開されている MiniWoB ベンチマーク (Shi et al., 2017) に含まれる 12K のデモを使用します。これには、電子メール転送からソーシャル メディア インタラクションまで、62 の Web サイト アプリケーションが含まれます。各デモンストレーションは、(命令、HTML、アクション)タプルのシーケンスです。MiniWoB デモンストレーションのすべての要素には、それぞれのページ内で一意の参照番号が付いています。この数は要素セレクターとして使用でき、すべてのタスクとタイム ステップでアクション スペースを統一します。たとえば、図 1a のアクションは type(ref=5, "username@email.com") になります。ここで、5 は上から下に数えたときの入力のインデックスを指します。モデル入力として、自然言語命令と HTML を単一のテキスト入力シーケンスに連結します。同様に、モデルが予測するテキスト シーケンスとしてアクションを扱います。

セマンティック分類。環境生成のコンテキストで以前に使用された形式 (HTML、要素、カテゴリ) の 66 の異なるカテゴリを含む 28K のラベル付きの例のデータセットを使用します (Gur et al., 2021)。このデータセットは、実際のショッピング Web サイトの HTML ページと、これらの Web サイトでの支払いおよびチェックアウト時のフォーム入力に関連するカテゴリで構成されています。

説明 世代。このタスクでは、CommonCrawl からデータセットを取得します。2 CommonCrawl には、HTML 内のどのテキストがどの要素に関連付けられているかを明らかにするレンダリングや注釈は含まれていません。代わりに、for と呼ばれる HTML スキーマの特別な属性を利用して、さまざまな要素の説明を推測します。図 1a の例のように、HTML の最初のラベルには、値 uName を持つ for 属性があります。これは、ラベルによって記述された要素の ID です。この場合、id はページの最初の入力の ID です。この注釈は影響しません

ページのレンダリングであり、通常はアクセシビリティの目的で使用されます。これらによって与えられる属性情報を利用して大規模なデータセットを作成し、記述生成を研究します。補足資料で小さなサンプルを利用できますが、データセット全体は公開時に利用できません。

具体的には、CommonCrawl プロジェクトから 100 個の WARC ファイル (2019 年 4 月以降) を収集し、for 属性を持つすべての HTML ラベルを抽出しました。HTML ラベルから非 Unicode および英数字のテキストを削除すると、サンプル データセットが 400K になります。ラベルの分布のバランスを取り、データセットを効果的に 85K サンプルにダウンサンプリングします。各例は (HTML、要素、説明) として表されます。ここで、HTML はページの HTML プレーンテキスト、要素はラベルの for 属性に表示されるものと一致する id 属性を持つ要素、説明はラベル要素内のテキストです (図 1a の例)。データセットの詳細については、付録 A.1 を参照してください。

6 前処理

HTML をトークン シーケンスとして扱うことで、モデル入力前の HTML ツリーの前処理を最小限に抑えます。したがって、HTML を生のテキスト (つまり、テキスト トークンのシーケンス) として提供し、典型的な LLM コンテキスト ウィンドウに収まらないほど大きすぎるページに対してのみ、スニペット抽出の前処理を適用します。

スニペット抽出。実際の HTML ページは非常に大きくなり、数千の要素に達し、調査した最大の LLM のコンテキスト ウィンドウ (PaLM で 1920 トークン (Chowdhery et al., 2022)) をはるかに超えます。LLM は通常、このような長いシーケンスを切り捨てます。これは、HTML ページが直線的に構造化されていないため、HTML の理解に悪影響を与える可能性があります。要素中心のアプローチを採用し、顕著な要素を囲む HTML スニペット (HTML コードのごく一部) を抽出します (図 5)。ツリーの幅と深さを制御する単純なヒューリスティックがプロセスをガイドします。顕著な要素から開始し、停止条件が満たされるまで、HTML ツリー内のその祖先をトラバースします。上にたどると、ツリーの高さと新しいルートの子孫の数の増加を推定します。いずれかのメトリックが事前定義された制限に違反すると停止し、結果のサブツリーをスニペットとして取得します。他の要素と区別するために、target と呼ばれる特別な属性を使用して顕著な要素をマークします。セマンティック分類および記述生成データセットのスニペット抽出を実行し、完全な HTML ページを MiniWoB に保持します。これらのページは通常、実際の HTML よりもはるかに小さいためです。

HTML の非解析。モデルには、一連のトークンの形式で解析されていない平文の HTML を提供します。この正規表現は、階層ネットワークなどの特定のモデル アーキテクチャを必要とせず (Liu et al., 2018; Gur et al., 2021)、任意の LLM にフィードできます。すべての HTML ページまたはスニペットをシーケンスに変換することで、すべてのデータセットを変換します。MiniWoB の場合、さらに (アクション履歴、命令、HTML) タプルを 1 つのシーケンスに連結します。

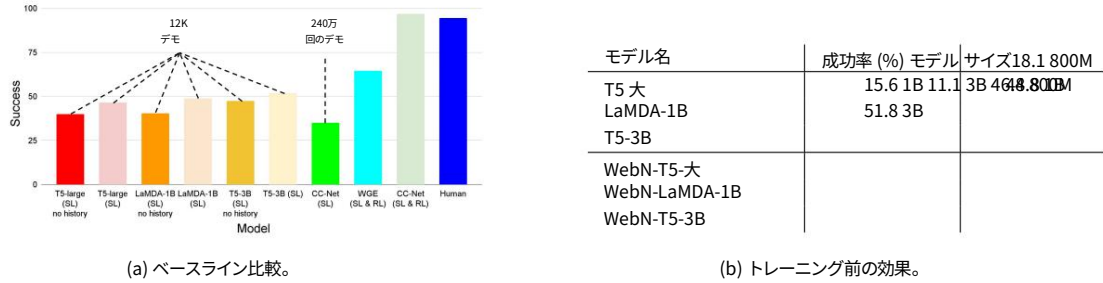
7 モデルトレーニング

私たちは、HTML 理解タスク用のさまざまなサイズとアーキテクチャを持つさまざまな変換ベースの LLM (Vaswani et al., 2017) を研究しています (表 1)。本文の残りの部分では、自律的な Web ナビゲーション、説明生成、およびセマンティック分類用に微調整されたモデルに、それぞれ WebN-、WebD-、および WebC- というプレフィックスを付けます。たとえば、WebD-T5-3B は、記述生成タスク用に微調整された 30 億パラメータの T5 モデル (Raffel et al., 2020) です。このセクションの残りの部分では、トレーニングの詳細について詳しく説明します。

エンコーダー/デコーダーおよびデコーダーのみのモデル。エンコーダー/デコーダー モデル、つまり T5 (Raffel et al., 2020)、およびデコーダーのみのモデル、つまり LaMDA (Thoppilan et al., 2022) および PaLM (Chowdhery et al., 2022) をテキストでトレーニングします。入力とテキスト出力 (図 1b)。入力は生の HTML ページまたはスニペット テキストです。同様に、出力はカテゴリ、自然言語の説明、またはテキストとして表されるアクションです。

つまり、セマンティック分類では、以前の NLP の分類問題と同様に、カテゴリのテキスト表現を使用します (Raffel et al., 2020)。Autonomous Web Navigation の場合、アクションは、最初にキーと値のペアに変換し、次にペアを連結することによってテキストに変換されます。

MiniWoB の多くの Web サイトでは、クリック ボタン シーケンスやチェックボックスのクリックなど、複数のインタラクションが必要であり、各インタラクションによって Web サイトの状態が微妙に変化する可能性があります。例えば、



(a) ベースライン比較。

(b) トレーニング前の効果。

図 2: a) MiniWoB ベンチマークでの以前の SOTA モデルと比較した WebN-T5* パフォーマンス。

WebN-T5-3B は、最高の教師あり学習 (SL) モデルである CC-Net (SL) と比較して、192 分の 1 のデータを使用しながら、タスクの成功率を 16% 向上させます。LLM のパフォーマンスは、RL を利用した作品によってのみ凌駕されており、ウェブサイトとのオンラインでのやり取りをはるかに多く必要とします。b) Autonomous Web Navigation タスクの事前トレーニングを行った場合と行わない場合の LLM。事前トレーニング (「WebN-」接頭辞で示される) を使用すると、パフォーマンスが 2.5 ~ 4.5 倍向上します。

click-checkboxes Web サイトのチェックボックスをクリックすると、その値が正から負に、またはその逆に反転します。これは、LLM の予測に常に反映されるとは限らず、アクションの繰り返しにつながります。過去の一連のアクション(アクション履歴、命令、HTML、アクション) でデータセット内のタブを拡張し、LLM が過去の実験から学習できるようにすることで、この問題を解決します。

エンコーダのみのモデル。テキスト入力とカテゴリ出力を使用して、エンコーダのみのモデル、つまり BERT (Devlin et al., 2018) をトレーニングします。セマンティック カテゴリは、個別のワンホット クラスとして保持します。エンコーダのみのモデルをトレーニングするには、最終的なエンコーダー レイヤーの後に新しい分類レイヤーを追加して、セマンティック カテゴリの分布を生成します。典型的な BERT モデルに加えて、逆ボトルネックを持つ BERT-large から抽出された Mobile BERT (Sun et al., 2020) と、パラメータ共有と埋め込み分割を持つ Albert-XL (Lan et al., 2020) を研究します。

8件の結果

ここで、HTML を理解するために微調整された LLM の結果を提示します。モデルのパフォーマンスを、可能な場合は既存のベースライン (自律的な Web ナビゲーション) と比較し、他の LLM アーキテクチャおよびトレーニング体制 (すべてのタスク) と比較します。セクション 8.1、8.2、および 8.3 ではタスク固有のパフォーマンスを評価し、セクション 8.4 ではすべてのタスクにわたるパフォーマンスを評価します。

メトリクス: 自律的な Web ナビゲーションの場合、モデルの成功率を評価します。これは、タスクあたり 100 エピソード以上の平均です。他のタスクでは、Accuracy を使用して、予測とグラウンド トゥールズの正確な一致を測定します。説明生成タスクでは、代替の「ソフト」テキスト評価メトリックである BLEU と ROUGE-1 を使用して評価を追加し、予測テキストとグラウンド トゥールズ テキストの類似性を測定します。

8.1 自動 Web ナビゲーションの結果

Autonomous Web Navigation では、人間が注釈を付けた実際の Web サイトの 12k デモンストレーションで、2 つの WebN-encoder-decoder アーキテクチャ (WebN T5-large および WebN-T5-3B) を微調整します。MiniWoB (Liu et al., 2018) ベンチマークでモデルを評価し、240 万人の人間の専門家によるデモンストレーション CC-Net (SL) (Humphreys et al., 2022) で教師あり学習 (SL) を使用してトレーニングされた特殊なアーキテクチャと比較します。、および SL でブートストラップされた 2 つの RL モデル、CC-Net (SL) (CC-Net (SL & RL) (Humphreys et al., 2022)、および WGE (SL & RL) (Liu et al., 2018))。さらに、デコーダーのみのアーキテクチャ (WebN-Lambda-1B) と比較し、入力にアクション履歴を含めることの影響についてアブレーション研究を実行します。

SoTA との比較。以前の作業では、Mini WoB の Web サイトのサブセットのみで成功が報告されているため、すべてのモデルに共通する 62 の Web サイトのうち 48 について評価します。付録の表 8 は詳細な結果を報告し、図 2a はすべての Web サイトの平均結果を示しています。

240 万回のデモンストレーションすべてでトレーニングされた CC-Net (SL) と比較して、WebN-T5-3B は成功率を 16% 向上させ、公開されている 12K のデモンストレーションのみでトレーニングし、サンプル効率を 192 倍以上向上させました。LLM のすべての選択肢が、以前の SL モデルよりも優れていることがわかりました。

モデル名	テスト(%)	開発(%)	モデル サイズ	トレーニング中のコード コーパス	77.7	24.6	M 58.9
WebC-MobileBERT	78.1		M 83.1 83	6 38.7	M 85.2	50.8	M 29.0
WebC-アルバート-XL	83.5				8.		
WebC-BERT-最小	84.4						
WebC-BERT-小	84.4						
WebC-BERT-中	85.2						
WebC-BERTベース	83.9						0%
WebC-BERT-大	84.1						
WebC-T5ベース	86.8						
WebC-T5-大	87.0						
WebC-T5-3B	87.7						
WebC-LaMDA-1B	87.4				1 B 12.5%	コード 8 B 5%	コード
WebC-PaLM-8B	86.6				(0.875% HTML)		
WebC-PaLM-62B	88.7				62 B 5%	コード (0.875% HTML)	
T5-ラージ	75.2	800 M	73.8	3 B	0%	75.4	PaLM-8B 70.1
	77.2						
	73.3						

表 2:セマンティック分類タスクでの LLM のパフォーマンス。市販の事前トレーニング済み LLM (プレフィックス「Web*」が付いたモデル名) を微調整すると、HTML データセット (プレフィックス「Web*」のないモデル名) で同じアーキテクチャをゼロからトレーニングする場合と比較して、LLM の転送が改善され、精度が向上します。PaLM-8B の 12% 以上。WebC-PaLM-62B は明らかに他のすべてのモデルよりもパフォーマンスが優れていましたが、WebC-T5-large は WebC-LaMDA-1B や WebC-PaLM-8B などのはるかに大きなモデルと競合することがわかりました。

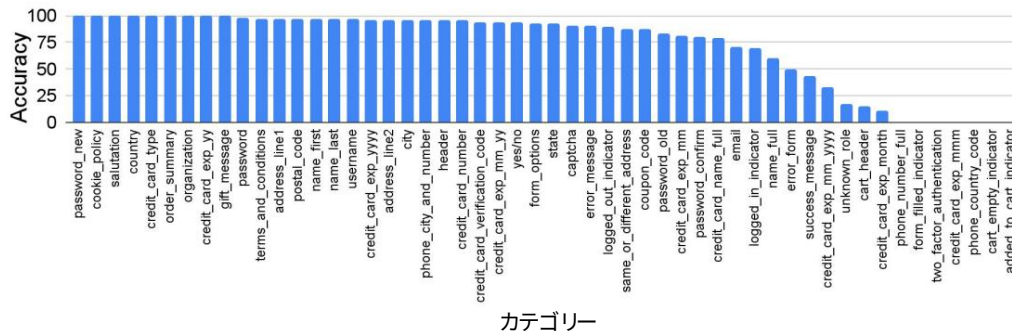


図 3:開発データセットの WebC-T5-3B モデルの分類カテゴリごとの精度。

特に、WebN-T5-3B は、チェックボックスのクリックなどの複数のアクション シーケンスが必要な Web サイトや、ログイン ユーザーなどのテキスト入力が必要な Web サイトで大幅に改善されています (表 8)。LLM のパフォーマンスは、RL を利用した以前の作品によってのみ凌駕されていることがわかります。これは、桁違いに多くのオンライン経験の相互作用を必要とします。微調整された LLM を RL 設定に拡張することは、将来の作業の有望な手段です。

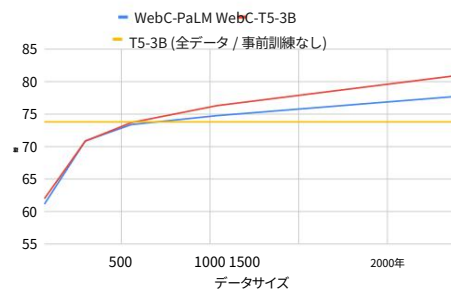
行動履歴アブレーション。すべての LLM で、過去のアクションを入力から除外すると、平均 6.4% の成功率の低下が一貫して観察されます (図 2a)。アクション履歴は、複数のテキストを入力する必要がある Web サイトや、検出が困難な小さな変更 (チェックボックスのクリックや複数のレイアウトなど) を理解するのに役立ちます。マルチ レイアウトでは、Web サイトに 3 つの異なるテキストを入力する必要があり、各エピソードでレイアウトがランダム化されますが、驚くべきことに、行動履歴のない (比較的小さい) WebN-T5-large モデルでさえ、CC-Net (SL) モデルよりも優れています。行動履歴を組み込むことが、より良い成功に貢献する唯一の要因ではないことを示しています。

8.2 意味分類タスクの結果

セマンティック分類タスクを評価するために、T5 エンコーダー/デコーダー アーキテクチャの 3 つのサイズのバリエーション (WebC-T5-base、WebC-T5-large、および WebC-T5-3B) を比較し、22K の実際の人間がラベル付けたトレーニング Web サイトで微調整します。微調整されたエンコーダーのみのアーキテクチャ (WebC-*BERT*)、3 つの微調整されたデコーダーのみのアーキテクチャ (WebC-LaMDA および PaLM)、および人間がラベル付けた Web サイトでトレーニングされたエンコーダー/デコーダー およびデコーダーのみのモデルの両方と比較します。傷。結果

新しい 子孫(%)	高さテスト(%)偏差(%)		
	25	50	
50	3	87.7	90.3
	4	88.6	89.2
	3	88.4	90.0
	4	89.3	89.2
300	5	87.8	88.8
500	7	75.8	74.5

(a)



(ロ)

図 4: a) WebC-T5-3B に対するスニペット抽出パラメーターの影響。新しい子孫と 4 の高さが 50% を超えて増加します。両方のパラメーターが大幅に増加すると、スニペットが大きくなり、精度が低下します。 b)

トレーニング データ サイズに対する精度。1000 のラベル付きサンプル (すべてのトレーニング データセットの 4.4%) のみを使用すると、WebC-T5-3B は、利用可能なすべてのラベル付きデータを使用してトレーニングされた T5-3B (完全なデータ / 事前トレーニングなし) よりも優れており、回数である WebC-PaLM-8B よりも優れています。大きさの大きい。

モデル名	Test Dev Accuracy(%) BLEU ROUGE-1 Accuracy(%)									
	BLEU	ROUGE-1	94.1	87.4	92.3	83.2	94.7	87.8	35.8	17.1
WebD-T5-大	90.1					94.7			76.9	87
WebD-LaMDA-1B	89.7					94.6			77.5	86.5
WebD-T5-3B	90.8					95			78.1	87.5
最も近い説明	61.2					62.7			56.8	58.4

表 3: LLM の記述生成精度。

すべての WebC-LLM のパフォーマンスが良好で、事前トレーニングなしの同じアーキテクチャよりも大幅に優れていることがわかります。

カテゴリごとの精度。図 3 に、開発データセットでの WebC-T5-3B モデルの精度分布を示します。微調整されたエンコーダー/デコーダー モデルは、ほとんどのカテゴリ (図 3) で、サンプルが非常に少ないカテゴリでも強力に機能します。たとえば、モデルはパスワード new で 100% 正確です。これは、クラスが明確であるため、56 のトレーニング例しかありません。

一方、当然のことながら、ユーザー名と間違われることが多い電子メール カテゴリなど、カテゴリがあいまいな場合はパフォーマンスが低下します。

スニペット世代のアブレーション。スニペットの生成を制御する 2 つのハイパーパラメーター: 新しい子孫の割合と新しいレートの高さ。両方のパラメーターを少し変更してもパフォーマンスは変わりませんが、両方を増やすとパフォーマンスが大幅に低下します (表 4a)。新しい子孫が最大 500%、高さが最大 7 の場合、パフォーマンスは 15% 以上低下します。両方のパラメーターが無期限に増加すると、スニペット生成はフルページ HTML を返すことに注意してください。

データサイズへの影響。図 4b で微調整トレーニング データ サイズ (クラスごとに 1, 5, 10, 20, または 50 サンプル) を変化させると、WebC-T5-3B は、1 桁大きい WebC-PaLM-8B よりもわずかに優れています。事前トレーニングなしで利用可能なすべての HTML データでトレーニングされた T5-3B と比較して、WebC-T5-3B は、ラベル付けされたデータ (1000 サンプル) の 3.4% のみを使用しながら、より優れたパフォーマンスを達成します。HTML を理解するための LLM。

8.3 記述生成タスクの結果

説明の生成では、URL トップレベル ドメインに基づいて CommonCrawl データセットを分割し、目に見えない HTML に一般化する LLM の機能をテストします。エンコーダー/デコーダー アーキテクチャ (WebD-T5*) とデコーダーのみのモデル (WebD-LaMDA*) を微調整し、結果を表 3 に示します。また、顕著な要素に最も近い説明を単純に見つける強力なヒューリスティック ベースラインも評価します。HTML テキスト (最も近い説明)。

精度と類似性のパフォーマンス評価の結果を表 3 に示します。すべての事前トレーニング済みモデルは、すべてのメトリックで高いスコアを達成し、次の点で精度で $\approx 90\%$ を達成します。

BLEU と ROUGE-1 に基づく完全一致スコアとわずかに高い非完全一致スコア ($\approx 94\%$)。この違いは、モデルが記述を見つけることができるが、常に正確な出力を生成するとは限らないことを示しています。

8.4 タスク全体での LLM パフォーマンス分析の HTML 理解

ここで、結果を集計して分析し、主な結論を導き出します。

8.4.1 事前訓練効果: 大きなテキストコーポラに関する事前訓練

微調整された事前トレーニング済みの LLM は、HTML のみのデータでトレーニングされた LLM よりも優れており、自律 Web ナビゲーション (表 2b) で 34.1% 以上、セマンティック分類タスク (表 2) で 10% から 12.7% パフォーマンスが向上しています。

Autonomous Web Navigation は最も困難なタスクであるため、パフォーマンスの向上は、HTML 理解タスクにおける LLM の価値の有望な証拠です。具体的には、事前トレーニングなしの LLM は、単純なテキスト マッチングを必要とする Web サイトでのみ、微調整された事前トレーニング済みモデルに匹敵することがわかります。対照的に、クリック チェックボックスなどの Web サイトでは、テキスト マッチングが難しく、事前トレーニングが良好なパフォーマンスの鍵であることがわかりました。また、事前トレーニングを行わないと、モデルの出力が、無効な辞書や非整数値の無効な参照など、誤った形式になることがよくありました。これは、事前トレーニングに使用される大規模なコーパスが、モデルが一般的な HTML 構造を学習するのに役立つことを示唆しています。

8.4.2 アーキテクチャの影響: T5 ベースのモデルはすべてのタスクで最高のパフォーマンスを発揮

エンコーダー/デコーダー T5 ベースのモデルは、3 つのタスクすべてでより優れたパフォーマンスを発揮します。Autonomous Web Navigation タスクでは、エンコーダー/デコーダー (WebN-T5) アーキテクチャが WebN LaMDA-1B よりも優れているか、同等です (図 2a)。セマンティック分類では、最小のエンコーダー/デコーダー モデル (WebC-T5 ベース) は、はるかに大きなデコーダーのみのモデル (WebC-LaMDA-1B または WebC-PaLM-8B) および最大のエンコーダーのみのモデル (WebC-T5 ベース) と同等のパフォーマンスを発揮します。BERT-large) には 85M 以上のパラメーターがあります (表 2)。また、HTML データのみでトレーニングした場合、デコーダーのみの PaLM-8B は、はるかに小さいエンコーダー/デコーダー T5-large よりもパフォーマンスが悪いことがわかります。最後に、Description Generation エンコーダー デコーダー アーキテクチャでは、BLEU スコアが高くなります。

T5 ベースのモデルの強力なパフォーマンスの 1 つの考えられる説明は、これらのモデルのエンコーダー/デコーダー アーキテクチャです。つまり、T5 モデルは、LaMDA および PaLM デコーダーには存在しない双方向アテンション メカニズムを備えたエンコーダーを利用します。双方向アテンション メカニズムは、HTML ページを両端から処理できるため、ツリー構造の HTML ページが固定された直線的なテキスト シーケンスに変換される際の情報の損失を潜在的に克服できます。

8.4.3 モデルサイズの影響: サイズ (サブリニアリー) が重要

タスク全体で、モデルのパフォーマンスにおいてアーキテクチャが重要な役割を果たしているようです。モデルのサイズとパフォーマンスも正の相関がありますが、収益は減少します。たとえば、セマンティック分類のモデル サイズに関して、モデルのパフォーマンスはおおよそ $O(\log \log n)$ です (付録の図 4b)。Autonomous Web Navigation タスクでは、パフォーマンスはモデルのサイズに応じてゆっくりと増加します (表 8) が、Description Generation では横ばいになります (表 3)。

8.5 議論

双方向の注意とコーパスのトレーニング: 大規模なコーパスでの事前トレーニングにより、 ≤ 4.5 倍のパフォーマンス向上が得られます。モデルが大きいほど優れている傾向があり、タスク全体で T5 の全体的なパフォーマンスが最も優れているのは、双方向の注意のおかげだと考えています。PaLM と LaMDA には事前トレーニング コーパスに HTML とその他のコードが含まれていますが、BERT と T5 アーキテクチャには含まれていませんでした。これは、HTML を理解するために微調整されている場合、強力なパフォーマンスを得るために HTML での事前トレーニングが必要ないことを示しています。これにより、双方向の注意の役割の背後にある仮説が強化され、HTML を使用してコーパスで事前トレーニングすることにより、T5 アーキテクチャのパフォーマンスをさらに向上させる可能性が開かれます。

ラベル付けへの実際的な影響:事前トレーニング済みの LLM が利用可能な場合、新しいエキスパート データはほとんど必要ありません (Web ナビゲーションと分類タスクがそれぞれ 200 分の 1 と 30 分の 1 に削減されます)。これは、実際のアプリケーションに大きな影響を与える可能性があり、データ収集の時間とコストを桁違いに削減します。

大きいほど良いとは限りません。モデルのサイズを選択するときは、モデルのトレーニングと評価の時間とコストとともに、予想されるパフォーマンスの向上 (せいぜい準線形、最悪の場合漸近) を考慮する必要があります。たとえば、分類タスクでは、最大のモデル WebC-PaLM 62B は微調整に数日かかり、30 Hz で評価されますが、WebC-T5-large は数時間で微調整され、700 Hz で評価されます。精度が 1% 向上するのに、さらに費用がかかります。一方、BERT モデルは数分でトレーニングします。アプリケーションが高い精度を必要としない場合、これらは適切な選択かもしれません。

コンテキスト ウィンドウがボトルネック: HTML 理解タスクの主なボトルネックは、現在の LLM がサポートするコンテキスト ウィンドウの長さのようです。MiniWoB のページよりも桁違いに大きい実際の Web サイトで Web ナビゲーション タスクを評価することは依然として不可能です。同様に、スニペットのサイズを大きくすると、パフォーマンスが大幅に低下することがわかりました。これにより、HTML の理解は、将来の LLM 開発にとって興味深いベンチマークになります。たとえば、LLM コンテキスト ウィンドウで使用する Web コンテンツの状態表現を圧縮するには、新しいメソッドを開発する必要がある場合があります。

9 結論

HTML を理解するために、正規のタスクと微調整された LLM を提示しました。さまざまなアーキテクチャ、データセットのサイズ、およびベースラインに対する包括的な評価と分析により、実用的な調査結果が得られ、これらのモデルの現在の制限が強調されます。a) 事前トレーニングはパフォーマンスにとって重要であり、ラベル付きデータの要件を減らし、サンプル効率を最大 200 倍向上させることができます。b) モデル アーキテクチャは 2 番目に重要な要素であり、双方向の注意とエンコーダ/デコーダ アーキテクチャを備えた T5 モデルは、ボード全体で最高のパフォーマンスを発揮します。c) 選択肢が与えられた場合、モデルのサイズはモデルのトレーニングと評価のパフォーマンスのコンテキストで評価する必要があります。これは、モデルのサイズがそのパフォーマンスと準線形的に相関するためです。最後に、提案された HTML 理解タスクは、現在の LLM を制限する比較的短いコンテキスト ウィンドウを強調しており、この制約を組み込むか排除する将来の研究の可能性を示唆しています。

謝辞

Web ナビゲーションに関する貴重な議論してくれた Marcin Moczulski 氏、MiniWoB ベースラインの結果を提供してくれた Peter Humphreys 氏、フィードバックとサポートを提供してくれた Alex Rizkowsky、Douglas Eck、Zoubin Ghahramani 氏に感謝します。

参考文献

- レナード・アドルフス、ベンジャミン・ベルシンガー、クリスチャン・バック、ミシェル・チェン・ヒューブシャー、マシミル・イアーノ・チャラミタ、ラッセ・エスペホルト、トーマス・ホフマン、ヤニック・キルヒャー。インタラクティブなエージェントで検索エンジンを強化します。arXiv プレプリント arXiv:2109.00527, 2021.
- Armen Aghajanyan, Dmytro Okhonko, Mike Lewis, Mandar Joshi, Hu Xu, Gargi Ghosh, Luke Zettlemoyer. Htlm: ハイパーテキストの事前トレーニングと言語モデルのプロンプト。arXiv プレプリント arXiv:2107.06955, 2021.
- ジェイコブ・オースティン、オーガスタス・オデナ、マクスウェル・ナイ、マールテン・ボスマ、ヘンリック・ミハレフスキー、デビッド・ドーハン、エレン・ジャン、キャリー・カイ、マイケル・テリー、クオック・リー 他大規模言語モデルによるプログラム合成。arXiv プレプリント arXiv:2108.07732, 2021.
- リシボンマサニ、ドリュエ・A・ハドソン、イーサン・アデリ、ラス・アルトマン、シムラン・アローラ、シドニー・フォン・アルクス、マイケル・S・バーンスタイン、ジャネット・ボーグ、アンソニー・ボセルト、エマ・ブランスキル 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,
- トム・ブラウン、ベンジャミン・マン、ニック・ライダー、メラニー・サブピア、ジャレッド・D・カブラン、ブラフラ・ダリワル、アービンド・ニールカタン、プラナフ・シャム、ギリッシュ・サストリー、アマンダ・アスケル 他言語モデルは数ショット学習器です。神経情報処理システムの進歩, 33:1877–1901, 2020.

- アンドレア・バーズ、デニス・アルサン、サンジナ・アグラワル、ランジータ・クマール、ケイト・セーンコ、ブライアン・A・ブラマー。
不確実または不十分な自然言語コマンドを使用したインタラクティブなモバイル アプリ ナビゲーション。 arXiv プレプリント arXiv:2202.02312, 2022.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann 他 Palm: パスウェイによる言語モデリングのスケールアップ。 arXiv プレプリント arXiv:2204.02311, 2022.
- ジェイコブ・デブリン、ミンウェイ・チャン、ケントン・リー、クリスティーナ・トウタノフ。パート: 言語理解のための深い双方向トランスフォーマーの事前トレーニング。 arXiv プレプリント arXiv:1810.04805, 2018.
- オスカー・ディアス、イツィアール・オタドゥイ、ゴルカ・ブエンテ。 Web フォーム入力のユーザー主導の自動化。の Web エンジニアリングに関する国際会議, pp. 171–185. スプリングer, 2013年。
- Izzeddin Gur, Natasha Jaques, Yingjie Miao, Jongwook Choi, Manoj Tiwari, Honglak Lee, Alexandra Faust. ゼロショット構成強化学習の環境生成。ニューラル情報処理システムの進歩, 34:4157–4169, 2021.
- Zecheng He, Srinivas Sunkara, Xiaoxue Zang, Ying Xu, Lijuan Liu, Nevan Wichers, Gabriel Schu biner, Ruby Lee, Jindong Chen. Actionbert: ユーザー インターフェイスのセマンティックな理解のためにユーザー アクションを活用します。人工知能に関する AAAI 会議の議事録, 第 35 巻, 5931 ~ 5938 ページ, 2021 年。
- ピーター・C・ハンフリーズ、デビッド・ラボソ、トビアス・ポーレン、グレゴリー・ソーントン、ラチータ・チャバリア、アリスター・マルダル、ジョシュ・エイブラムソン、ペトコ・ゲオルギエフ、アダム・サントロ、ティモシー・リリックラップ。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。機械学習に関する国際会議, pp. 9466–9482. PMLR, 2022年。
- Sheng Jia, Jamie Ryan Kiros, Jimmy Ba. DOM-q-NET: 構造化言語に基づいた強化学習。 In International Conference on Learning Representations, 2019. URL <https://openreview.net/forum?id=HJgd1nAqFX>.
- チャック・ジョーゲンセンとキム・ピンステッド。 emg ベースのサブボカル音声認識を使用した Web ブラウザ コントロール。システム科学に関する第 38 回年次ハイワイ国際会議の議事録, pp. 294c–294c. IEEE, 2005 年。
- ジャレッド・カブラン、サム・マッカンドリッシュ、トム・ヘニハン、トム・B・ブラウン、ベンジャミン・チェス、リウオン・チャイルド、スコット・グレイ、アレック・ラドフォード、ジェフリー・ウー、ダリオ・アモダイニューラル言語モデルのスケールアップ。 arXiv プレプリント arXiv:2001.08361, 2020.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Sori がカットしました。 Albert: 言語表現の自己管理型学習のためのライト ベルト。学習表現に関する国際会議, 2020年。
- Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tianlin Shi, Percy Liang. ワークフローに基づく探索を使用した Web インターフェイスでの強化学習。 arXiv プレプリント arXiv:1802.08802, 2018.
- Sahisnu Mazumder と Oriana Riva. Flin: Web ナビゲーション用の柔軟な自然言語インターフェース。 arXiv プレプリント arXiv:2010.12844, 2020.
- 中野礼一郎、ジェイコブ・ヒルトン、スチル・バラジ、ジェフ・ウー、ロン・オウヤン、クリスティーナ・キム、クリストファー・ヘッセ、シャンタ・ジェイン、ビニート・コサラジュ、ウィリアム・サンダース 他 Webgpt: 人間のフィードバックによるブラウザ支援の質問応答。 arXiv プレプリント arXiv:2112.09332, 2021.
- ロドリゴ・ノゲイラとチョキョンヒョン。エンド ツー エンドの目標主導型 Web ナビゲーション。神経情報処理システムの進歩, 29, 2016.
- クリストファー・オルストン、マーク・ナジューク 他 Web クローリング。 Foundations and Trends® in Information Retrieval, 4(3):175–246, 2010.
- Evan Zheran Liu, Kelvin Guu, Percy Liang. 自然言語コマンドの Web 要素へのマッピング。 arXiv プレプリント arXiv:1808.09132,

コリン・ラフェル、ノーム・シェイザー、アダム・ロバーツ、キャサリン・リー、シャラン・ナラン、マイケル・マテナ、ヤンチー・チョウ、ウェイ・リー、ピーター・J・リュー 他統合されたテキストからテキストへのトランスフォーマーを使用して転移学習の限界を探ります。 J. マッハ、学び。解像度、21(140):1-67, 2020。

Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, および Percy 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, pp. 3135-3144. PMLR, 2017年。

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, Denny Zhou. モバイル BERT: リソースが限られたデバイス向けの、タスクに依存しないコンパクトな BERT。計算言語学協会の第 58 回年次総会の議事録。計算言語学会, 2020。

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Boss, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Kathleen S. Meier Hellstern, Meredith Ringel Morris, トウルシー・ドシ, レネリート・デロス・サントス, トー・ジュ・デューク, ジョニー・ソー・レイカー, ベン・ゼベンベルゲン, ヴィノードクマール・プラバカラン, マーク・ディアス, ベン・ハッチンソン, クリストン・オルソン, アレハンドラ・モリーナ, エリン・ホフマン=ジョン, ジョシュ・ローラ・アロヨ, ラヴィ・ラジャクマール, アレナ Butryna, Matthew Lamm, Victoria Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguerre-Arcas, Claire Cui, Marian Croak, Ed H. Chi, Quoc Le. ラムダ: ダイアログ アプリケーションの言語モデル。CoRR, 2022年。

ダニエル トヤマ, フィリップ ハメル, アニタ ゲルゲリー, ゲオルゲ コマニチ, アメリア グライス, ザファラリ アーメド, タイラー ジャクソン, シンプル ムラド, ドイナ プレカップ。Androidenv: 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。

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, Illia Polosukhin. 必要なのは注意だけです。神経情報処理システムの進歩, 30, 2017。

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, Colin Raffel. mt5: 大規模な多言語の事前トレーニング済みのテキストからテキストへのトランスフォーマー。arXiv プレプリント arXiv:2010.11934, 2020。

Shunyu Yao, Howard Chen, John Yang, および Karthik Narasimhan. ウェブショップ: 言語エージェントによるスケラブルな現実世界の Web インタラクションに向けて。arXiv プレプリント arXiv:2207.01206, 2022。

付録

A.1 データセットの詳細

説明の分布を調べると、元の 400K データセットが非常に歪んでいることがわかりました。データセットの 50% をカバーしていたのは、20 の説明 (電子メールやパスワードなど) だけでした。一意の説明ごとに最大 10 個のデータ ポイントが含まれるように、データセットをサブサンプリングしました。また、ほとんどの場合、HTML ラベルには for 属性が定義されていることもわかりました。これにより、モデルがオーバーフィットし、HTML 内の label 要素だけが検出され、他のすべてが無視される可能性があります。この種の「ごまかし」を避けるために、{div, span, a, label} からランダムにサンプリングして HTML ラベルのタグを置き換えます。これらのタグは、HTML にテキストを挿入するためにも頻繁に使用されますが、属性にはほとんど使用されません。

最後に、HTML にテキストが 1 つしかない例を削除しました。これは、モデルが HTML 内の唯一のテキストを見つけることで簡単に説明を生成できるため、モデルの重みと評価指標にバイアスがかかるからです。この最終ステップの後、合計 85,000 のラベル付けされた例が得られます。

A.1.1 スニペットの生成

図 5 に、スニペット生成手順の概要を示します。

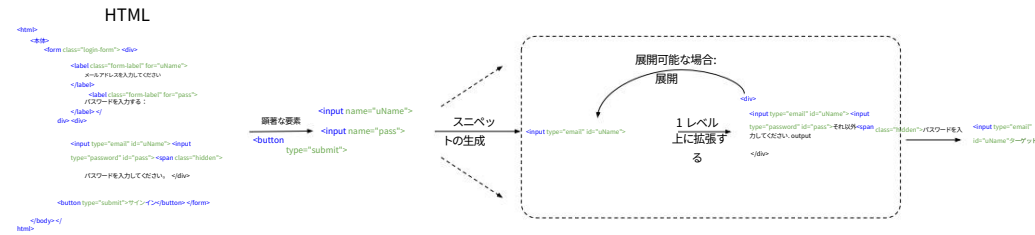


図 5: 完全な HTML Web ページからスニペットを生成するための前処理パイプラインの概要。ページを指定して、顕著な要素を検出し、検証ヒューリスティックが失敗するまで HTML ツリーを再帰的に上に移動することにより、それらのそれぞれについてスニペットを抽出します。

A.2 その他の結果

A.2.1 意味分類

エラー分析。開発セット全体で T5-3B モデルの 50 個のエラーを手作業で調べ (表 4)、それらを考案した 9 つのエラータイプのいずれかに割り当てました。エラーの 32% は HTML スニペットの情報不足が原因であることがわかりました。これは主に、スニペット抽出プロセス中に情報が失われた結果です。注釈エラーまたは電子メール/ユーザー名のあいまいさがエラーの 30% を占めています。これらは、注釈付きデータを修正したり、あいまいさを解決するための追加情報を追加したりせずに改善することはできません。また、モデルがより一般的なカテゴリを選択する場合や、近くのテキストがモデルを誤解させる場合があることもわかりました。後者は通常、HTML スニペットが長く、要素の大部分がノイズである場合に発生します。

エラーの種類	例の割合
HTML スニペットの情報不十分	30 12 8 10 8 8 4 2 18
誤った注釈 (例: 「組織」ではなく「不明な役割」) 注釈ツールがユーザーの選択を正しく変換しない メール/ユーザー名のあいまいさ	
より一般的なカテゴリ (例: 「カート ヘッダー」ではなく「ヘッダー」) すぐ隣のテキストが誤解を招く 日付の形式が正しくない (例: 「mmm」ではなく「mm」) HTML スニペットに情報がありません その他	

表 4: 手動で調べた 50 を超えるエラーの種類。エラーの 32% は HTML スニペットの情報不足が原因であり、エラーの 30% は注釈に関連するか、あいまいさ (電子メール/ユーザー名) のために改善できないため、残りのエラーはモデルによる誤った予測です。

少数ショット プロンプト表 5 に、540B PaLM モデルの少数ショット プロンプトのパフォーマンスを示します。プロンプト テンプレート <html> Role: <category> を使用してモデルをプロンプトし、カテゴリごとに 1 つの例を示し、貪欲なデコードを使用してカテゴリを生成します。予備実験では、少数ショット プロンプトは 45.6 の精度しか達成できず、同じデータで微調整されたモデルよりもはるかに低いことがわかりました (図 6)。2 つの一般的な問題が見つかりました。モデルは予測をカテゴリに正規化できず、コンテキストの長さのために多くの例が削除されます。

正規化の問題を軽減するための後処理方法と、例の長さを減らすための前処理方法を開発しました。予測に辞書ベースのマッピング (手動で精選された言い換え辞書) を追加すると、パフォーマンスが 52.1 に向上します。

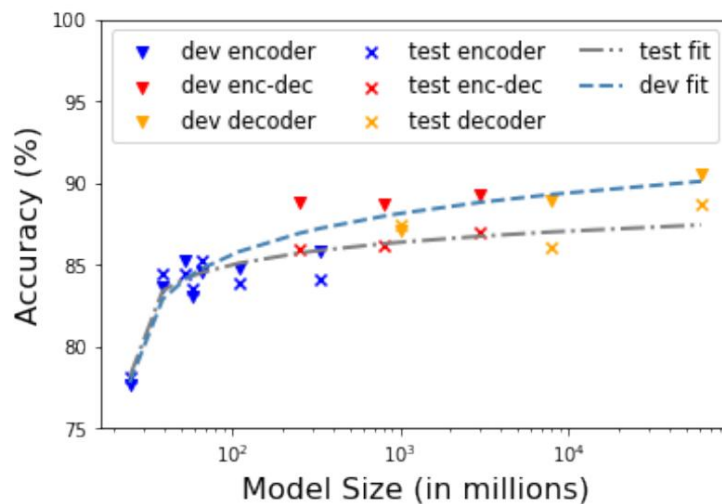
また、「」の周りのトークンの順序を変更して予測を書き直してみました。たとえば、名から名へなど、パフォーマンスが 57.9 にさらに改善されました。最後に、特定の el を削除して、プロンプトの例をきれいにしました。

モデル名	Test	Dev	PaLM-540B	64.2	60.3	- サンプル
ルグリニングなし	57.9	57.2	- カテゴリ	置き換え	なし	
辞書マッピングなし	45.6	45.1				

表 5: 異なる前処理ステップと後処理ステップによる少数ショット プロンプトのパフォーマンス。

「svg」、「path」、「img」、「iframe」などの要素と、すべての要素からクラス属性を削除します。この前処理ステップにより、64.2 が得られます。

図 6: モデル サイズの増加に対するパフォーマンスの比較。モデルのサイズが大きくなるにつれて、PaLM-8B の 7 倍の大きさでありながら最高の精度を達成する PaLM-62B モデルで、全体的な精度の向上が見られます。



A.3 MINIWOBからのサンプル エピソード

微調整された LLM によって推測される Web ナビゲーションのエピソードの例については、表 7 を参照してください。

A.4 詳細なミニウォブの結果

MiniWob でのさまざまなモデルの詳細なパフォーマンスについては、表 8 を参照してください。

A.5 リソース要件

表 6 を参照してください。

表 6: LLM のリソース要件と実行時間。

モデル名	モデル サイズ	TPU バージョン	バッチ サイズ	入力シーケンスの長さ	1 秒あたりの例 (トレーニング)	1 秒あたりの例 (評価)
手のひら	62B	TPU v4 1920			9.313	30.51
手のひら	8B	TPU v4 1920		8	64.4	184.3
T5	3B	TPU v4 512		32	163.8	734.5
ラムダ	1B	TPU v2 512		128 128	363.1	1416

表 7: T5-3B モデルを使用したサンプル Web ページと対応するエピソード。各時間ステップで、前のアクション、命令、および HTML が連結されて 1 つの HTML テキストになります。エピソードの冒頭では、過去のアクションはなく、単に命令と HTML を連結していることに注意してください。

アクションは、後で辞書に解析される一連のトークンとして生成されます。アクションの ref は、同じ値を持つ ref 属性を持つ要素を指しています。たとえば、エピソードの冒頭では、ref: 6 は ref=6 の入力に対応します。エピソードの最後に、モデルが送信ボタンをクリックすると、エピソードが終了します。

ウェブページ

Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit.

qw

mMpY

YVYMQVa

DJfG

lI3Z

Submit

HTML テキスト

アクション テキスト

```
Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit. <body ref="1"><div id="wrap" ref="2"><div id="area" ref="3"><div id="boxes" ref="4"><label ref="5"><input type="checkbox" id="ch0" ref="6" value="False"></input><t class="TEXT\_CLASS" ref="None">qw</t></label><label ref="7"><input type="checkbox" id="ch1" ref="8" value="False"></input><t class="TEXT\_CLASS" ref="None">hHfDn8</t></label><label ref="9"><input type="checkbox" id="ch2" ref="10" value="False"></input><t class="TEXT\_CLASS" ref="None">mMpY</t></label><label ref="11"><input type="checkbox" id="ch3" ref="12" value="False"></input><t class="TEXT\_CLASS" ref="None">YVYMQVa</t></label><label ref="13"><input type="checkbox" id="ch4" ref="14" value="False"></input><t class="TEXT\_CLASS" ref="None">DJfG</t></label><label ref="15"><input type="checkbox" id="ch5" ref="16" value="False"></input><t class="TEXT\_CLASS" ref="None">lI3Z</t></label></div><button id="subbtn" class="secondary-action" ref="17">Submit</button></div></div></body>
```

{アクション: クリック、参照: 6}

```
{action: click, ref: 6} Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit. <body ref="1"><div id="wrap" ref="2"><div id="area" ref="3"><div id="boxes" ref="4"><input type="checkbox" id="ch0" ref="6" value="True"></input><t class="TEXT\_CLASS" ref="None">qw</t></label><label ref="7"><input type="checkbox" id="ch1" ref="8" value="False"></input><t class="TEXT\_CLASS" ref="None">hHfDn8</t></label><label ref="9"><input type="checkbox" id="ch2" ref="10" value="False"></input><t class="TEXT\_CLASS" ref="None">mMpY</t></label><label ref="11"><input type="checkbox" id="ch3" ref="12" value="False"></input><t class="TEXT\_CLASS" ref="None">YVYMQVa</t></label><label ref="13"><input type="checkbox" id="ch4" ref="14" value="False"></input><t class="TEXT\_CLASS" ref="None">DJfG</t></label><label ref="15"><input type="checkbox" id="ch5" ref="16" value="False"></input><t class="TEXT\_CLASS" ref="None">lI3Z</t></label></div><button id="subbtn" class="secondary-action" ref="17">Submit</button></div></div></body>
```

{アクション: クリック、参照: 10}

```
{action: click, ref: 6},{action: click, ref: 10} Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit. <body ref="1"><div id="wrap" ref="2"><div id="area" ref="3"><div id="boxes" ref="4"><input type="checkbox" id="ch0" ref="6" value="True"></input><t class="TEXT\_CLASS" ref="None">qw</t></label><label ref="7"><input type="checkbox" id="ch1" ref="8" value="False"></input><t class="TEXT\_CLASS" ref="None">hHfDn8</t></label><label ref="9"><input type="checkbox" id="ch2" ref="10" value="True"></input><t class="TEXT\_CLASS" ref="None">mMpY</t></label><label ref="11"><input type="checkbox" id="ch3" ref="12" value="False"></input><t class="TEXT\_CLASS" ref="None">YVYMQVa</t></label><label ref="13"><input type="checkbox" id="ch4" ref="14" value="False"></input><t class="TEXT\_CLASS" ref="None">DJfG</t></label><label ref="15"><input type="checkbox" id="ch5" ref="16" value="False"></input><t class="TEXT\_CLASS" ref="None">lI3Z</t></label></div><button id="subbtn" class="secondary-action" ref="17">Submit</button></div></div></body>
```

{アクション: クリック、参照: 12}

```
{action: click, ref: 6},{action: click, ref: 10},{action: click, ref: 12} Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit. <body ref="1"><div id="wrap" ref="2"><div id="area" ref="3"><div id="boxes" ref="4"><input type="checkbox" id="ch0" ref="6" value="True"></input><t class="TEXT\_CLASS" ref="None">qw</t></label><label ref="7"><input type="checkbox" id="ch1" ref="8" value="False"></input><t class="TEXT\_CLASS" ref="None">hHfDn8</t></label><label ref="9"><input type="checkbox" id="ch2" ref="10" value="True"></input><t class="TEXT\_CLASS" ref="None">mMpY</t></label><label ref="11"><input type="checkbox" id="ch3" ref="12" value="True"></input><t class="TEXT\_CLASS" ref="None">YVYMQVa</t></label><label ref="13"><input type="checkbox" id="ch4" ref="14" value="False"></input><t class="TEXT\_CLASS" ref="None">DJfG</t></label><label ref="15"><input type="checkbox" id="ch5" ref="16" value="False"></input><t class="TEXT\_CLASS" ref="None">lI3Z</t></label></div><button id="subbtn" class="secondary-action" ref="17">Submit</button></div></div></body>
```

{アクション: クリック、参照: 14}


```
{action: click, ref: 6},{action: click, ref: 10},{action: click, ref: 12},
{action: click, ref: 14} Select qw, mMpY, YVYMQVa, DJfG, lI3Z and click Submit.
<body ref="1"><div id="wrap" ref="2"><div id="area" ref="3"><div id="boxes"
ref="4"><label ref="5"><input type="checkbox" id="ch0" ref="6" value="True">
</input><t class="TEXT\_CLASS" ref="None">qw</t></label><label ref="7"><input
type="checkbox" id="ch1" ref="8" value="False"></input><t class="TEXT\_CLASS"
ref="None">hHfDn8</t></label><label ref="9"><input type="checkbox" id="ch2"
ref="10" value="True"></input><t class="TEXT\_CLASS" ref="None">mMpY</t></label>
<label ref="11"><input type="checkbox" id="ch3" ref="12" value="True"></input>
<t class="TEXT\_CLASS" ref="None">YVYMQVa</t></label><label ref="13"><input
type="checkbox" id="ch4" ref="14" value="True"></input><t class="TEXT\_CLASS"
ref="None">DJfG</t></label><label ref="15"><input type="checkbox" id="ch5"
ref="16" value="False"></input><t class="TEXT\_CLASS" ref="None">lI3Z</t>
</label></div><button id="subbtn" class="secondary-action" ref="17">Submit
</button></div></div></body>
```

{アクション: クリック、参照: 16}

```
{action: click, ref: 6},{action: click, ref: 10},{action: click, ref: 12},
{action: click, ref: 14},{action: click, ref: 16} Select qw, mMpY, YVYMQVa,
DJfG, lI3Z and click Submit. <body ref="1"><div id="wrap" ref="2"><div id="area"
ref="3"><div id="boxes" ref="4"><label ref="5"><input type="checkbox" id="ch0"
ref="6" value="True"></input><t class="TEXT\_CLASS" ref="None">qw</t></label>
<label ref="7"><input type="checkbox" id="ch1" ref="8" value="False"></input><t
class="TEXT\_CLASS" ref="None">hHfDn8</t></label><label ref="9"><input
type="checkbox" id="ch2" ref="10" value="True"></input><t class="TEXT\_CLASS"
ref="None">mMpY</t></label><label ref="11"><input type="checkbox" id="ch3"
ref="12" value="True"></input><t class="TEXT\_CLASS" ref="None">YVYMQVa</t>
</label><label ref="13"><input type="checkbox" id="ch4" ref="14" value="True">
</input><t class="TEXT\_CLASS" ref="None">DJfG</t></label><label ref="15"><input
type="checkbox" id="ch5" ref="16" value="True"></input><t class="TEXT\_CLASS"
ref="None">lI3Z</t></label></div><button id="subbtn" class="secondary-action"
ref="17">Submit</button></div></div></body>
```

{アクション: クリック、参照: 17}

プレプリント: 審査中

表 8: MiniWoB タスクにおけるさまざまなモデルの成功率の比較。ベースラインの結果は (Humphreys et al., 2022) から借用しています。これらは 0 から 1 の間で正規化されていることに注意してください。

タスク	人間	WebC-T5-3B WebC	T5-3B (SL & RL) (SL) (無難なし)	CCネット CCネット	ビットの世界	ワークフローの序 探索 (SL & RL)	DOM-Q-Net (SL & RL) (Web)	ワークフローのガイ 付探索 (RL)	非探索 ナビゲート	学ぶ Web の付 ゲート (既 習) n/a 1 n/an/an/	集約集約 下 (SL & RL) (増幅)	下	
二等分角ブロック	0.92	該当	該当	0.97	0.29	0.8	該当	n/an/	n/a/	該当	an/a 0.26	0.8	0.8
フライトチェイ	0.87	な	な	0.87	0	0.1	な	an/	a/a/	な	n/an/a 1	0.1	1.1
スクロール	0.82	し 0	し 0	0.93	0.8	0	an/	a/a	a/a	し 0	n/an/an/	該当	該当
図を簡単に選択	0.99	該当	該当	0.99	0.42	N/	該当	an/	1/a/	該当	an/an/a 1	該当	該当
choose-date-medium 0.98 choose-date-soft	0.97	し	し	0.99	0.26	AN/	なし	an/a/	a/a	なし	n/an/a 1.1	なし	なし
choose-list 0.98 circle-center 0.96click-	0.93			0.97	0.12	A	0	0.26	1	0	n/an/an/a	1	1
button-sequence 0.94 click-button 0.96	0.0	0.05	0.99	0.19	0.0.25	0.16	n/an/	a/a/	0.16	0.16	n/an/an/	0.26	0.26
click-checkboxes-large 0.87 click-	0.26	0.0	0.97	0.36	0.98	0.98	該当	a 1	a/a	該当	an/an/an/	0.98	0.98
checkboxes-soft 0.73 click-checkboxes-	該当			1.1	0.47	0.22	なし	n/an/	1	な	an/a n/	1.1	1.1
transfer 0.98 click-checkboxes-0.96	0.0	0.14			0.78	0.62		an/		し	an/an/an/		
collapsible-2 0.97 click-collapse 0.97 click-	1.1	該当	0.71	0	N/	0.99	1	an/		1.1	an/an/an/	0.68	0.84
0.97 click-dialog-2 0.99 click-dialog-1 1.0	0.22	なし	0.95	0.04	AN/	0.68	an/a/			0.84	an/an/an/	0.51	0.94
0.99 click-menu-2 0.98 click-menu 0.97 click-	0.54	10.96	0.99	0.36	AN/	0.51	1 n/			0.94	an/an/an/	0.64	0.64
option 0.99 click-pie 0.98 click-accordion	0.63	0.43	0.98	0.32	A 0.11	0.64	an/a/			0.64	n/an/an/	1	1
click-shades 0.91 click-shape 0.88 click-2-	0.96	0.34	0.98	0.17	0.10	0.98	1 n/	n/a/	0.99	an/an/an/	0.65	0.99	
easy 0.99 click-tab-2-hard 0.96 click-2-	0.0	0.84	1.1	0.81	0.23		an/	a/a/	1.1	n/an/an/	1.1	1.1	1.1
medium 0.97 click-tab-2 0.97 click-	0.27	0.01	1.1	0.82	0.53		an/a	a 1	1.1		an/an/an/	1.1	1.1
0.99 click-test-2 0.99 click-test 1.0	0.241	0.01		0.88	1.031		1 n/	1 a/	1		an/an/an/	1	1
click-test-1 click-widget 0.82 copy-	1.8			0.95			an/	a/	n/		an/an/		
paste-2 0.94 copy-paste 0.94 collapse	0.9	0.23	0.99	0.59	0.16		an/	a	a/		an/an/an/		
0.82 count-sides 0.98 drag-box 0.98	し	0.35	0.83	0.52	0.13	0.65	an/	1 a/	a 1		an/an/an/	0.16	0.16
cube 0.93 drag-item 0.98 drag-items-	0.37	10.96	0.94	0.22	0.28	1.1	an/	a/a/	0.32		an/an/a	0.13	0.13
0.87 drag-items 0.93 drag-slides 0.98	0.87	該当		0.15	0.15	1.8	an/	a/			n/	1.1	1.1
sort numbers 0.92 email-check-delete	0.810	し	0.97	0.15	0.07	0.07	an/	a/a/	a		n/an/an/		
0.99 email-inbox-forward-ni-turk 1.0	0	0.6	0.01	0.27	1		a 1	a/a	0.99		an/an/an/	0.07	0.07
inbox-forward-ni 0.91 email-inbox-	0.53	1	0.04	0.11	0.32	n/an/	1.1	0.64	an/a		an/a	0.27	0.99
forward 0.96 email-inbox-inbox-	0.8	0.38	0.95	0.11	N/AN/	0.9	an/	1 a/	n/a/		n/a/	0.64	0.64
電子メール inbox-ni-turk 0.93 email-inbox-	し	0.78	0.99	0.61	A		し	an/	a 1		a/a	該当	n/a/
noscroll 0.96 email-inbox-reply 0.94email-	0.12	0.14	1.98	0.19	0.08	0.22	an/	1	0.98		なし	an/a	
inbox-star-reply 0.95 email-inbox-star-	0.8	0.0	0.18	0.54	0.97	0.64	an/				1.1	1.1	1.1
date 0.97 enter-password 0.96 enter-2	0.15	0.54	0.27	0.83	該当	an/	n/a	n/a 1			なし	なし	なし
0.91 enter-text-dynamic 0.97 enter-	0.241	し		0.95	18	し	an/				なし	なし	なし
0.98 入力時間 0.98 検索中 0.94 検索	該当	当	0.95	0.30	0.2		an/				し	1	1
0.96 フォーカス テキスト 2 0.99 フォーカス	なし	なし		0.94		0.64	an/		0.93		1.1	1.1	1.1
1 グリッド検索 0.87 検索 ハイパー	1			1			an/				n/		
テキスト 2 ハイライト テキスト 1.0	0.13		0.96			0.55	an/				a/a		
ン ユーザー ポップアップ ウィンドウ 0.98	n/a/	該当	0.63	0.02		n/a 1	an/	n/a/	0.76		0	0	0
items multi-layout multi-drag	a	なし	0.79	0.04		0.93	an/	a/n/	n/a/		0	0	0
navigate-tree number-checkbox-	0.41	7 該当	0.85	0.21		n/an/	an/	a/a/	a/a/		0.59	0.76	
table-2 read-table resize-textare	n/a/	なし	1.1	0.74		a	an/	a/a/	a/a/		0.3	0.3	
angle scroll-text-2 scroll-text search-	a/a/	0.09		0.61		0.59	an/	a/a/	an/		0.31	0.31	
engine simon-says simple algebra	a/a/	1.1	0.79	0.23		n/an/	an/	a/a/	an/		0.18	0.18	
simple 算術 ショール メニュー 0.97	a/a/	該当	1	0.61		an/	an/	1 n/	an/		該当	1	1
シヤル ステアーズ 一部の ショール メニュー	a/a	なし	0.98	0.05		an/	an/	a/a/	an/		なし	なし	なし
テキスト テキスト エディター テキスト 変換	0.33	9/a/	0.99	0.13		an/	an/	a/a/	an/		0.01	0.01	
Unicode テキスト 使用 オートコンプリ	0.60	a	0.99	0.26		an/	an/	a/a/	an/		0.41	0.41	
使用 カラー ボール 2 使用 カラー ボール	n/a/	0.43	0.97	0.11		an/	an/a/	a/a	an/		0.92	0.92	
スライダー 2 使用 スライダー 使用 スライダー	a	n/a/	1.1	0.22		an/	1.1	a			0.66	0.66	
加算	0.23	a/a/a/	1.1	0.0		0.34	an/	n/a	0.54	0.93	0.66	0.66	
	n/a/	n/a/	1.1			an/	1 n/	n/a	n/		1.8	1.8	
	a/a	a/a	1.1	0.01		an/	an/	1 n/	an/		該当	該当	
	0.38	0.09	1.1	0.3		an/	an/	a 1	an/		なし	なし	
	0.097	0.09	0.05			an/an/	an/	1.8	a		0.77	0.93	
	n/a	n/a/	0.13			an/	an/	該当	0.99		該当	該当	
	0.98	a	0	an/	an/	し 1	0.96				なし	なし	
	0.89	0.26	0.11	an/	an/a	1.8	1 n/				0.54	0.99	
	0 n/	n/a/	0.09	an/an/		該当	なし	a 1.1			1.10	1.10	
	an/	a/a	0.02					0.9 n/			1.1	1.1	
	a 1.1	0.21	0.02			a 0.77		an/			0.9	0.9	
	0.49	0	0.98	0.04		n/an/		a 1					
	0 n/	0.92	1.1	0.39		an/		1.1					
	an/a	n/a	0.35			a							
	0.88	0.99	0.97	0.35		0.99					0.52	0.31	
	0.72	0.01	0.88	0.05		n/a					0.1	1	
	n/	1.1	0.96			1.1					1.1	1.1	
		おま	1	0.99									
	0.82	0/a 1.1 0.42		0.66	0.30 0.34 0.18 n/	a 0.03 0.4 0.93 0.66	an/an/ah/an/an/an/a/a	0.03 0.61 0.0 1 0 0.08	0.31 0.0 0.83 0.95 0.26				
	0.99	n/a	0	0.21	0.2	0	n/	n/	0	n/a/	0.2	0.3	
	0.97	0.83	該当	0.4	0.13	該当	an/	an/	n/a	a/a/	0.15	0.13	
	0.97	0.88	なし	0.51	0.9	なし	an/	an/	1 n/	a/a	0.9	0.91	
	0.98	0.91	0.89	0.68	0.36	0.9	an/	an/	a	1 a/	0.9	1.8	
	0.94	n/an/	0.40	0.02	該当	an/	an/	1 n/	a/a/	該当	該当	該当	
	0.96	an/	0.64	0	なし	なし	a	a	a	なし	なし	なし	
	0.18	an/	該当	1.11	0.13	し 0	0.99	1 n/	1 n/	1.1	1 a/	し 1	10.78
	0.95	an/	なし	1	0.0	0.78	該当	an/	an/	a/a/	0.78	1.1	
	0.96	an/	0.48	1		該当	し	an/	an/	a/a/	0.99	1	
	0.98	an/a	0.64	1	0.32	なし	0.99	a	a	0.99	a/		
	0.96	0.34	0.99	0.88	0.0	0.2	0.05	1 n/	1 n/	n/a/	a/a/	0.05	0.16
	0.95	n/	該当	1.1		0.99	an/	an/	a/a/	a/a/	1	0.0	
	0.97	an/	なし	0.99	0.01	該当	an/	an/	a/a/	a/a			
	0.94	an/a	0.34	0.99	0.27	0.16	なし	an/	an/	a/a	1	0.18	0.11
	0.87	0	該当	0.99	0.26	0.0	0.26	an/a	an/	0.99	0.0	0.38	
	0.97	0.02	なし	0.88	0.11	該当	an/	an/	n/a/		0.11	0.96	
	0.97	0.21	0.0	0.94	0.04	なし	an/	an/	a/a		0.1	0.1	
	0.97	n/	0.24	0.15		0.01	an/	a	0.01				
	0.62	該当	該当	0.02	0.38	0.01	an/	1 n/	0.42		0.38	0.28	
	0.86	し	なし	0.97	0.03	0.96	0.39	an/	an/	1 n/	0.96	0.04	
	0.96	0.48	0.40	1	0.01	0.0	該当	an/	an/	a/a/	0.1	0.071	
	0.89	該当	該当	0.98	0.28	なし	an/	an/	a 0.47		0.28	0.28	
	0.96	なし	なし	1	0.03	0.04	0.37	an/	an/	n/a	該当	0.04	0.42
	0.88	0.22	0.15	0.96	0.11	0.07	該当	an/	a		なし	1.0	
	0.88	該当	該当	1.0		該当	し/a	an/	1 n/	0.98	し/		
	0.86	なし	なし	0.75	0.19	なし	0.78	and/	n/a/	a/a/	0.07	0.01	
	0.71	0.07	0.05	0.86	0.32		n/a/	and/	and/	a/a/	a/a/	0	
	0.96	該当	該当	0.75	0.07	0.23	a/a/a	and/	a 0.04	a/a/	0.01	0.47	
	0.54	し	し	0.07	0	0.04	and/	and/	n/a	a/a/	0.01	0.16	
	0.94			0.38		n/a	and/	and/		a/a/	1.0	なし	
	0												