

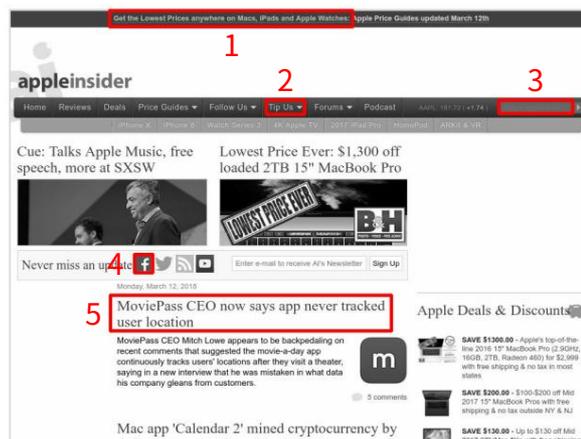
自然言語コマンドを Web 要素にマッピングする

Panupong Pasupat Tian-Shun Jiang Evan Zheran Liu Kelvin Guu Percy Liang スタンフォード大学
{ppasupat,jiangts,evanliu,kguu,pliang}@cs.stanford.edu

概要

Web は、テキスト、構造、および空間のプロパティを備えたリッチでオープン ドメインの環境を提供します。我々は、この環境での言語の新しいタスクを提案します: 自然言語コマンド (例えば、「2 番目の記事をクリック」) が与えられた場合、Web ページ上の正しい要素 (例えば、ハイパーリンクまたはテキスト ボックス) を選択します。

機能参照 (「このサイトの作成者を見つける」など)、リレーショナル推論 (「john による記事」など)、視覚的推論 (「一番上の記事」など) などのさまざまな現象を捉える 50,000 を超えるコマンドのデータセットを収集しました。また、データセットに存在するさまざまな現象を捉える 3 つのベースライン モデルを実装し、分析しました。



- 1: Apple Deals をクリック
- 2: ヒントを送る
- 3: iPhone 7 を検索に入力する
- 4: Facebook をフォローする
- 5: 最新のニュース更新を開く

図 1: Web ページ appleinsider.com の自然言語コマンドの例。

1 はじめに

Web ページは、構造化されたプロパティ (内部ツリー表現など) と構造化されていないプロパティ (テキストや画像など) の両方を含む複雑なドキュメントです。コンテンツとデザインが多様であるため、Web ページは自然言語の土台となるタスクに豊富な環境を提供します。

特に、図 1 に示すように、自然言語コマンドを Web ページ要素 (リンク、ボタン、フォーム入力など) にマッピングするタスクを検討します。一部のコマンドは要素のテキストを直接参照しますが、他の多くのコマンドはより複雑なものを必要とします。テキスト、属性、スタイル、ドキュメント オブジェクト モデル (DOM) からの構造データ、およびレンダリングされた Web ページからの空間データなど、Web ページのさまざまな側面に関する推論。

私たちのタスクは、自然言語の発話をデータベースクエリやオブジェクト操作などのアクションにマッピングすることを目的としたセマンティックパーシング文学に触発されています (Zelle and Mooney, 1996; Chen and Mooney, 2011; Artzi and Zettlemoyer, 2013; Berant et al. al., 2013; Misra et al., 2015; Andreas and Klein, 2015)。これらのアクションは通常、固定された既知のスキーマを持つ環境で動作しますが、

Web ページにはさまざまな構造が含まれているため、作業がより自由になります。同時に、私たちのタスクは参照ゲーム (Golland et al., 2010; Smith et al., 2013; Andreas and Klein, 2016) と見なすことができ、システムは自然言語参照を与えられたオブジェクトを選択する必要があります。Web ページ要素の属性の多様性と、要素を解釈するためにコンテキストを使用する必要があるため、Web ページは特に興味深いものになります。

自然言語による要素の識別には、いくつかの実世界での応用があります。主なものは、Web ページと対話するための音声インターフェイスを提供することです。これは、視覚障害者向けの補助技術として特に役立ちます (Zajicek et al., 1998; Ashok et al., 2014)。もう 1 つの使用例は、ブラウザーの自動化です。自然言語コマンドは、CSS や XPath SE よりも脆弱ではありません。

レクチャー (Hammoudi et al., 2016) であり、さまざまな Web サイトで一般化できます。

50,000 を超える自然言語コマンドのデータセットを収集しました。図 1 に示すように、

| 現象 | 説明コマンドに | 例 | 額 |
|---|--|--|-------|
| 部分文字列一致 | は、要素のテキストの部分文字列のみが含まれます (ステミング後)。 | 「energy.gov でインターンシップを見る」 → 「キャリアとインターンシップ」リンク 「サインインをクリック」 → 「ロ | 7.0% |
| 言い換え | このコマンドは、要素の文章。 | グイン」リンク | 15.5% |
| 目標の説明 | コマンドは、アクションを説明したり、質問したりします。 | 「言語を変更」 → 「英語」というテキストが表示されたクリック可能なボックス 「ベンガル貿易に関する記事に移動」 → 記事のタイトル リンク | 18.0% |
| 要約 | このコマンドは、要素内のテキストを要約します。 | 「青いボタンをクリック」 | 1.5% |
| 要素の説明 このコマンドは、要素のプロパティを説明します。 | | | 2.0% |
| 関係推論 コマンドは、他の要素またはその周囲のコンテキストとの推論を必要とします。 | | 「Cookie 情報を表示」 → ニュース セクションではなく Cookie 警告バーの「詳細情報」 「最初の記事ををクリック」 | 2.5% |
| 順序推論 空間推論 | コマンドは順序を使用します。 | | 3.5% |
| | コマンドは要素の位置を記述します。 | 「ページの左上にある 3 つのスラッシュをクリックします」 「お気に入りボタンを選択します」 | 2.0% |
| 画像ターゲット | 対象は画像 (テキストなし) です。 | | 11.5% |
| フォーム入力ターゲット | ターゲットは入力 (テキスト ボックス、チェック ボックス、ドロップダウン リストなど) です。 | 「検索バーで、testing と入力します」 | 6.5% |

表 1: データセット内のコマンドに存在する現象。各例には、複数の現象が含まれる場合があります。

コマンドには、関係的、視覚的、機能的な推論などの多くの現象が含まれており、セクション 2.2 でさらに詳しく分析します。また、検索、要素の埋め込み、およびテキストの配置に基づいて、タスクの 3 つのモデルを実装しました。私たちの実験的分析は、関数参照、関係参照、および視覚的推論が、自然言語コマンドから要素を正しく識別するために重要であることを示しています。

2 タスク

要素 e_1, \dots, e_k とコマンド c の場合、タスクは要素 $e \in \{e_1, \dots, e_k\}$ をコマンド c で記述する。トレーニング データとテスト データには、 (w, c, e) トリプルが含まれています。

2.1 データセット

1,835 の Web ページで 51,663 コマンドのデータセットを収集しました。データを収集するために、まず上位 10,000 の Web サイト 1 のホームページを Google Chrome でレンダリングしてアーカイブしました。動的コンテンツを読み込んだ後、DOM ツリーと各要素のジオメトリを記録し、レンダリングされた Web ページを保存しました。正しくレンダリングされ、不適切なコンテンツが含まれていない英語の Web ページをフィルタリングしました。次に、クラウド ワーカーにそれぞれのアクションについてブレインストーミングを依頼しました。

各アクションは、フィルタリングされたインタラクティブな要素 (表示されるリンク、入力、およびボタンを含む) のリストから (選択した) 1 つの要素を正確に参照する必要があります。奨励しました

労働者は、選択された要素の正確な言葉遣いを含まない各コマンドにボーナスを与えることにより、要素の正確な言葉を使用することを避けます。最後に、データを 70% のトレーニング、10% の開発、および 20% のテスト データに分割します。

3 つのセットの Web ページは重複しません。

収集された Web ページには平均 1,051 の要素があり、コマンドは平均で 4.1 ~ ケンの長さです。

2.2 コマンドに存在する現象要素の正確なテキストを参照する

以外に、コマンドはさまざまな方法で要素を参照できます。トレーニング データから 200 の例を分析し、これらのコマンドに存在する現象を分析しました (表 1 を参照)。

コマンドが要素のテキストを直接参照する場合でも、ページ上の他の多くの要素もコマンドと単語が重複しています。平均して、コマンドにはページ上の 5.9 個のリーフ要素と重複する単語があります (ストップワードは数えません)。

3 モデル

3.1 検索ベースのモデル

多くのコマンドは、テキストの内容によって要素を参照します。そのため、最初に、コマンドを検索クエリとして使用して、TF-IDF スコアに基づいて最も関連性の高い要素を取得する単純な検索ベースのモデルを検討します。

具体的には、各要素は、(1) テキスト コンテンツのトークン化とステミング、および (2) テキスト コンテンツのトークン化によって計算されるバッグ オブ トークンとして表されます。

¹ <https://majestic.com/reports/majestic-million>

句読点およびキャメルケース境界での属性 (id、クラス、プレースホルダー、ラベル、ツールチップ、aria-text、name、src、href)。単語の頻度を計算するとき、(2) からケンへの属性を係数 α でダウンウェイトします。実験用の開発セットで調整された $\alpha = 3$ を使用します。

ドキュメントの頻度は、トレーニング データセットの Web ページで計算されます。複数の要素が同じスコアを持つ場合、ヒューリスティックに最も顕著な要素、つまり DOM 階層の事前注文トラバーサルで最初に表示される要素を選択します。

3.2 埋め込みベースのモデル

2 つのテキストを照合する一般的な方法は、それらを別々に埋め込み、2 つの埋め込みからスコアを計算することです (Kiros et al., 2015; Tai et al., 2015)。コマンド `c` と `elek` に対して、次の調味料 `e1` を定義します。...要素上の分布：

$$p(e_i | c) \propto \exp[s(f(c), g(e_i))]$$

ここで、 s はスコアリング関数、 $f(c)$ は c の埋め込み、 $g(e_i)$ は e_i の埋め込みです。以下で説明します。モデルは最大化するようトレーニングされています

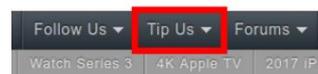
トレーニング データの正しい要素の対数尤度。

コマンド埋め込み。 $f(c)$ を計算するには、 c の各トークンを固定次元に埋め込みます。

ベクトル化して、トークン埋め込みディエン グの平均 2 を取得します。(トークンの埋め込みは GloVe ベクトルで初期化されます。)

要素の埋め込み。 $g(e)$ を計算するには、 e のプロパティを埋め込み、結果を連結し、線形層を適用して $f(c)$ と同じ長さのベクトルを取得します。図2は、モデルが受け取るプロパティの例を示しています。プロパティには次のものがあります。

- テキスト コンテンツ。 e のテキスト コンテンツにコマンド embed der f を適用します。関心のあるほとんどの要素 (リンク、ボタン、および入力) のテキストは短いため、メモリを節約するには、テキストを最初の 10 個のトークンに制限するだけで十分であることがわかります。
- テキスト属性。通常、いくつかの属性 (アリア、タイトル、ツールチップ、プレースホルダー、ラベル、名前) には自然言語が含まれています。それらの値を連結し、結果の文字列にコマンド embed der f を適用します。



```
<a class="dd-head" id="tip-link" href="submit_story/">
  ヒント</a>
```

テキスト コンテンツ: ヒント
us 文字列属性: ヒント リンクヘッド 視覚的特
徴: 位置 = (0.53, 0.08)
目に見える=真

図 2: 要素 e の埋め込み $g(e)$ を計算するために使用されるプロパティの例。

- 文字列属性。句読点とキャメルケースの境界で、トリビュート (タグ、ID、クラス) で他の文字列をトークン化します。次に、それらを個別のルックアップ テーブルで埋め込み、結果のベクトルを平均化します。
- ビジュアル機能。要素の中心の座標 (ページの幅と高さの分数として) と可視性 (ブール値として) で構成されるベクトルを形成します。

採点機能。スコア $s(f(c), g(e))$ を計算するには、まず $f(c)$ と $g(e)$ を 2 つの埋め込みを単位ノルムに正規化した結果とします。次に、連結されたベクトル $[f(c); g(e)]$ に線形レイヤーを適用します。 $g(e); f(c) \circ g(e)$ (ここで、 \circ は要素ごとの積を示します)。

空間的文脈を組み込む。状況によっては、コンテキストが重要です。たとえば、テキスト ボックスを選択するには、隣接するラベル テキストを知っている必要があります、著者に基いて記事を選択するには、著者の名前が近くにあることが必要です。コマンドに基づいてどの関連要素を考慮する必要があるかを特定することは、困難な作業です。

モデルに空間コンテキストを追加して実験します。各方向 $d \in \{\text{上、下、左、右}\}$ に対して、 g を使用して、その方向で e に直接隣接する隣接要素 $nd(e)$ を埋め込みます。

(そのような要素が複数ある場合は、1 つをサンプリングします。そのような要素がない場合は、ゼロ ベクトルを使用します。) 結果を正規化して $g(nd(e))$ を取得した後、 $g(nd(e))$ と f を連結します。 $(c) \circ g(nd(e))$ をスコアリング関数の線形層入力に。

3.3 配置ベースのモデル埋め込みベー

スのモデルの欠点の 1 つは、 c と e のテキスト トークンが直接相互作用しないことです。文の一致に関する以前の研究では、通常、この問題に取り組むために一方または双方向の注意を採用しています (Seo et al., 2016; Yin

ら、2016; Xiong et al., 2017; Yu et al., 2018)。

Hu らと同様の単一のアライメントマトリックスに基づく単純な方法を選択します。(2014)以下のとおり。

$t(e)$ を e のテキスト コンテンツと e のテキスト属性の連結とし、10 個のトークンにトリミングします。

各エントリ $A_{ij}(c, e)$ が c の i 番目のトークンと $t(e)$ の j 番目のトークンの埋め込みの間のドット積である行列 $A(c, e)$ を作成します。次に、行列にサイズ 3×3 の 2 つの畳み込み層を適用し、サイズ 2×2 の最大プーリング層を適用し、タグ埋め込みを連結し、結果に線形層を適用して 10 次元ベクトル $h(c, e)$ 。

$h(c, e)$ に最終的な線形レイヤーを適用してスカラー スコアを計算し、エンコーディング ベースのモデルと同じ目的関数でトレーニングします。

コンテキストを組み込むには、隣接する $nd(e)$ の 4 つのベクトル $h(c, nd(e))$ を最終的な線形層入力に単純に連結します。

4 実験

精度、つまりモデルが正しい要素を選択する例の割合でモデルを評価します。Adam (Kingma and Ba, 2014) を使用して、初期学習率 10^{-3} でニューラル モデルをトレーニングします。

現象セットに基づいて早期停止を適用します。モデルは、レンダリング時にページに表示される任意の要素を選択できます。

実験結果を表2に示します。

どちらのニューラル モデルも、検索モデルよりも大幅に優れています。

4.1 アブレーション解析

Web ページの各タイプの情報の重要性を測定するために、モデルが要素の次の側面の 1 つを観察しないアブレーション研究を実行します。

当然のことながら、表2の結果は、テキスト コンテンツが最も重要な入力信号であることを示しています。

ただし、属性は、埋め込みモデルと配置モデルの両方で重要な役割も果たします。最後に、空間コンテキストは配置モデルのパフォーマンスを向上させますが、ゲインは非常に小さく、適切なコンテキストをモデルに組み込むことは、コンテキストの種類の多様性と信号のまばらさのために困難な作業であることを示唆しています。

| モデルの | 精度 (%) |
|------------|-------------|
| 検索 埋め込み | 56.05 23.62 |
| し テキストなし | 55.43 58.87 |
| 属性なし 空間 | 50.74 15.94 |
| コンテキストの | 48.51 50.66 |
| アライメントなし | |
| テキスト | |
| なし 属性なし | |
| 空間コンテキストなし | |

表 2: モデルとそのアブレーションの精度。

| エラーの種類 | 埋め込み | アライン |
|---------------|-------|------------|
| 文字列の一致に失敗しました | 26.8% | 11.6% 3.8% |
| 文字列を誤って一致させる | 14.2% | 8.9% 7.9% |
| 言い換えを理解できない | 12.1% | 17.4% 6.9% |
| 説明が理解できない | 13.7% | 19.8% |
| 推論を実行できない | 24.8% | 12.7% |
| 目立たない要素を選択する | 10.5% | |
| うるさい注釈 | | |

表 3: 100 例の埋め込みモデルとアラインメント モデルのエラーの内訳。埋め込みモデルは暗黙の説明を適切に処理しますが、アラインメント モデルは文字列の一致に優れています。

4.2 エラー分析

モデルがさまざまな現象をどのように処理するかをよりよく理解するために、少なくとも 1 つのモデルがエラーを起こした 100 の開発例について、埋め込みベースのモデルとアラインメント ベースのモデルの予測を分析します。表 3 に要約されているエラーを以下に詳しく説明します。

文字列の一致に失敗しました。多くのコマンドは、要素のテキスト コンテンツを指定するだけです (たとえば、「カスタマイズされたガレージをクリック」→「カスタマイズされたガレージ、キャンピー、カーポート」というテキストのリンク)。コマンド全体を単一のベクトルとしてエンコードするエンコード モデルは、部分的に一致するテキストを持つ要素の選択に失敗することがあります。対照的に、アラインメント モデルはテキスト マッチングを明示的にモデル化するため、このタイプのコマンドに適しています。

文字列を誤って一致させます。テキスト マッチングに依存しているため、多くの要素がコマンドと部分文字列を共有している場合 (たとえば、多くの要素に「shop」という単語が含まれている場合に「ニットウェアを購入」)、または一致する部分文字列を持つ要素が一致しない場合、配置モデルは苦労します。正しいターゲット (たとえば、「プログラムを入手」→「Microsoft 開発者プログラム」リンクではなく「ダウンロード」リンク)。

説明が理解できない。表 1 に見られるように、多くのコマンドは、要素の言い換え、目的の説明、またはプロパティを使用して要素を間接的に記述します。エンコーディングモデル、

要素のさまざまなプロパティを埋め込みベクトルとして要約し、これらのコマンドの処理にはアライメント モデルよりも優れていますが、難しい例ではいくつかのエラーが発生します (たとえば、「この通知を閉じてください」→「X」非表示のテキスト「非表示」のボタン)。

推論を実行できません。ほとんどの場合、モデルはリレーショナル、序数、または空間推論を処理できません。最も頻繁に発生するエラー モードは、要素がテキスト ボックスであり、コマンドが近くのラベルを参照として使用する場合があります。一部のテキスト ボックスには、モデルが使用できるセマンティック アノテーション (ツールチップや aria 属性など) がありますが、多くの Web ページではそのようなアノテーションが提供されていません。

これらのケースを処理するには、モデルは、論理的または視覚的なコンテキストに基づいてテキスト ボックスのラベルを識別する必要があります。

その他のエラー。注釈ノイズとは別に、Web ページ上の複数の要素が特定のコマンドを満たす場合があります (たとえば、「ログイン」は、Web ページの任意の「サインイン」ボタンと一致する場合があります)。このような場合、注釈は通常、可能な候補の中で最も顕著な要素を示します。

ユーザーに自然なインターフェイスを提供するために、モデルは間違いなく、あいまいな要素ではなく、そのような顕著な要素を予測することを学習する必要があります。

5 関連する仕事と議論

自然言語を行動にマッピングする。セマンティック解析に関する以前の研究では、さまざまな環境で自然言語の発話によって記述されたアクションを実行することを学習しました。このようなアクションの例には、API 呼び出し (Young et al., 2013; Su et al., 2017; Bordes and Weston, 2017)、データベース クエリ (Zelle and Mooney, 1996; Zettlemoyer and Collins, 2007; Berant et al., 2017). 2013; Yih et al., 2015)、ナビゲーション (Artzi and Zettlemoyer, 2013; Janner et al., 2018)、オブジェクト操作 (Tellex et al., 2011; Andreas and Klein, 2015; Guu et al., 2017年; フリードら、2018年)。

Web ページとグラフィカル ユーザー インターフェイスについては、自然言語を使用して Web テーブルで計算を実行し (Pasupat and Liang, 2015; Zhong et al., 2017)、Web フォームを送信する (Shi et al., 2017) という以前の研究があります。私たちのタスクは、ユーザー インターフェイスの指示を解釈する以前の研究と似ています (Branavan et al., 2009, 2010; Liu et al., 2018)。彼らの研究は遠く離れた監督から学ぶことに焦点を当てていますが、私たちはより浅い相互作用を考えていますが、はるかに広い領域について考えています。

以前の研究では、オブジェクトの自然言語記述を生成するという逆の問題も調査しています (Vinyals et al., 2014; Karpathy and Fei Fei, 2015; Zariaif and Schlangen, 2017)。私たちのデータセットが、Web ページでのアクションを記述するという逆のタスクを探索するのにも役立つことを願っています。

参考ゲーム。参照ゲームでは、システムは与えられた発話によって参照される正しいオブジェクトを選択する必要があります (Frank and Goodman, 2012)。

参照ゲームに関するこれまでの研究では、同様の特性を持つ少数のオブジェクトに焦点を当て、語用論を適用してあいまいな発話を処理しました (Golland et al., 2010; Smith et al., 2013; Çe likyilmaz et al., 2014; Andreas and Klein, 2016; Yu ら、2017)。私たちのタスクは、いくつかの課題を伴う参照ゲームと見なすことができます。より多くのオブジェクト、多様なオブジェクト プロパティ、コンテキストに基づいてオブジェクトを解釈する必要性などです。

Web ページとの対話。自動化されたスクリプトは、Web 要素と対話するために使用されます。その間

ほとんどのスクリプトは論理セレクター (CSS や XPath など) を使用して要素を参照しますが、画像 (Yeh et al., 2009) や単純な自然言語の発話 (Soh, 2017) など、いくつかの代替手段がありました。

Web ページをナビゲートするためのその他のインターフェイスには、キーストローク (Spalteholz et al., 2008)、音声 (Ashok et al., 2014)、触覚 (Yu et al., 2005)、視線 (Kumar et al., 2007) などがあります。 . . .

6 結論

私たちは、オープンエンドで半構造化された Web ページで自然言語コマンドを接地するという新しいタスクを提示しました。要素を参照するさまざまな方法、テキスト要素属性と非テキスト要素属性の混合、およびコンテキストを適切に組み込む必要性により、私たちのタスクは、現実世界のアプリケーションに大きな可能性を秘めた言語理解のための挑戦的な環境を提供します。

データセットとコードは <https://github.com/stanfordnlp/phrasenode>.

https の CodaLab プラットフォームで利用可能な再現可能な実験:

[//worksheets.codalab.org/worksheets/0x009f7249cd944284a81af331093c3579/](https://worksheets.codalab.org/worksheets/0x009f7249cd944284a81af331093c3579/).

謝辞

この作品は、No. IIS-1552635 とアマゾンの下で NSF キャリア賞によってサポートされていました。

研究賞。

参考文献

- J. アンドレアスとD. クライン。 2015. 命令追従のためのアラインメントベースの構成セマンティクス。自然言語処理における経験的方法 (EMNLP)。
- J. アンドレアスとD. クライン。 2016. ニューラル リスナーとスピーカーを使用した語用論に関する推論。自然言語処理における経験的方法 (EMNLP)、ページ 1173-1182。
- Y. Artzi と L. Zettlemoyer。 2013. 命令をアクションにマッピングするためのセマンティック パーサーの弱教師あり学習。計算言語学協会 (TAACL) のトランザクション、1:49-62。
- V. Ashok, Y. Borodin, S. Stoyanchev, Y. Puzis, および IV Ramakrishnan。 2014. 視覚障害者向けの音声駆動型 Web ブラウジング インターフェイスのオズの魔法使いによる評価。Web for All Conference で。
- J. Berant, A. Chou, R. Frostig, および P. Liang。 2013. 質問と回答のペアからの Freebase での意味解析。自然言語処理における経験的方法 (EMNLP)。
- A. Bordes と J. Weston。 2017. エンドツーエンドの目標指向の対話を学習。学習表現に関する国際会議 (ICLR)。
- S. Branavan, H. Chen, L. S. Zettlemoyer, および R. Barzilay が横たわっていました。 2009. 命令から行動へのマッピングのための強化学習。Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL IJCNLP)、82 ~ 90 ページ。
- S. Branavan, L. Zettlemoyer, および R. Barzilay。 2010年。行間を読む: 高レベルの命令をコマンドにマッピングする方法を学びます。Association for Computational Linguistics (ACL)、1268 ~ 1277 ページ。
- A. チェリキルマズ, Z. フェイゾラヒ, DZ ハッカニ-トゥール, R. サリカヤ。 2014. 自然なユーザー インターフェイスのための会話ダイアログでの参照表現の解決。自然言語処理における経験的方法 (EMNLP)。
- DL チェンと RJ ムーニー。 2011. 観察から自然言語によるナビゲーションの指示を解釈することを学ぶ。Association for the Advancement of Artificial Intelligence (AAAI)、859 ~ 865 ページ。
- M. フランクと ND グッドマン。 2012. 言語ゲームにおける実用的な推論の予測。科学、336:998-998。
- D. フリード, J. アンドレアス, D. クライン。 2018. 命令を生成して従うための統一された実用モデル。北米計算言語学協会 (NAACL)。
- D. ゴランド, P. リャン, D. クライン。 2010. 空間記述を生成するためのゲーム理論的アプローチ。自然言語処理における経験的手法 (EMNLP)、410 ~ 419 ページ。
- K. Guu, P. Pasupat, EZ Liu, および P. Liang。 2017年。言語からプログラムへ: 強化学習と最大限界可能性の橋渡し。計算言語学協会 (ACL)。
- M. Hammoudi, G. Rothermel, および P. Tonella。 2016年。Web アプリケーションの記録/再生テストが壊れるのはなぜですか? ソフトウェアのテスト、検証、検証に関する IEEE 国際会議。
- B. Hu, Z. Lu, H. Li, および Q. Chen。 2014. 自然言語の文を照合するための畳み込みニューラル ネットワーク アーキテクチャ。神経情報処理システム (NIPS) の進歩。
- M. Janner, K. Narasimhan, および R. Barzilay。 2018. 根拠のある空間的推論のための表現憤慨学習。計算言語学協会 (TAACL) のトランザクション、6。
- A. カルパシーとL. フェイフェイ。 2015. 画像の説明を生成するための深い視覚的セマンティック アラインメント。Computer Vision and Pattern Recognition (CVPR) の 3128 ~ 3137 ページ。
- D. キングマとJ. バ。 2014. Adam: A method arXiv プレプリント 確率的最適化のために。arXiv:1412.6980。
- R. Kiros, Y. Zhu, R. Salakhutdinov, RS Zemel, R. Ur tasun, A. Torralba, および S. Fidler。 2015. スキップ思考ベクトル。神経情報処理システム (NIPS) の進歩。
- M. Kumar, A. Paepcke, および T. Winograd。 2007. アイポイント: 視線とキーボードを使用した実用的なポインティングと選択。コンピューティング システムにおけるヒューマン ファクターに関する会議 (CHI)。
- EZ Liu, K. Guu, P. Pasupat, T. Shi, および P. Liang。 2018. 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。学習表現に関する国際会議 (ICLR)。
- DK ミスラ, K. タオ, P. リャン, A. サクセナ。 2015年。高レベルの命令のための環境主導のレキシコン誘導。計算言語学協会 (ACL)。
- P. バスパットとP. リャン。 2015. 半構造化テーブルの合成セマンティック解析。計算言語学協会 (ACL)。
- M. Seo, A. Kembhavi, A. Farhadi, および H. Hajishirzi。 2016. 機械理解のための双方向注意フロー。arXiv。
- T. Shi, A. Karpathy, L. Fan, J. Hernandez, および P. Liang。 2017. World of Bits: 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 (ICML)。
- NJ スミス, ND グッドマン, MC フランク。 2013年。他のエージェントについての再帰的な実用的な推論による言語の学習と使用。Advances in Neural Information Processing Systems (NIPS) の 3039 ~ 3047 ページ。

- K.ソー。2017. TagUI: ユーザー インタラクシオンを自動化するための RPA / CLI ツール。 <https://github.com/kelaberetiv/TagUI> .
- L. Spalteholz, KF Li, N. Livingston, および F. Hamidi。 2008. Keysurf: 身体障害者向けの文字制御ブラウザ。ワールド ワイド ウェブ (WWW)。
- Su Y, Awadallah AH, Khabasa M, Pantel P, Ga mon M, および Incarnation MJ。 2017. Web ミツパチへの自然言語インターフェースの構築。情報と知識の管理に関する会議 (CIKM)。
- KS Tai, R. Socher, および CD Manning。 2015年。 ツリー構造の長短期記憶ネットワークからの意味表現の改善。計算言語学 (ACL) のための協会として。
- S. Tellex, T. Kollar, S. Dickerson, MR Walter, AG Banerjee, SJ Teller, N. Roy。 2011. ロボット ナビゲーションとモバイル操作のための自然言語コマンドの理解。 Association for the Advancement of Artificial Intelligence (AAAI)。
- O. Vinyals, A. Toshev, S. Bengio, および D. Erhan。 2014年。 Show and Tell: ニューラル イメージ キャプション ジェネレーター。 arXiv プレプリント arXiv:1411.4555.
- C. Xiong, V. Zhong, および R. Socher。 2017. 質問応答のための動的コーディングテンションネットワーク。学習表現に関する国際会議 (ICLR)。
- T. Yeh, T. Chang, および R. Miller。 2009. Sikuli: 検索と自動化に GUI スクリーンショットを使用。ユーザー インターフェイス ソフトウェアとテクノロジ (UIST)。
- W. Yih, M. Chang, X. He, および J. Gao。 2015. 段階的なクエリ グラフ生成によるセマンティック解析: ナレッジ ベースによる質問応答。計算言語学協会 (ACL)。
- W. Yin, H. Schütze, B. Xiang, および B. Zhou。 2016年。 ABCNN: センテンス ペアをモデル化するための注意ベースの畳み込み ニューラル ネットワーク。計算言語学協会 (TACL) のトランザクション、4.
- S. ヤング, M. ガシッチ, B. トムソン, JD ウィリアムズ。 2013. POMDP ベースの統計的音声対話システム: レビュー。 IEEE の議事録、5、1160 ~ 1179 ページ。
- AW Yu, D. Dohan, M. Luong, R. Zhao, K. Chen, M. Norouzi, および QV Le。 2018. QANet: 読解力のためのグローバルな自己注意とローカル畳み込みを組み合わせた。学習表現に関する国際会議 (ICLR)。
- L. Yu, H. Tan, M. Bansal, および TL Berg。 2017. 表現を参照するための話者 - 聞き手 - 強化子の共同モデル。コンピュータ ビジョンとパターン認識 (CVPR)。
- W. Yu, R. Kuber, E. Murphy, P. Strain, および G. McAlister。 2005. 視覚障害者の Web アクセシビリティを改善するための新しいマルチモーダル インターフェイス。バーチャル リアリティ 9。
- M. ザジチェック, C. パウエル, C. リーブス。 1998 年。視覚障害者向けの Web ナビゲーション ツール。支援技術に関する国際 ACM 会議。
- S. Zarrabi & D. Schlangen。 2017. 視覚情報と分布情報から参照語の意味を取得する: オブジェクトの命名に関する実験。計算言語学 (ACL) のための協会として。
- M. ゼルと RJM ーニー。 1996. 帰納的論理プログラム ming を使用してデータベース クエリを解析することを学ぶ。 Association for the Advancement of Artificial Intelligence (AAAI)、1050 ~ 1055 ページ。
- LS Zettlemoyer と M. Collins。 2007. 論理形式への構文解析のための緩和された CCG 文法のオンライン学習。自然言語処理における経験的方法と計算による自然言語学習 (EMNLP/CoNLL)、678 ~ 687 ページ。
- V. Zhong, C. Xiong, および R. Socher。 2017. Seq2sql: 強化学習を使用した自然言語からの構造化クエリの生成。 arXiv プレプリント arXiv:1709.00103.