物体指示理解タスクにおける クロスモーダル言語生成に基づくデータ拡張

○飯田紡 *， 九曜克之 *，石川慎太朗，杉浦孔明（慶應義塾大学）

1. はじめに

高齢化が進展する現代社会において，在宅介護者の不足は深刻な社会問題となっている．このような社会問題の解決策として，物理的に支援可能な生活支援ロボットに注目が集まっている．しかしながら，生活支援ロボットが人間と自然な対話をする能力は現状不十分である．そこで，本研究では生活支援ロボットにおける物体指示理解タスクを扱う．

本研究の目的は，物体の把持命令文が与えられたうえで正しく命令内容を解釈し，対象物体を特定することである．例えば，図 2 左図の状態において “Move the yellow container to the top left box.” という命令文が与えられたうえで，右下の箱内にある黄色い容器を対象物体として予測することが望ましい．

ただし，人間の発する命令にはしばしば曖昧性が含まれ，正確な内容の理解は容易ではない．上述した把持命令文においても，黄色い容器が複数ある場合，対象の容器を特定するためには正確な参照表現の解釈が必要である．このような参照表現の含まれる命令文では対象物体の特定が難しく，実際に誤るケースが報告されている [1].

上記のタスクを扱った既存研究として，[1–3] などがある．[2] では UNITER [4] に基づく手法が提案され，[1] を超える精度が報告されている．一方，[2] では正例と負例のサンプル数を等量にするために大量の負例サンプルを使用していなかった．そのために，サンプル数が少ない場合に精度が低いという問題がある．

そこで，本研究ではクロスモーダル逆翻訳データ拡張手法を提案する．図 1 に提案手法の概略を示す．提案手法は Target-dependent UNITER [2] とは異なり，クロスモーダル逆翻訳によるデータ拡張を導入する．クロスモーダル逆翻訳においては，まず生成モジュールを用いて候補物体の画像から命令文を生成する．次に，理解モジュールによって生成した命令文から候補物体を特定する．この処理により，候補物体が正しく特定された命令文のみを訓練集合に追加することで，データ拡張を行う．データ拡張により正例サンプルを増加させることで，これまで使用されていなかった負例サンプルを訓練集合に加えることも可能である．訓練集合のサンプル数が増加することで，理解モジュールの汎化性能が高まることが見込まれる．

本研究の独自性は以下である．

- Case Relation Transformer を用いて画像から命令文を生成し，正例に関してデータ拡張を行う．

*両者は同等に貢献した.

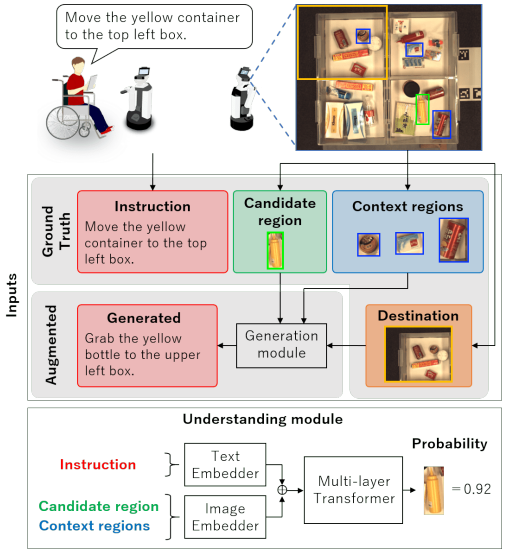

図 1 提案手法の概略.

2. 関連研究

マルチモーダル言語処理分野の代表的なサーベイ論文として [5] が挙げられる．[5] では問題の定式化，手法，データセット，評価方法について議論し，対応する最先端手法との結果の比較を行っている．マルチモーダル言語処理分野は，扱うモダリティの組み合わせにより様々な分野に分かれる．言語と画像を扱う分野には Visual Question Answering，Visual Referring Expression, Vision-and-Language Navigation 等がある．

CrossMap Transformer [6] は Vision-and-Language Navigation タスクを扱ったモデルである．CrossMap Transformer では，言語・画像・行動間の関係が Transformer を用いてモデル化され，double back translation を用いてデータ拡張が行われる．Case Relation Transformer [7] は，画像と画像中の物体・移動先を入力として，物体操作命令文を生成するモデルである．Case Relation Transformer は画像中の物体間の位置関係をモデル化し，参照表現を含む命令文が生成できる．

Visual Referring Expression 分野における有名なデータセットに関して，実画像を用いるものとしては，ReferCLEF [8]，RefCOCO [8]，GuessWhat?! [9] 等が挙げられ，合成画像を用いるものとしては，CLEVR-Ref+ [10] が挙げられる．物体指示理解タスクのデータセットとしては，PFN-PIC [3] と WRS-PV [11] が挙げられる．

3. 問題設定

本論文では以下のように用語を定義する．

- **対象物体/領域:** 命令文中で参照される対象物体/バウンディングボックス
- **候補物体/領域:** 対象物体かどうかを推定する物

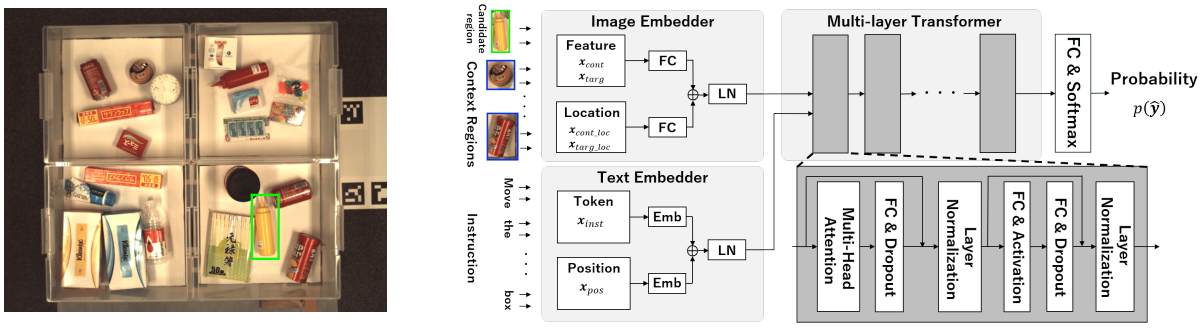

図 2 左：MLU-FI タスクにおける対象物体特定の例．緑のバウンディングボックスは予測対象物体（yellow container）である．右：提案手法における理解モジュールのネットワーク構造．FC，LN，Emb はそれぞれ全結合層，正規化層，埋め込み層を表す．

体/バウンディングボックス

- **コンテキスト物体/領域:** 物体検出器で検出された，候補物体以外の各物体/バウンディングボックス

本論文では，把持命令文と画像，候補物体が与えられたうえで，候補物体が対象物体かどうかを二値分類するタスクを扱う．本タスクを Multimodal Language Understanding for Fetching Instruction (MLU-FI) と定義する．図 2 左図に本タスクの例を示す．本タスクでは，図 2 左図に示す画像と図中に緑のバウンディングボックスで示された候補物体，“Move the yellow container to the top left box.” という命令文が与えられたうえで，候補物体が対象物体かどうかを予測する．本タスクの出力は，候補領域が対象領域である確率の予測値 $p(\hat{\boldsymbol{y}})$ である．$y^* = \mathrm{argmax}_{\hat{\boldsymbol{y}}}\, p(\hat{\boldsymbol{y}})$ として，候補物体と対象物体が一致しているときは $y^* = 1$，異なるときは $y^* = 0$ と出力することが望ましい．

タスクの評価尺度には分類精度 Acc を用いる．本論文では，物体検出誤りが十分に少ないことを前提とする．

4. 手法

提案手法は理解モジュールと生成モジュールの 2 つから構成される．理解モジュールは Target-dependent UNITER を拡張したものである．また，生成モジュールとして Case Relation Transformer を用いる．Target-dependet UNITER は UNITER に候補領域を入力する機構を加えて拡張したモデルである．提案手法の新規性は以下である．

- Target-dependent UNITER とは異なり，生成モジュールとして Case Relation Transformer を導入して，データ拡張を行うことができる．
- コンテキスト領域のうち，入力に含める領域数を，候補領域に近い順上位 N_{prox} 個までに制限する．このとき，距離として各領域の中心座標間のユークリッド距離を用いる．

4.1 理解モジュール

理解モジュールのネットワーク構造を図 2 右図に示す．図において，Instruction は命令文，Context Regions はコンテキスト領域群，Candidate Region は候補領域を表す．理解モジュールは大きく分けて Image Embedder, Text Embedder, Multi-layer Transformer という 3 つのモジュールから構成される．

ネットワークの入力 $\boldsymbol{x}$ を以下のように定義する．

$$\boldsymbol{x} = \{\boldsymbol{X}_{\mathrm{inst}}, \boldsymbol{X}_{\mathrm{cont}}, \boldsymbol{X}_{\mathrm{targ}}\} \tag{1}$$

ここに，$\boldsymbol{X}_{\mathrm{inst}}$ は命令文，$\boldsymbol{X}_{\mathrm{cont}}$ はコンテキスト領域群，$\boldsymbol{X}_{\mathrm{targ}}$ は候補領域を表す．さらに，

$$\boldsymbol{X}_{\mathrm{inst}} = \{\boldsymbol{x}_{\mathrm{inst}}, \boldsymbol{x}_{\mathrm{pos}}\} \tag{2}$$

$$\boldsymbol{X}_{\mathrm{cont}} = \left\{\boldsymbol{x}_{\mathrm{cont}(i)}, \boldsymbol{x}_{\mathrm{contloc}(i)}\right\} \tag{3}$$

$$\boldsymbol{X}_{\mathrm{targ}} = \{\boldsymbol{x}_{\mathrm{targ}}, \boldsymbol{x}_{\mathrm{targloc}}\} \tag{4}$$

である．ここに，$\boldsymbol{x}_{\mathrm{inst}}$ は命令文のトークン列，$\boldsymbol{x}_{\mathrm{pos}}$ は命令文中の各トークンの位置，$\boldsymbol{x}_{\mathrm{cont}(i)}$ は i 番目の物体のコンテキスト領域，$\boldsymbol{x}_{\mathrm{contloc}(i)}$ は i 番目のコンテキスト領域の位置情報，$\boldsymbol{x}_{\mathrm{targ}}$ は候補領域，$\boldsymbol{x}_{\mathrm{targloc}}$ は候補領域の位置情報を表す．ただし，$i = 1, \cdots, \min(N_{\mathrm{FRCNN}} - 1, N_{\mathrm{prox}})$ である．また，N_{FRCNN} は Faster R-CNN によって検出した画像中の領域の数，N_{prox} は $\boldsymbol{X}_{\mathrm{targ}}$ と近い順に上位何個まで $\boldsymbol{X}_{\mathrm{cont}}$ に含めるかを表すパラメータである．

命令文は，WordPiece によるトークン化を行い，単語埋め込みと位置埋め込みを足し合わせることによって $\boldsymbol{x}_{\mathrm{inst}} \in \mathbb{R}^{768}$ を得る．画像特徴量は，まず全体画像を Faster R-CNN に入力して各領域の特徴量 $\boldsymbol{x}_{\mathrm{img}}$ を獲得する．$\boldsymbol{x}_{\mathrm{targ}}$ は，$\boldsymbol{x}_{\mathrm{img}}$ から判定対象とする領域を選択する．また，それ以外の領域を $\boldsymbol{x}_{\mathrm{cont}}$ とする．これらの処理により，$\boldsymbol{x}_{\mathrm{targ}}, \boldsymbol{x}_{\mathrm{cont}} \in \mathbb{R}^{768}$ を得る．$\boldsymbol{x}_{\mathrm{contloc}}, \boldsymbol{x}_{\mathrm{targloc}}$ は，$[x_1, y_1, x_2, y_2, x_2 - x_1, y_2 - y_1, (x_2 - x_1) \times (y_2 - y_1)]$ である．ここに，バウンディングボックスの左上の頂点の座標を (x_1, y_1)，右下の座標を (x_2, y_2) とする．

図 2 右図に示すように，理解モジュールにおける Image Embedder は 2 つの全結合層と正規化層から構成され，$(\boldsymbol{x}_{\mathrm{targ}}, \boldsymbol{x}_{\mathrm{targloc}})$ または $(\boldsymbol{x}_{\mathrm{cont}}, \boldsymbol{x}_{\mathrm{contloc}})$ を入力して画像中の領域群の埋め込みを行う．Text Embedder は 2 つの埋め込み層と正規化層から構成され，命令文の埋め込みを行う．入力は $\boldsymbol{x}_{\mathrm{inst}}$ と $\boldsymbol{x}_{\mathrm{pos}}$ である．

図において，Multi-layer Transformer は Transformer を複数重ねたものである．入力は 2 つの Embedder の出力を結合したもので，出力は候補領域が対象領域である確率の予測値 $p(\hat{\boldsymbol{y}})$ である．画像と命令文の特徴量を結合してから Transformer に入力することで，画像中の領域群とトークンの関係がモデル化できる．損失関数にはクロスエントロピー関数を使用する．

4.2 生成モジュールによるデータ拡張

提案手法では，データ拡張を行うために生成モジュールを用いて命令文を生成する．生成モジュールの構造は [7] を参照されたい．

生成モジュールに対象領域とコンテキスト領域群，目標領域を入力して，命令文 $\tilde{\boldsymbol{x}}_{\text{inst}}$ を生成する．次に，$\tilde{\boldsymbol{x}}_{\text{inst}}$ を理解モジュールに入力し，出力 $p\left(\hat{\boldsymbol{y}}|\tilde{\boldsymbol{x}}_{\text{inst}}\right)$ が閾値 θ 以上の $\tilde{\boldsymbol{x}}_{\text{inst}}$ を収集する．上記より，データ拡張で得られるデータ集合 $\tilde{\boldsymbol{X}}_{\text{aug}}$ は以下のように表される．

$$\tilde{\boldsymbol{X}}_{\text{aug}} = \left\{ \tilde{\boldsymbol{x}}_{\text{inst}}^{(i)} | p(\hat{\boldsymbol{y}}^{(i)} | \tilde{\boldsymbol{x}}_{\text{inst}}^{(i)}) \geq \theta \right\} \tag{5}$$

ここに，i はインデックスである．

5. 実験設定

5.1 データセット

MLU-FI タスクで標準的に使用されている PFN-PIC [3] を用いて，提案手法を評価した．PFN-PIC は，画像および画像中の物体に関する命令文から構成される標準データセットである．

本実験では，Faster R-CNN [12] を用いて各画像から複数の領域を抽出した．Faster-RCNN の事前学習，fine-tuning にはそれぞれ ImageNet [13]，PFN-PIC [3] を使用した．Faster R-CNN が検出した領域のうち，真の対象領域との Intersection over Union (IoU) が 0.7 以上のものを正例サンプルとし，0.3 以下のものを負例サンプルとした．さらに，データセット内で正例と負例のサンプル数を等量にするため，負例のサンプル集合から正例サンプルと同じ数だけ無作為に選択し，正例のサンプル集合に加えてデータセットとした．拡張したデータセットも同様の処理を行った．

本実験では，上述の処理により，訓練集合を 63330 文，検証集合を 710 文，テスト集合を 612 文とした．訓練集合 63330 文のうち，使用する命令文数を N_{GT} として，$N_{\text{GT}} = 4000, 6000, 10000, 63330$ の場合について実験を行った．訓練集合を fine-tuning に，検証集合をハイパーパラメータを決定するために使用した．また，テスト集合をモデルの評価に使用した．

5.2 パラメータ設定

ネットワーク内の Transformer は，層数が 2，隠れ層の次元数が 768，Attention の Head 数が 12 とした．最適化には AdamW を使用し，学習率は 8×10^{-5}，ステップ数は 20000，バッチサイズは 8，ドロップアウト率は 0.1 とした．なお，1 ステップは 1 つのバッチの処理を意味する．N_{prox} は 20，θ は 0.999 とした．

生成モジュールのパラメータ数は 5900 万である．また，理解モジュールにおける事前学習のパラメータ数は 4200 万，そのうち fine-tuning に用いるパラメータ数は 3900 万である．20000 ステップの学習を行い，2000 ステップごとに検証集合およびテスト集合による評価を行った．検証集合において損失関数の値が最も低いときのテスト集合における精度を，最終的な精度とした．

学習にはメモリ 11GB 搭載の GeForce RTX 2080 および Intel Core i9-9900K を使用した．学習に要した時間は，事前学習に 3 時間，fine-tuning に 30 分であった．

6. 実験結果

6.1 定量的結果

ベースラインと提案手法の比較結果を図 3 に示す．MLU-FI タスクにおいて良好な結果が報告されている Target-dependent UNITER をベースラインとした．生成モジュールを用いて生成した命令文全体のうち，データセットに加える数を N_{aug} とする．図 3 において，$N_{\text{aug}} = 0$ がベースラインと同等である．なお，生成モジュールの学習セットサイズには注意が必要である．例えば，$N_{\text{GT}} = 4000$ の場合に生成モジュールの学習セットサイズについて $N_{\text{GT}} > 4000$ とすると理解モジュールの学習において $N_{\text{GT}} > 4000$ としたことと同様の効果がある．そのため，生成モジュールの学習セットサイズも N_{GT} と統一した．

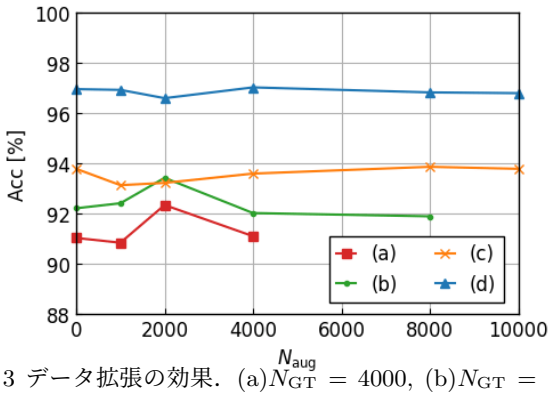

図 3 データ拡張の効果．(a)$N_{\text{GT}} = 4000$, (b)$N_{\text{GT}} = 6000$, (c)$N_{\text{GT}} = 10000$, (d)$N_{\text{GT}} = 63330$.

表 1 Ablation Study の定量的結果．5 回実験を行った結果の平均および標準偏差を示す．

Acc [%]		N_{prox}	
		20	N_{FRCNN}
(i)	$N_{\text{GT}} = 4000$	92.4 ± 0.7	91.7 ± 0.9
(ii)	$N_{\text{GT}} = 6000$	93.4 ± 0.6	93.2 ± 0.5
(iii)	$N_{\text{GT}} = 10000$	93.2 ± 0.5	93.7 ± 0.5
(iv)	$N_{\text{GT}} = 63330$	96.6 ± 1.1	97.1 ± 0.3

図 3 より，$N_{\text{GT}} = 4000$ において，$N_{\text{aug}} = 2000$ のとき精度が最大になった．$N_{\text{GT}} = 6000$ についても同様の傾向がみられた．このことは，N_{GT} が少ないときにはデータ拡張により精度が向上したことを示唆している．すなわち，既存手法と比較して提案手法が優れるという結果を得た．

一方，$N_{\text{GT}} = 10000$ では $N_{\text{aug}} = 8000$ で精度が最大になったが，ベースラインとほぼ同等の精度であった．$N_{\text{GT}} = 63330$ についても同様の傾向がみられた．このことは，N_{GT} が多いときにはデータ拡張が精度に与える影響が小さいことを示唆している．

6.2 Ablation Study

Ablation Study として，以下の 2 条件を定めた．

(i) データ拡張: $N_{\text{aug}} = 1000, 2000, 4000, 8000, 10000$ の場合について，性能への影響を調べた．

(ii) Proximity: $N_{\text{prox}} = 20, N_{\text{FRCNN}}$ の場合について，性能への影響を調べた．

図 3 に上記 (i) に関する Ablation Study の定量的結果を示す．図 3 より，精度が最も高い N_{aug} のときと比較して，N_{aug} をさらに増やすと全ての N_{GT} で精度が低下した．また，N_{aug} が N_{GT} の約半数を超えると精度は低下し，分散も大きくなった．これより，適切な量のデータ拡張を行うことが，モデルの性能向上に寄与していると考えられる．

図 4 TP（左），FP・MO（右）の例．緑色で囲まれている領域が候補領域であり，青色で囲まれている領域が対象領域である．左:“move the black coffee mug to the upper left box”．右：“move the packing plastic with the yellow head and put it in the lower right box”．

表 2　失敗例の分類

エラー ID	詳細	#Errors
OOV	命令文の分割失敗	7
CE	視覚情報や言語情報の処理におけるエラー	6
MO	候補領域が複数の物体を含む	2
OE	その他のエラー	3

また，表 1 に上記 (ii) に関する Ablation Study の定量的結果を示す．表 1 における N_{FRCNN} は，Faster R-CNN によって検出した画像中の領域の数を表す．$N_{\mathrm{GT}} = 4000, 6000$ の際は $N_{\mathrm{prox}} = N_{\mathrm{FRCNN}}$ とすることでそれぞれ 0.7, 0.2 ポイント精度が低下した．一方で，$N_{\mathrm{GT}} = 10000, 63330$ では $N_{\mathrm{prox}} = N_{\mathrm{FRCNN}}$ とすることでどちらも 0.5 ポイント精度が向上した．したがって，N_{GT} が小さいときにはコンテキスト領域の領域数を制限することがモデルの性能向上に寄与していると考えられる．

6.3　定性的結果

定性的結果を図 4 に示す．ここに，TP は True Positive，FP は False Positive，FN は False Negative，TN は True Negative を表す．

図 4 の左図は TP の例である．対象物体は右下の区画にある黒色のカップである．候補領域について，$p(\hat{\boldsymbol{y}}) = 0.999$ と出力しており，ほぼ正確に候補領域が対象領域だと判定できていることがわかる．

図 4 の右図は FP の例である．対象物体は右上の区画にあるパッケージである．候補領域は明らかに対象領域とは異なる領域を示しているにもかかわらず，$p(\hat{\boldsymbol{y}}) = 0.997$ と出力しており，候補領域が対象領域であると判定してしまっていることがわかる．

6.4　エラー分析

$N_{\mathrm{GT}} = 63330$ の際に最も精度の高かった $N_{\mathrm{aug}} = 4000$ において，TP は 301 サンプル，FP は 13 サンプル，FN は 5 サンプル，TN は 293 サンプルであった．すなわち，失敗例は合計 18 サンプルであった．

失敗例を人手で分析した結果を表 2 に示す．失敗の原因は，大きく分けて OOV，CE，MO，OE の 4 種類であった．OOV は，命令文が語彙外の単語を含むケースである．例えば，“rectangle” という単語は (“re”, “##ct”, “##ang”, “##le”) と分割されていた．CE は，モデルが視覚情報や言語情報の処理に失敗したケースである．MO は，候補物体と関係ない物体の画素が候補領域に多く含まれているケースである．MO のケースでは候補物体が遮蔽されていることが多かった．図 4 の右図に MO の例を示す．この例は，候補領域が候補物体の近くにある物体を含んでいる．OE は，上記に含まれないケースである．

OOV のケースは，語彙数を増加させることで低減できると考えられる．また，セマンティックセグメンテーションを用いることで MO のケースを減らすことができると考えられる．CE のケースは，CLIP [14] などを用いてマルチモーダルな理解モデルを構築することで解決できると考えられる．

7.　結論

本論文では，クロスモーダル言語生成に基づくデータ拡張手法を提案し，Target-dependent UNITER に適用した．本研究の貢献は以下である．

- Case Relation Transformer を用いて生成した命令文を使用し，正例に関してデータ拡張を行った．
- 標準データセット PFN-PIC において，ベースライン手法を分類精度で上回った．

謝辞

本研究の一部は，JSPS 科研費 20H04269，JST CREST，JST ムーンショット型研究開発事業 JPMJMS2011，NEDO の助成を受けて実施されたものである．

参　考　文　献

[1] A. Magassouba, et al., “A Multimodal Target-Source Classifier With Attention Branches to Understand Ambiguous Instructions for Fetching Daily Objects,” RA-L, vol.5, no.2, pp.532–539, 2020.
[2] S. Ishikawa and K. Sugiura, “Target-dependent uniter: A transformer-based multimodal language comprehension model for domestic service robots,” IROS, 2021.
[3] J. Hatori, Y. Kikuchi, S. Kobayashi, et al., “Interactively Picking Real-World Objects with Unconstrained Spoken Language Instructions,” ICRA, pp.3774–3781, 2018.
[4] Y.-C. Chen, L. Li, L. Yu, et al., “Uniter: Universal image-text representation learning,” ECCV, pp.104–120, 2020.
[5] A. Mogadala, et al., “Trends in integration of vision and language research: A survey of tasks, datasets, and methods,” arXiv preprint arXiv:1907.09358, 2020.
[6] A. Magassouba, et al., “Crossmap transformer: A crossmodal masked path transformer using double back-translation for vision-and-language navigation,” RA-L, 2021.
[7] M. Kambara and K. Sugiura, “Case relation transformer: A crossmodal language generation model for fetching instructions,” IROS, 2021.
[8] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, “ReferItGame: Referring to Objects in Photographs of Natural Scenes,” EMNLP, pp.787–798, 2014.
[9] H. deVries, F. Strub, S. Chandar, et al., “Guesswhat?! visual object discovery through multi-modal dialogue,” CVPR, pp.5503–5512, July 2017.
[10] R. Liu, C. Liu, Y. Bai, and A.L. Yuille, “Clevr-ref+: Diagnosing visual reasoning with referring expressions,” CVPR, pp.4185–4194, June 2019.
[11] T. Ogura, et al., “Alleviating the burden of labeling: Sentence generation by attention branch encoder-decoder network,” RA-L, vol.5, no.4, pp.5945–5952, 2020.
[12] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: towards real-time object detection with region proposal networks,” Trans. PAMI, vol.39, no.6, pp.1137–1149, 2016.
[13] J. Deng, W. Dong, R. Socher, et al., “Imagenet: A large-scale hierarchical image database,” CVPR, pp.248–255, 2009.
[14] A. Radford, J.W. Kim, C. Hallacy, et al., “Learning transferable visual models from natural language supervision,” CoRR, 2021.