常識知識グラフを活用した動作ラベル間の等価関係の自動構築手法

三辻 史哉 [1,2]　チャクラボルティ シュデシナ [1]　森田 武史 [1,2]　吉川 友也 [3]
山本 泰智 [4]　太田 葵 [2]　江上 周作 [2]　鵜飼 孝典 [2,5]　濱崎 雅弘 [2]
[1] 青山学院大学 [2] 産業技術総合研究所 [3] 千葉工業大学
[4]ROIS-DS ライフサイエンス統合データベースセンター [5] 富士通株式会社
morita@it.aoyama.ac.jp masahiro.hamasaki@aist.go.jp

概要

動作認識モデルの汎化性能を向上させるためには，多様な動画を用いた学習が重要である．しかし，データセットごとに動作ラベルの定義が異なるため，複数データセットの統合利用が困難であり，汎化性能の高いモデル学習が阻害されている．本研究では，常識知識グラフ ConceptNet を活用した動作ラベル間の等価関係自動構築手法を提案する．提案手法は，動作ラベルと概念ノードのリンキング，RDF 化，SPARQL による関係抽出で構成される．実験の結果，人手定義に対して F 値 0.750 を達成し，構築した関係に基づくデータ拡張により動作認識精度が最大 3.33 ポイント向上することを確認した．

1　はじめに

動作認識は，コンピュータビジョン分野における重要な研究課題の一つである．近年の深層学習モデルの発展と，大規模な教師あり学習データセットの整備により，動作認識の精度は着実に向上してきた．一方で，実環境への応用において，動作認識器の汎化性能の向上が重要な課題となっている．データセットごとに撮影環境や視点が異なるため，あるデータセットで高い性能を示す認識器が，別のデータセットでは十分な性能を発揮できないことが多い．また，動作ラベルの定義もデータセットごとに異なるため，複数のデータセットを横断して学習することや，既存モデルを新しいデータセットへ適用することは容易ではない．

この問題に対して，メタ動作認識データセット MetaVD [1] では，複数のデータセット間における動作ラベルの関係性を人手で定義し，それを活用することで認識器の汎化性能を向上させている．一方で，複数データセット間の動作ラベルの関係性を人手で定義することには，二つの重要な課題が存在する．第一に，データセットの規模が大きいため，全てのラベル間の関係を人手で定義するには高いコストを要する．第二に，人手による判断では，ラベル間の関係を見落とす可能性がある．

これらの課題に対して，本研究では常識知識グラフを活用した動作ラベル間の等価関係の自動構築手法を提案する．具体的には，常識知識グラフの一つである ConceptNet [2] を用いて，複数データセットに含まれる動作ラベルを統合することで，動作ラベル間の等価関係を自動的に抽出する．ConceptNet は，日常的な概念とその関係を大規模に収録したグラフ構造のデータベースである．これを活用することで，異なる定義方法の動作ラベルを，共通の概念ノードを介して対応付けることが可能となる．

2　関連研究

動作認識の汎化性能を向上させるため，複数のデータセット間の関係性を活用する研究がある．MetaVD [1] は，六つの代表的な動作認識データセット間のラベルの関係を，equal（等価），similar（類似），is-a（階層）という 3 種類の関係として人手で定義している．実験結果から，これらの関係を利用することで，認識器の汎化性能が向上することが示されている．

一方，データセット間の関係性を人手で定義するアプローチには，意味的な一貫性の維持が困難であるという課題がある [3]．この課題に対し，[3] では常識知識グラフである Commonsense Knowledge Graph (CSKG) [4] を活用したアプローチを提案している．CSKG は，ConceptNet や Visual Genome など七つの知識源を統合した知識グラフであり，これを

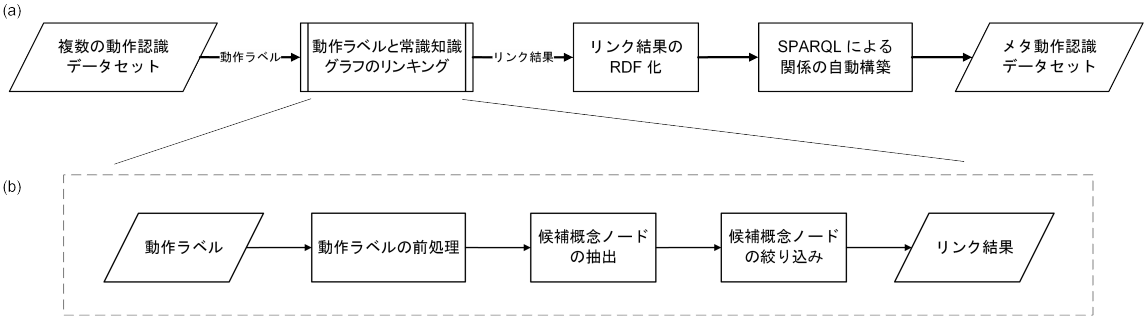

図 1: 提案手法の全体像および動作ラベルと ConceptNet のリンキングプロセス

MetaVD と組み合わせることで，データセットの意味的一貫性を向上させ，さらに抽象的な概念による特定の目的に応じた MetaVD のサブセット作成を可能にしている．

本研究では，常識知識グラフを活用して動作ラベル間の関係性を自動的に構築する手法を提案する．

3 提案手法

提案手法の全体像を図 1a に，動作ラベルと ConceptNet のリンキングプロセスを図 1b に示す．提案手法は，動作ラベルと常識知識グラフのリンキング，リンク結果の RDF 化，および SPARQL による関係の自動構築という三つの主要なプロセスから構成される．動作ラベルと常識知識グラフのリンキングでは，各データセットに含まれる動作ラベルを ConceptNet の概念ノードに対応付ける．次に，得られたリンク結果を RDF 形式に変換し，知識グラフとして構造化する．最後に，SPARQL クエリを用いて等価関係を自動的に構築し，メタ動作認識データセットを生成する．

3.1 動作ラベルと常識知識グラフのリンキング

本研究における動作ラベルと常識知識グラフのリンキング手法は，[5] によって提案された，テキスト中の概念を ConceptNet の概念ノードにリンクする手法を基礎としている．提案手法では，当該手法を動作ラベルと ConceptNet のリンキングに適用するよう拡張した．図 1b に示すように，本リンキングプロセスは，動作ラベルの前処理，候補概念ノードの抽出，候補概念ノードの絞り込みから構成される．

動作ラベルの前処理段階では，まずキャメルケースを分割する．次に，アンダースコア，ハイフン，スラッシュといった区切り文字を空白に統一的に変換する．続いて，全ての文字を小文字に変換し，ストップワードを除去した後，レンマ化を実施する．例えば，“Shoveling_snow” という動作ラベルは，これらの前処理により “shovel” と “snow” という二つのレンマに変換される．

候補概念ノードの抽出では，前処理で得られた各レンマを用いて，[5] で構築された ConceptNet の辞書を検索する．この辞書は，単語のレンマをキーとし，当該レンマを含む概念ノードのリストを値する構造を持つ．前処理で得られた各レンマについて辞書検索を行い，候補概念ノードを取得する．動作ラベルが複合語の場合は，各レンマに対する検索結果の和集合を候補概念ノードとする．

候補概念ノードの絞り込みでは，コサイン類似度に基づくフィルタリングを実施し，意味的類似性の低いノードを除外する．具体的には，動作ラベルおよび候補概念ノードのラベルに対してそれぞれ埋め込みを生成し，それらの間のコサイン類似度を計算する．この類似度が設定した閾値を下回るノードは，候補から除外する．埋め込みの生成には，MPNet [6]，Sent2vec [7]，ConceptNet Numberbatch[1)]，および Word2vec [8] の四つのモデルを用いた．また，閾値は 0.8 から 0.9 の範囲で設定し，実験を行った．これらのモデルの性能比較については，4.2 節で述べる．

3.2 リンク結果の RDF 化

動作ラベルと常識知識グラフのリンキングで得られたリンク結果を，Resource Description Framework（RDF）[9] 形式に変換する．RDF 形式を採用することで，動作ラベルと ConceptNet 概念ノード間の関係を形式的かつ機械可読な形で表現することが可能となる．

RDF 化の過程では，動作ラベルを主語，ConceptNet の概念ノードを目的語とし，両者の間にリンク関係を示す述語を設定する．これにより，動作

1） https://github.com/commonsense/conceptnet-numberbatch

ラベルと概念ノード間の意味的な関係が明示的に表現される．具体例として，ActivityNet データセットにおける動作ラベル “Applying_sunscreen” は，ConceptNet の “sunscreen”，“applying_sunscreen”，および “apply_sunscreen” という三つの概念ノードとリンクする（詳細は付録 A に示す）．

3.3 SPARQL による関係の自動構築

RDF 形式に変換されたデータを基に，異なるデータセット間で意味的に等価な動作ラベルを自動的に特定し，メタ動作認識データセットにおける動作ラベル間の関係として定義する．本研究では，MetaVD で定義されている関係の一つである等価関係に着目し，その自動構築を行う．

等価関係は，同一の ConceptNet 概念ノードにリンクする動作ラベル同士を等価とみなす関係として定義する．この関係を構築するため，SPARQL クエリテンプレートを作成し，RDF データから動作ラベル間の等価関係を抽出する．具体的には，指定した概念ノードとリンクする全ての動作ラベルを取得し，それらを等価集合として扱う．クエリの詳細は付録 B に示す．

4 実験

4.1 実験設定

データセットと評価タスク 提案手法により自動構築した等価関係の有効性を評価するため，MetaVD [1] が対象とする UCF101 [10], HMDB51 [11], ActivityNet [12], STAIR Actions [13], Charades [14], Kinetics-700 [15] の六つのデータセットに含まれる動作ラベル間の関係を構築した．3.1 節で述べた候補概念ノードの絞り込みにおいては，複数の埋め込み手法および閾値を用いて等価関係を構築し，その性能を検証した．

評価指標 [1] では，データ拡張と認識器の訓練・テストを通じて有効性を示しているが，この評価方法は実験コストが高いため，本研究では以下の二つの手法により評価を行う．

(1) 人手定義との一致度：[1] で定義された等価関係を正解データとし，適合率，再現率，F 値を算出する．さらに，正解データに含まれない予測結果（偽陽性）について，LLM による追加評価を実施し，等価と判定されたペアを真陽性として再計算する（LLM Adjusted）．

(2) 動作認識性能：UCF101，HMDB51，STAIR Actions から二つのデータセットペアを選び，等価関係を持つ動作ラベルに限定して実験を行う．ターゲットデータセットのみで学習したベースラインと，等価関係にあるソースデータセットの動画を追加して学習した場合の精度を比較する．

実装詳細 動作認識には，18 層の R(2+1)D モデル [16] を使用し，Kinetics-400 で事前学習された公開モデルで初期化した．動画は [1] に従い前処理を行った：連続する 16 フレームを抽出（学習時はランダム，テスト時は中央），128 × 171 ピクセルにリサイズ後 112 × 112 ピクセルにクロップ（学習時はランダム，テスト時は中央），Kinetics-400 の統計値で正規化．学習は 32 サンプルのミニバッチ，SGD（モーメンタム 0.9，重み減衰 10^{-4}），学習率 0.001（10 エポックごとに 1/10 に減衰），合計 45 エポック実施した．

4.2 実験結果

ベクトル化手法の比較 表 1 に，四つのベクトル化手法の性能比較を示す．文の埋め込みを行う MPNet と Sent2vec は，単語単位の埋め込みを行う Numberbatch および Word2vec より高い性能を示した．特に MPNet は閾値 0.85 で F 値 0.650（LLM 調整後 0.741）を達成し，最も高い性能を示した．さらに，MPNet について閾値 0.80 から 0.90 まで 0.01 刻みで評価した結果，閾値 0.84 で LLM 調整後 F 値 0.750 と最高値を達成した（付録 C 参照）．以降の実験では MPNet（閾値 0.84）を採用する．

表 1: 異なるベクトル化手法の性能比較（閾値 0.80 と 0.85）

手法	閾値	Default			LLM Adjusted		
		適合率	再現率	F 値	適合率	再現率	F 値
Numberbatch	0.80	0.107	0.653	0.183	0.164	0.743	0.268
	0.85	0.265	0.538	0.355	0.350	0.605	0.443
Sent2Vec	0.80	0.318	0.584	0.412	0.425	0.653	0.515
	0.85	0.422	0.488	0.452	0.522	0.541	0.531
Word2Vec	0.80	0.140	0.625	0.228	0.214	0.718	0.330
	0.85	0.272	0.519	0.357	0.342	0.576	0.429
MPNet	0.80	0.436	0.719	0.543	0.559	0.766	0.646
	0.85	0.709	0.600	0.650	0.867	0.647	0.741

データ拡張による動作認識性能の評価 表 2 に，提案手法で自動構築された等価関係を用いたデータ拡張の有効性を検証した結果を示す．HMDB51 をターゲットとして STAIR Actions で拡張した場合に，

ベースラインの 90.00%から 93.33%へと 3.33 ポイントの精度向上が確認され，提案手法の有効性が実証された．一方，他のデータセットペアでは精度向上が見られず，ベースライン精度が高い場合やデータセット間の特性の違いが大きい場合に，単純なデータ拡張では効果が限定的であることが示唆された．

表 2: 等価関係を用いたデータ拡張の効果

ソース (拡張用)	ターゲット (学習用)	データ拡張	訓練数	精度 (%)
UCF101	HMDB51	なし	280	97.50
		あり	679	95.83
HMDB51	UCF101	なし	399	100.00
		あり	679	100.00
STAIR Actions	UCF101	なし	485	98.94
		あり	5327	97.88
UCF101	STAIR Actions	なし	4842	98.40
		あり	5327	98.20
STAIR Actions	HMDB51	なし	280	90.00
		あり	4104	**93.33**
HMDB51	STAIR Actions	なし	3824	97.75
		あり	4104	97.50

5 おわりに

本研究では，常識知識グラフを活用し，動作認識データセット間における動作ラベルの等価関係の自動構築手法を提案した．提案手法では，異なるデータセットの動作ラベルを ConceptNet の概念ノードに対応付け，同一の概念ノードにリンクするラベル間に等価関係を定義する．これにより，人手による関係定義を必要としない効率的な関係構築を実現した．実験結果から，提案手法により自動構築された関係は，先行研究において人手で定義された関係と高い一致度を示すことを確認した．

今後は，ConceptNet が持つ概念間の階層関係および意味的な関係を活用し，MetaVD で定義されている is-a 関係および similar 関係の自動構築を目指す．

謝辞

本研究成果の一部は，国立研究開発法人新エネルギー・産業 技術総合開発機構 (NEDO) の委託業務 (JPNP20006，JPNP25006) の結果得られたものです．本研究は JSPS 科研費 23K11221，25K03232 の助成を受けたものです．

参考文献

[1] Yuya Yoshikawa, et al. MetaVD: A Meta Video Dataset for enhancing human action recognition datasets. **Computer Vision and Image Understanding**, 2021.

[2] Robyn Speer, et al. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In **AAAI**, 2017.

[3] Yasunori Yamamoto, et al. Towards Semantic Data Management of Visual Computing Datasets: Increasing Usability of MetaVD. In **ISWC (Posters/Demos/Industry)**, 2023.

[4] Filip Ilievski, et al. CSKG: The CommonSense Knowledge Graph. In **ESWC**, 2021.

[5] Maria Becker, et al. COCO-EX: A Tool for Linking Concepts from Texts to ConceptNet. In **EACL (System Demonstrations)**, 2021.

[6] Kaitao Song, et al. MPNet: Masked and Permuted Pre-training for Language Understanding. In **NeurIPS**, 2020.

[7] Matteo Pagliardini, et al. Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features. In **NAACL-HLT**, 2018.

[8] Tomas Mikolov, et al. Efficient Estimation of Word Representations in Vector Space. **arXiv**, 2013.

[9] Graham Klyne and Jeremy J. Carroll. Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation, 2004.

[10] Khurram Soomro, et al. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. **arXiv**, 2012.

[11] H. Kuehne, et al. HMDB: A large video database for human motion recognition. In **ICCV**, 2011.

[12] Fabian Caba Heilbron, et al. ActivityNet: A large-scale video benchmark for human activity understanding. In **CVPR**, 2015.

[13] Yuya Yoshikawa, et al. STAIR Actions: A Video Dataset of Everyday Home Actions. **arXiv**, 2018.

[14] Gunnar A. Sigurdsson, et al. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding. In **ECCV**, 2016.

[15] Joao Carreira, et al. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In **CVPR**, 2017.

[16] Du Tran, et al. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In **CVPR**, 2018.

A　RDF 化の具体例

本文 3.2 節で述べたリンク結果の RDF 化について，ActivityNet データセットにおける動作ラベル “Applying_sunscreen” を例として，図 2 にその具体的な RDF 形式のリンク結果を示す．この動作ラベルは，ConceptNet の “sunscreen”，“applying_sunscreen”，および “apply_sunscreen” という三つの概念ノードとリンクし，それぞれが `act:relatedToConcept` 述語によって関連付けられている．

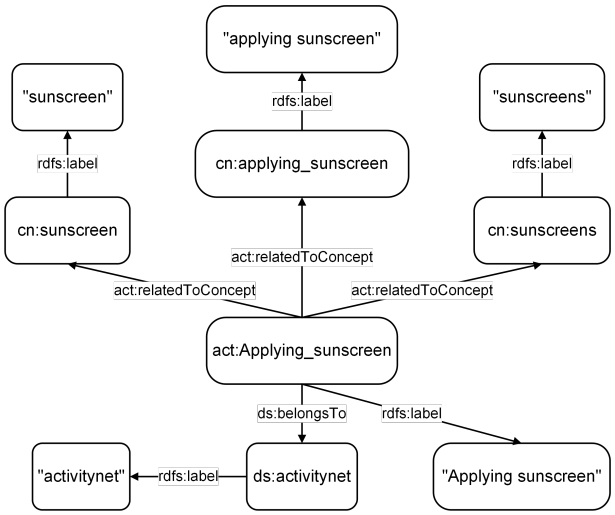

図 2: 動作ラベル “Applying_sunscreen”（ActivityNet データセット）の RDF 形式のリンク結果

B　SPARQL クエリ

本文 3.3 節で述べた等価関係の自動構築に使用した SPARQL クエリテンプレートをソースコード 1 に示す．このクエリでは，9 行目の`{concept}`に概念ノードラベルを埋め込むことで，指定した概念とリンクする動作ラベルの等価集合を取得できる．

```
1  PREFIX act: <http://example.org/action>
2  PREFIX ds: <http://example.org/dataset>
3  PREFIX cn: <http://conceptnet.io/c/en/>
4  PREFIX rdf: <http://www.w3.org/1999/02/22-
       rdf-syntax-ns#>
5  PREFIX rdfs: <http://www.w3.org/2000/01/
       rdf-schema#>
6
7  SELECT DISTINCT ?action_label ?
       dataset_label
8  WHERE {
9    ?input_concept rdfs:label "{concept}" .
10   ?action act:relatedToConcept ?
       input_concept;
11     rdfs:label ?action_label ;
12     ds:belongsTo ?dataset .
13   ?dataset rdfs:label ?dataset_label .
14 }
15 ORDER BY ?action_label ?dataset_label
```

Listing 1: 動作ラベル間の等価関係を取得する SPARQL クエリテンプレート

C　MPNet の閾値別詳細結果

表 3 に，MPNet を用いた場合の閾値別の詳細な評価結果を示す．閾値 0.80 から 0.90 まで 0.01 刻みで変化させた結果，閾値 0.84 において LLM 調整後の F 値が 0.750 と最も高い値を示した．閾値を高く設定するほど適合率は向上するが，再現率が低下する傾向が確認できる．

表 3: MPNet の閾値別評価結果

閾値	Default			LLM Adjusted		
	適合率	再現率	F 値	適合率	再現率	F 値
0.80	0.436	0.719	0.543	0.559	0.766	0.646
0.81	0.475	0.706	0.568	0.595	0.751	0.664
0.82	0.535	0.688	0.602	0.669	0.733	0.700
0.83	0.627	0.663	0.644	0.766	0.706	0.735
0.84	0.697	0.625	0.659	0.850	0.670	**0.750**
0.85	0.709	0.600	0.650	0.867	0.647	0.741
0.86	0.731	0.578	0.646	0.893	0.626	0.736
0.87	0.773	0.563	0.651	0.944	0.611	0.742
0.88	0.812	0.525	0.638	0.971	0.569	0.718
0.89	0.839	0.503	0.629	0.995	0.546	0.705
0.90	0.870	0.481	0.620	0.994	0.515	0.678