人間の動作を日本語で説明するためのキャプションデータセット

重藤 優太郎[1]　吉川 友也[1]　蘭 佳慶[1]　竹内 彰一[1,2]
[1]千葉工業大学 STAIR Lab　[2]産業技術総合研究所 人工知能研究センター
{shigeto,yoshikawa,lin,takeuchi}@stair.center

1 はじめに

動画のキャプション生成は, 入力として動画が与えられたときに, その動画の内容を説明する文 (キャプション) を出力するタスクである [19, 20, 24]. 図 1 に動画とそのキャプションの例を示す.

現在の動画キャプション生成に関する研究の大部分は, 英語キャプションの生成に注目しており, 英語以外の言語を対象としたものは少ない. キャプション生成は, 高度な動画検索, 生活支援, コミュニケーションロボットなどへの応用が期待されるため, 英語以外でのキャプション生成に対するニーズも大きいものの, 現状ではそのニーズに応えられていない.

そこで本研究では, そういったニーズの一つである, 日本語での人間の動作説明に取り組む. より具体的には, 動作説明において重要であると考えられる「誰がどこで何をしているか」を日本語で説明できる方法を開発する. これを実現するために, 本研究では, (i) 日本語の動画キャプションデータセットの構築 (2 節) と, (ii) 日本語のための動画キャプション生成法の開発 (3 節) を行った. 構築した日本語キャプションデータセットは `http://sa-captions.stair.center/` からダウンロードすることができる.

2 日本語キャプションデータセット

2.1 アノテーション

日本語キャプションを付与するにあたり, アノテーションの対象として STAIR Actions [25] の訓練用データに含まれる動画 79,822 本を用い, 各動画に対し約 5 キャプションのアノテーションを行なった. キャプションの総数は, 399,233 文となった.

元データである STAIR Actions は家庭やオフィスで見られる人間の 100 種類の動作からなる動画データセットであり, 本研究の目的である人間の動作の認識・説明という目的に適している.

日常動作の説明において重要な要素は「どこでだれが何をした」であると思われる. この考えに基づき, 本研究におけるアノテーションの指針として, 「場所に関する説明 (scene)」, 「人に関する説明 (person)」, 「動作に関する説明 (action)」の 3 要素が記述されることとした.

目的が日常動作の説明であることから, 文法に関する多様性は不要であると考えた. そのため, scene, person, action をそれぞれ名詞句および動詞句としてアノテーションを行い, 最後に助詞「で」と「が」を補完することにより文となるように設計した (図 1). したがって, 本データセットのキャプションは全て「scene で person が action」という形で構成されている.

さらに, キャプション付与に際して, 以下のガイドラインを設けた.

- 事実のみを記述する (会話などを想像して記述しない)
- 作業者の感情・意見・予想を書かない
- 場所がわからない場合, 部屋, 屋内, 屋外などと記述する
- どんな人かがわからない場合, 人と記述する
- 「ですます」調ではなく「である」調で記述する

アノテーションは, 株式会社バオバブを通し, 125 名の作業者によって 4ヶ月間で行われた. 表 1 に付与したキャプションの統計を示す. 統計を計算する際の単語分割は KyTea [11] を用いた.

表 1: 付与したキャプションの統計.

	語彙数	平均文字数	最大文字数	最小文字数
scene	5,214	6.3	60	1
person	4,383	10.0	55	1
action	10,098	12.0	73	1
sentence	13,836	30.2	135	8

表 2: 動画キャプションデータセット. MSVD は英語以外に 15 言語のキャプションが少量ではあるが付与されている.

データセット	クリップ数	説明文数	言語
MSVD [1]	2k	70k	英 + 15 言語
MSR-VTT [23]	10k	200k	英
Charades [17]	10k	16.1k	英
LSMDC [16]	118k	118.1k	英
YouCook [2]	–	2.7k	英
ActivityNet [7]	100k	100k	英
VideoStory [4]	123k	123k	英
Ours	79.8k	399.2k	日

2.2 関連研究

表 2 に最近の主要な動画のキャプションデータセットを示す. 説明文を付与する対象の動画は, データセット毎に様々なリソースから収集されている. YouCook, MSR-VTT, ActivityNet Captions は, YouTube から収集された動画が利用されている. 特に, YouCook は料理動画だけを集め, MSR-VTT はシーンが偏らな

scene	person	action
街中	青い洋服の男の子	写真を撮っている
屋外	青い服を着た男性	写真を撮っている
黒い柱のある道路	水色の服を着た少年	怪物のコスプレをした人と写真を撮ってもらっている
車と黒い柱のある屋外	金色の仮装をした男性	立って子供を抱えている
石造りの建物のある歩道	羽のついている金の衣装を着た人	子供と一緒に写真を撮っている

図 1: 動画とキャプションの例. 「scene で persion が action」というように「で」と「が」を補完することで文となる.

いように様々なクエリで動画を検索して収集している. LSMDC は, 映画から動画を収集している. また, Charades はキーワードからの台本作成を AMT で行い, その台本に基いて AMT で動画撮影を依頼し著作権フリーの動画を収集している.

説明文付与はクラウドソーシングで行われる. MSR-VTT は動画を数十秒の短い動画にして, それらに対して単文の説明文付与を行っている. ActivityNet Captions では, 動画 1 本を説明する 1 パラグラフ分の説明文をワーカーに付与してもらい, その後, そのパラグラフの各文が動画中のどの時間区間に対応するかを付与してもらうことで. 時間区間の重複を含む説明文が付けられている.

STAIR Actions は, 人間の動作認識を目的としたデータセットであるため, 各動画の長さが平均 5 秒と短い. そのため, 本研究で構築したキャプションデータセットは, 動画 1 本に対してキャプションの付与を行なった.

現存する動画キャプションデータセット基本的に英語で記述されている. MSVD に関しては, 英語の他に 15 言語で少量のキャプションが付与されている. しかしながら, 日本語キャプションは付与されていない.

3 注意機構を用いたキャプション生成

入力となる動画を x, 出力となるキャプションを $s = (y_1, y_2, \cdots, y_N)$ で表す. y_t は t 番目の単語である.[1] キャプション生成モデルは, 動画 x から文 s が生成される条件付き確率 $P(s \mid x)$ を以下の式でモデル化する:

$$P(s \mid x; \theta) = \prod_{t=1}^{N} P(y_t \mid x, y_{<t}; \theta).$$

θ はモデルのパラメータであり, $y_{<t}$ は t 番目までに生成された単語列である.

t 番目の単語の予測には, 再帰型ニューラルネットワーク (RNN) を用いる. 先行研究において, 動画から種々の特徴を抽出し, それらを RNN に入力することで, 洗練されたキャプション生成が行えることが報告されている [3, 10, 12, 14, 21]. これらに従い, 本論文でも動画から種々の特徴を抽出し, それらを RNN の入力とする.

3.1 動画からの特徴抽出

先行研究に従い [3, 10, 12, 14, 21], 動画からの特徴抽出器として事前学習済みの画像分類器, 動画分類器, 物体認識器を用いる.

画像分類器 フレームの特徴を利用するため, 画像分類器の最終層を特徴ベクトルとして使用する. 本研究では, 画像分類器として ImageNet で学習された ResNet-152 [6] を使用した.[2] 動画から 3 fps でフレームを取り出し, 取り出した全てのフレームから特徴ベクトル $\mathbf{i} \in \mathbb{R}^{2048}$ を生成した. 得られた特徴ベクトルの集合を $\mathcal{I} = \{\mathbf{i}_j\}_{j=0}^{N_\mathcal{I}}$ で表す.

動作認識器 動画そのものの特徴を利用するため, 動画分類器の最終層を特徴ベクトルとして使用する. 動画分類器として Kinetics-400 で学習された 3D ResNeXt-101 [5] を使用した.[3] 動画から連続した 16 フレームを取り出し, 逐次 ResNeXt によって特徴ベクトル $\mathbf{v} \in \mathbb{R}^{2048}$ に変換した. 得られた特徴ベクトルの集合を $\mathcal{V} = \{\mathbf{v}_j\}_{j=1}^{N_\mathcal{V}}$ と表す.

物体認識器 動画中に存在する物体の情報を利用するため, 物体認識器の認識結果を使用する. 物体認識器として MS COCO で学習された YOLOv3 [15] を使用した.[4] 画像特徴と同様に 3 fps でフレームを切り出し, 各フレームに対して物体認識を行なった. ここで, 認識された物体名はフレームを考慮せずにまとめた. 物体名は単語ベクトル $\mathbf{a} \in \mathbb{R}^d$ に変換し, 得られた特徴ベクトルの集合を $\mathcal{A} = \{\mathbf{a}_j\}_{j=1}^{N_\mathcal{A}}$ と表記する.

[1] y_t を便宜上単語と呼ぶが, 文字やサブワードなども取り扱えるものとする.

[2] https://pytorch.org/docs/master/torchvision/models.html#id3

[3] https://github.com/kenshohara/video-classification-3d-cnn-pytorch

[4] https://github.com/OlafenwaMoses/ImageAI/

3.2 注意機構付きデコーダ

動画 x の特徴ベクトル $\mathcal{A}, \mathcal{I}, \mathcal{V}$ を RNN によってキャプション s へと変換する. t 番目の単語 y_t の生成確率 $\mathbf{p}_t$ は以下によって計算できるとする:

$$\begin{aligned}\mathbf{h}_t &= \mathrm{RNN}(\mathbf{y}_{t-1}, \mathbf{h}_{t-1}),\\ \mathbf{x}_t &= [f_i(\mathcal{I}, \mathbf{h}_t), f_v(\mathcal{V}, \mathbf{h}_t), f_a(\mathcal{A}, \mathbf{h}_t)],\\ \mathbf{o}_t &= \tanh(\mathbf{W}_o[\mathbf{h}_t, \mathbf{x}_t] + \mathbf{b}_o),\\ \mathbf{p}_t &= \mathrm{softmax}(\mathbf{W}\mathbf{o}_t + \mathbf{b}).\end{aligned} \tag{1}$$

$f_*(\cdot,\cdot)$ は注意機構を計算する関数である. $\mathcal{Z} = \{\mathbf{z}_i\}_{j=1}^{N_\mathcal{Z}}$ とした場合, 注意機構は以下で計算される:

$$f_z(\mathcal{Z}, \mathbf{h}_t) = \sum_{j=1}^{N_\mathcal{Z}} \alpha_j \mathbf{z}_j,$$

$$\alpha_j = \frac{\mathbf{m}_z^{\mathrm{T}} \tanh(\mathbf{W}_z[\mathbf{z}_j, \mathbf{h}_t] + \mathbf{b}_z)}{\sum_{k=1}^{N_\mathcal{Z}} \mathbf{m}_z^{\mathrm{T}} \tanh(\mathbf{W}_z[\mathbf{z}_k, \mathbf{h}_t] + \mathbf{b}_z)}.$$

RNN の隠れ状態ベクトルの初期値 $\mathbf{h}_0$ は以下の値を用いた.

$$\mathbf{h}_0 = \mathbf{W}_x[g(\mathcal{I}), g(\mathcal{V}), g(\mathcal{A})] + \mathbf{b}_x$$

$g(\cdot)$ は平均プーリング関数である. RNN の各ユニットには gated recurrent unit (GRU) を用いた.

学習データ $\mathcal{D}$ が与えられた時, モデルのパラメータ (物体名ベクトル $\mathbf{a}_*$ およびデコーダのパラメータ $\mathbf{W}_*, \mathbf{m}_*, \mathbf{b}_*$) は以下の最適化問題を解くことで得られる:

$$\min_\theta - \sum_{(x,s)\in\mathcal{D}} \log P(s \mid x; \theta).$$

実際の生成時には, 式 (1) の $\mathbf{p}_t$ 基づいてスコアが大きい単語列を選択する. 単語列の探索にはビームサーチを用いた. スコアを正規化するために length normalization [22] を行なった.

3.3 文生成

文の生成方法として直接生成とテンプレート生成の 2 種類の方法でキャプションを生成する. 直接生成は, 通常の文生成法であり, RNN によって文そのもの (「scene で person が action」) を生成する. テンプレート生成は, scene, person, action のそれぞれを別の RNN で生成し, それらに「が」と「で」を補完することで文を生成する. テンプレート生成の場合, scene, person, action のそれぞれを生成するための RNN が必要となる.

4 実験

構築した日本語キャプションデータセットを用いて, 実際にキャプション生成を行う. 本実験の目的は, 日本語のキャプションがどの程度できるか (どの程度人間の動作を説明できるか), 及び, どうすればできるかを検証することである. 具体的には, (i) 構築したデータセットのベンチマーク, (ii) キャプションの直接生成とテンプレート生成の比較, (iii) どの動画特徴 ($\mathcal{A}, \mathcal{I}, \mathcal{V}$) がキャプション生成に効果的か, を示す.

実験設定 全動画の 80% (63,856 本) を学習用データ, 10% (7,983 本) を開発用データ, 残りを評価用データとした.

各キャプションは SentencePiece [8] を用いて, サブワードに分割した.[5] この際, 語彙サイズは 8,000 とした.

先行研究に従い [3, 10, 12, 14, 21], 生成されたキャプションの評価指標として BLEU [13], ROUGE-L [9], CIDEr [18] を用いる.[6] 評価時の単語分割には KyTea を用いた.[7]

開発用データを用いて, CIDEr を基準にハイパーパラメータ (物体名ベクトルの次元数 d, RNN の隠れ状態ベクトルの次元数, RNN の層数, Adam の学習率, weight decay パラメータ, dropout 確率, ビーム幅, length normalization の係数) を調整した.

実験結果 表 3 に scene, person, action をそれぞれ生成した結果を示す. 表から, scene 以外の場合においては, 全ての特徴を入力した場合 ($\mathcal{I}, \mathcal{V}, \mathcal{A}$) に最もよい結果となることがわかる. 一方, scene に関しては, $\mathcal{I}$ のみを入力した方がよい結果となった.

表 4 に文を生成した場合の実験結果を示す. 表中の Greedy は, 開発用データで最も良いスコア (CIDEr) を得た scene, person, action 生成の出力を組み合わせたものである (開発用データでは, scene 生成には $\mathcal{I}$, person 生成, action 生成には $\mathcal{I}, \mathcal{V}, \mathcal{A}$ を入力する場合が最も良い結果となった).

入力となる動画特徴に注目した場合, $\mathcal{I}, \mathcal{V}, \mathcal{A}$ をそれぞれ単独で入力する場合よりも, 全てを入力する方がよい結果となった. それぞれの特徴を単独で入力した場合の結果を比較した場合, $\mathcal{I}, \mathcal{V}, \mathcal{A}$ の順でよい結果を得た.

文の生成方法に注目した場合, どのような動画特徴を入力した場合においても, 文を直接生成するよりも, テンプレート生成を行なった場合の方がよい結果となることがわかる.

この実験を通じて最も良い結果を得たのは Greedy であった.

5 おわりに

本論文では, 人間の動作を認識・説明するために, 動画日本語キャプションデータセットを構築した. このデータセットは, STAIR Actions が提供している動画の一部 (79,822 本) と日本語キャプション (全 399,233 文) で構成されている. 既存の動画キャプションデータセットと比較した場合 (表 2), 本データセットを除いて日本語のキャプションが付与されているデータセットは存在しておらず, また, 本データセットよりも規模の大きいデータセットもない.

[5] https://github.com/google/sentencepiece
[6] 評価指標の計算には cocoapi を利用した.
[7] http://www.phontron.com/kytea/

表 3: 文を構成する各要素 (scene, person, action) を生成した結果.

	scene			person			action		
input	BLEU-4	ROUGE-L	CIDEr	BLEU-4	ROUGE-L	CIDEr	BLEU-4	ROUGE-L	CIDEr
$\mathcal{I}$	**0.825**	**0.867**	**1.889**	0.737	0.799	**1.755**	0.805	0.864	3.295
$\mathcal{V}$	0.804	0.854	1.821	0.656	0.740	1.451	0.787	0.857	3.251
$\mathcal{A}$	0.555	0.667	1.083	0.365	0.430	0.577	0.477	0.673	1.127
$\mathcal{I}, \mathcal{V}, \mathcal{A}$	0.813	**0.867**	1.842	**0.750**	**0.807**	1.752	**0.825**	**0.878**	**3.471**

表 4: 日本語キャプション生成の実験結果. 各評価指標で最も良い数値を, 太字で表している. Greedy は, 開発用データで最も良いスコアを得た scene, person, action 生成器の出力を組み合わせた結果である.

生成方法	動画特徴	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	CIDEr
直接生成	$\mathcal{I}$	0.874	0.811	0.756	0.704	0.790	1.798
	$\mathcal{V}$	0.856	0.784	0.723	0.667	0.771	1.689
	$\mathcal{A}$	0.663	0.532	0.442	0.369	0.598	0.535
	$\mathcal{I}, \mathcal{V}, \mathcal{A}$	0.879	0.820	0.767	0.717	0.799	1.883
テンプレート生成	$\mathcal{I}$	0.897	0.850	0.802	0.755	0.794	1.933
	$\mathcal{V}$	0.871	0.816	0.762	0.711	0.775	1.805
	$\mathcal{A}$	0.671	0.560	0.472	0.399	0.609	0.584
	$\mathcal{I}, \mathcal{V}, \mathcal{A}$	0.902	0.857	0.811	0.765	0.800	2.016
	Greedy	**0.905**	**0.861**	**0.815**	**0.768**	**0.801**	**2.026**

繰り返しになるが, 本データセットを構築した目的は人間の動作を認識・説明することにある. そのため, 付与したキャプションは「だれがどこで何をしている」が記述されている. 「場所」,「人」,「動作」はそれぞれ別々に付与されており, その結果として, 文を直接生成するのみではなく, 文の要素ごとに生成することができる. また, それら生成された要素を組み合わせることで文を構築 (テンプレート生成) することもできる. 実験の結果, 本データセットにおいては, 文を直接生成するよりも, テンプレート生成を行なった場合の方がよい結果を得ることを示した.

謝辞 この成果は, 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) の委託業務の結果得られたものです. キャプションのアノテーションにご協力いただいた株式会社バオバブ, アノテーションツールの開発にご協力いただいた株式会社 mokha 蒲地輝尚氏に感謝いたします.

参考文献

[1] D. L. Chen and W. B. Dolan. Collecting Highly Parallel Data for Paraphrase Evaluation. In *ACL*, pages 190–200, 2011.

[2] P. Das, C. Xu, R. F. Doell, and J. J. Corso. A Thousand Frames in Just a Few Words: Lingual Description of Videos through Latent Topics and Sparse Object Stitching. In *CVPR*, pages 2634–2641, 2013.

[3] Z. Gan, C. Gan, X. He, Y. Pu, K. Tran, J. Gao, L. Carin, and L. Deng. Semantic Compositional Networks for Visual Captioning. In *CVPR*, pages 5630–5639, 2017.

[4] S. Gella, M. Lewis, and M. Rohrbach. A Dataset for Telling the Stories of Social Media Videos. In *EMNLP*, pages 968–974, 2018.

[5] K. Hara, H. Kataoka, and Y. Satoh. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet? In *CVPR*, pages 6546–6555, 2018.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016.

[7] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles. Dense-Captioning Events in Videos. In *ICCV*, pages 706–715, 2017.

[8] T. Kudo and J. Richardson. SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing. In *EMNLP: System Demonstrations*, pages 66–71, 2018.

[9] C.-Y. Lin. ROUGE: A Package for Automatic Evaluation of Summaries. *Text Summarization Branches Out*, 2004.

[10] X. Long, C. Gan, and G. de Melo. Video Captioning with Multi-Faceted Attention. *TACL*, 6:173–184, 2018.

[11] G. Neubig, Y. Nakata, and S. Mori. Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis. In *ACL*, pages 529–533, 2011.

[12] Y. Pan, T. Yao, H. Li, and T. Mei. Video Captioning with Transferred Semantic Attributes. In *CVPR*, 2017.

[13] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL*, pages 311–318, 2002.

[14] S. Phan, Y. Miyao, and S. Satoh. MANet: A Modal Attention Network for Describing Videos. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1889–1894, 2017.

[15] J. Redmon and A. Farhadi. YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1609.08144*, abs/1804.02767, 2018. URL `http://arxiv.org/abs/1804.02767`.

[16] A. Rohrbach, A. Torabi, M. Rohrbach, N. Tandon, C. Pal, H. Larochelle, A. Courville, and B. Schiele. Movie Description. *IJCV*, 123(1):94–120, 2017.

[17] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding. In *ECCV*, pages 510–526, 2016.

[18] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. CIDEr: Consensus-Based Image Description Evaluation. In *CVPR*, pages 4566–4575, 2015.

[19] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to Sequence-Video to Text. In *ICCV*, pages 4534–4542, 2015.

[20] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating Videos to Natural Language Using Deep Recurrent Neural Networks. In *NAACL*, 2015.

[21] X. Wang, Y.-F. Wang, and W. Y. Wang. Watch, Listen, and Describe: Globally and Locally Aligned Cross-Modal Attentions for Video Captioning. In *NAACL*, pages 795–801, 2018.

[22] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation. *arXiv preprint arXiv:1609.08144*, 2016.

[23] J. Xu, T. Mei, T. Yao, and Y. Rui. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *CVPR*, pages 5288–5296, 2016.

[24] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing Videos by Exploiting Temporal Structure. In *ICCV*, pages 4507–4515, 2015.

[25] Y. Yoshikawa, J. Lin, and A. Takeuchi. STAIR Actions: A Video Dataset of Everyday Home Actions. *arXiv preprint arXiv:1804.04326*, 2018. URL `http://arxiv.org/abs/1804.04326`.