動画からの動作記述の生成とその評価

平川 幸司　　小林 哲則　　林 良彦
早稲田大学理工学術院
hirakawa@pcl.cs.waseda.ac.jp

1　はじめに

動画内容の言語による表出を目的としたキャプション生成の研究 [1] が盛んになっている．一般に，キャプションでは動画において描写されている情景や動きなどが言語化されるが，これらの情報がすべて応用において必要とは限らない．特に重要と考えられるのは人間などの動作に関する情報であり，この場合は，動作内容に限定した言語記述の生成が求められる．本研究では，動作に関する言語記述を動作記述と呼び，あらかじめ別の手段で生成された一般的なキャプションを変換することにより動作記述を生成する手法を提案し，その内的・外的評価について議論する．

2　動作記述とその生成

本研究では図 1 に示すように，キャプションを適切に生成できるキャプション生成器の存在を仮定し，その出力を Seq2Seq モデルにより動作記述へ書き換えるアプローチをとる．

一般に，動画中の区間に対して生成されるキャプションは，一文で表され，動作以外の状況描写をも含み，複文の構造を持つことが多い．これに対し，我々が想定する動作記述は，人間が動作を行っている動画中の区間に対して，その動作を簡潔に記述するものであり，(1) 独立した単文の系列であり，(2) 各単文は人間が主語で，動詞は現在形もしくは現在進行形，という言語特徴を持つものとして規定する．以上から，一般的なキャプションから動作記述への書き換えは，動作に関係ない部分を排除しつつ，一連の動作のそれぞれを単文として表す書き換えタスクとなる．

2.1　動作記述データセット

動作記述書き換え器の学習・評価のために，ActivityNet Captions [1] のキャプションデータを上記の基準に合致するように人手により書き換えることで，動作記述データセットを作成した．具体的には，分割された単文から動作を含まないキャプション (例: ”We see a title card after a shot.”) を除き，時制，主語をそろえる，という方法をとった．また，書き換えの際には原文に出現する表現をなるべく残すようにした．ActivityNet Captions のデータには動作記述を一切含まないキャプションも存在し，その場合は文の除去を表す記号”None”を書き換え後の表現とした．現在までに 11,772 件のキャプションに対するデータを作成した．

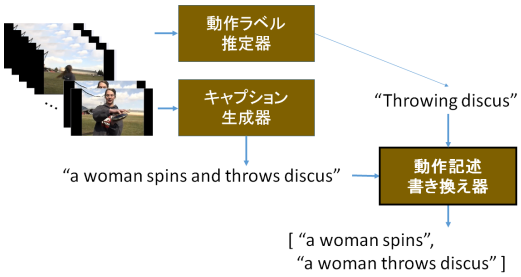

図 1: 書き換えによる動作記述の生成

2.2　書き換えによる動作記述の生成

上記の方法で作成した学習データを用いて Seq2Seq モデルによる動作記述への書き換えを学習するが，タスク・学習データの特性から以下を考慮した．

要約生成モデル:　本タスクは一種の要約タスクであり，通常の要約タスク堂奥に，入力中の一定の単語は出力においても保存されるべきである．このことからコピー機構 [2] が導入された Seq2Seq の要約生成モデルである Pointer-generator networks [3] を用いた．

類似タスクからのドメイン適応:　複文の単文群への分割という観点で本タスクと類似したタスクである Split and Rephrase [4] のデータ (S&R データと呼ぶ) によ

り基本的な単文への書き換えを学習 (out domain) し，この結果を動作記述生成の本ドメイン (in domain) に適応させる [5] ことにより，動作記述でない文のフィルタリングが達成されることを目的とした．具体的には，まず in domain の動作記述のデータと out domain の S&R データを合わせた全体で初期の学習を行い，次にこの学習で得られたパラメータを初期値として用い，in domain の動作記述書き換えデータのみでファインチューニングを行った．S&R データ[1]は，WebNLG data [2]に含まれる複文を単文に分割することで作成されており，データの規模は 886,857 件に達する．

動作ラベルの利用: 書き換えにおけるサブドメイン識別子として動作ラベルと呼ばれるラベルを用い，その効果を評価する．ここで，動作ラベルとは，動画区間中で描画されている動作の種別を簡潔に表すラベルであり，ActivityNet [6] では ActivityNet Captions と同じ動画を対象に，200 種類の動作ラベルを用いて動作区間へのラベル付けが行われている．同じ動作ラベルが付与されている動画は似た表現のキャプションが付与されていることが期待できるので，動作の種別を反映した書き換えが行えることが期待できる．

3 内的評価: 動作記述の評価

3.1 評価の考え方

内的評価の目的は，書き換え器自体の優劣を評価することにあり，このために生成された動作記述が想定する動作記述とどのくらい合致するかを評価する．動作記述は動作を簡潔に表す単文群であるため，この評価は基本的には単文群間のマッチングの評価となる．具体的には，図 2 に示すように，参照文と生成された候補文との文間対応付けを行い，これから漏れたものを「生成漏れ」,「過剰生成」として検出し，これらが少ない動作記述が高く評定されるように，以下の評価指標 AE を設定する．

$$AE = (1 - R_{ug})(1 - R_{og}) \tag{1}$$

$$R_{ug} = \frac{N_{ug}}{N_{ref}} \tag{2}$$

$$R_{og} = \frac{N_{og}}{N_{can}} \tag{3}$$

ここで，R_{ug} は参照文群からみた生成漏れの割合，R_{og} は候補文群からみた過剰生成の割合である．

[1] https://github.com/shashiongithub/Split-and-Rephrase

[2] https://gitlab.com/shimorina/webnlg-dataset

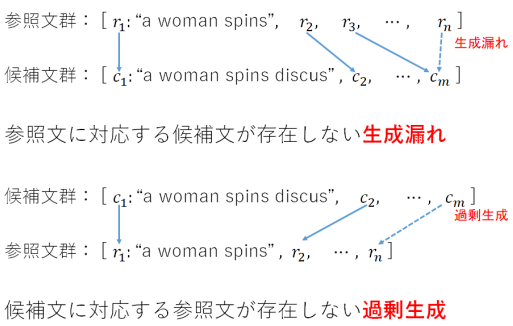

図 2: 文群間の対応付けにおける生成漏れと過剰生成

表 1: 評価指標 AE による書き換えシステムの評価 (手動対応付け)

no_tag	true_tag	rand_tag
0.326	**0.700**	0.687

3.2 手動対応付けによる書き換え器の評価

動作記述データセットからランダムに抽出した 200 件を参照データとし，以下に述べる 3 種類の書き換え器の出力との間で，主語・動詞・目的語の一致を条件として，人手により文群を対応付けた．この結果から評価指標 AE の値を算出し，書き換え器を評価した．

書き換え器の設定:

- **no_tag:** S&R の全データセット (886,857 件) で事前学習を行ったあと動作記述書き換えデー (全 8,758 件) でドメイン適応した．
- **true_tag:** no_tag と同様に事前学習したあとに，動作記述の書き換え前のキャプションの冒頭に ActivityNet の正しい動作ラベルを追加したデータでドメイン適応した．
- **rand_tag:** true_tag の時にキャプションの冒頭に追加するラベルをランダムな別のラベルに変えた．

評価結果: 200 件に対する各システムの出力それぞれに対して評価指標 AE を求めた後，平均値を計算した結果を表 1 に示す．true_tag の結果が最もよかったことから，正しい動作ラベルをサブドメイン識別子として利用する有効性が確認された．また，rand_tag と no_tag の結果を比較するとランダムなラベルを付与した方がよかったことから，動作ラベルの付与の精度が

低くても一定の有効性があると考えられる．なお，事前実験の結果から，S&R データを用いた事前学習は有効であり，また，そのデータ量についても，現在の規模においては，多いほどよいことを確認している．

3.3 自動対応付けによる書き換え器の評価

実際の適用場面においては，自動で生成漏れ・過剰生成を検出する必要がある．このため，参照文 r_i に対して候補文 c_j の対応付の良さをスコアリングし，次にこれを閾値でフィルタリングすることにより文間対応付けを行う．

文間の対応付けスコアリング: 以下のスコアリング手法を比較した．

- **ROUGE:** 要約で用いられる評価指標 ROUGE を求め，この最大値を与える文を対応付け候補とする．
- **BERT:** 参照文と候補文のベクトル表現を bert–as–service [3]を用いて取得し，ベクトル間のコサイン類似度の最大値を利用する．
- **Sultan [7]:** 対象の文対中の単語列に対して完全に一致する系列，固有名詞を先に対応付けた後に，構文解析し，候補単語の類似度を測り，対応付けを行う．候補単語の類似度の計算においては，英語の言い換えデータである Paraphrase Database (PPDB) を用いている．候補の単語ペアがこのデータベース中に存在すれば類似度が 1，存在しなければ類似度は 0 となる．本実験においては Sultan らの手法を拡張することで文間の意味的類似度を取得できる A Short Answer Grader [8] を利用した．

閾値によるフィルタリング: 事前実験の結果から，生成漏れ検出でのフィルタリングの閾値は BERT では 0.85，Sultan では 0.99，ROUGE では 0.5 とした．過剰生成でのフィルタリングの閾値は BERT では 0.8，Sultan では 0.9，ROUGE では 0.5 とした．

評価結果: 事前実験の結果，生成漏れの検出においては Sultan の対応付け手法が最も有効 (F 値: 0.85) であり，過剰生成の検出においては ROUGE が最も有効 (F 値: 0.62) であった．そこで，これらの手法を用いて評価指標 AE を前節と同様の 3 種類の書き換え器に対して求めた結果を表 2 に示す．3 種類の書き換え器の優劣は，true_tag > rand_tag > no_tag であり，表 1 の結果と一致した．また，双方の場合の全データ (200 件) に対する順位付けのピアソン相関係数は 0.805 と高く，自動対応付けに基づく評価は一定の精度で可能であることが確認できた．

表 2: 評価指標 AE による書き換えシステムの評価 (自動対応付け)

no_tag	true_tag	rand_tag
0.279	**0.641**	0.630

[3]https://github.com/hanxiao/bert-as-service

4 外的評価: 動画検索への適用

外的評価の目的は，ある書き換え器により生成された動作記述の特定の応用における有効性・妥当性を評価することにより，その書き換え器の有用性を評価することにある．動作記述は，動画中の動作のある区間に対して付与するため，動画検索における実際の検索対象として利用できる．すなわち，動作記述を検索対象とする動画検索によって書き換え器の外的評価を行うことができる．検索実験においては，検索対象を ActivityNet Captions のキャプションとした場合と，書き換え器が出力した動作記述とした場合の比較を行い，さらに書き換え器の比較も行う．

4.1 検索実験の設定

データ: 動作記述書き換えデータの学習方法は 3.2 節と同様とし，S&R データの学習量は 886,857 件に固定した．本実験で検索対象として利用した動作記述の量は 837 件である．これは，ランダムに選んだ 1,000 件の書き換えを行い，そこから正解の動作記述が”None”のもの（検索クエリがなくなくなるもの）を取り除いたものである．検索クエリには動作記述書き換えデータにおいて人手で作成された正解の動作記述を用いた．

評価指標: 本実験では正解の動画が必ず一つしかないことが明らかになっている Known–item search であるため，Mean Reciprocal Rank (MRR) を用いる．

検索手段: 検索結果の良否は，用いる検索エンジン・アルゴリズムにも依存しうるため，以下の検索手段における結果を比較する．

- **コサイン類似度:** 検索クエリ文，検索対象の動作記述文の文ベクトルを求め，そのコサイン類似度により候補となる動画区間をランキングする．文ベクトルを求める方法として，bert–as–service で文ベクトルを取得した場合 (BERT) と scikit–learn による TF–IDF ベクトルを用いた場合 (TF-IDF) を比較する．なお，TF–IDF の計算においては，各検索クエリと動作記述を 1 文書とした．
- **既存の検索エンジン:** 標準的な検索エンジンの一つとである Elasticsearh [4] を用いた (Elastic)．Elasticsearch におけるデフォルトのランキングは， TF，IDF，および，検索対象の長さを利用しており，検索クエリ内の単語を多く含み，文長の短いものが上位に来るようになっている．

4.2 検索実験の結果

書き換え前のキャプションを検索対象にした場合 (norm_cap) と前節で示した 3 種類の書き換え器を用いた場合の検索実験の結果を表 3 に示す．一部の場合 (BERT) を除いて動作記述を対象とする検索結果は，書き換え前のキャプションを用いる場合より劣っているが，3 種類の書き換え器の優劣は，true_tag > rand_tag > no_tag となった．

書き換えの際に，書き換え結果が得られない場合もあり，検索クエリに含まれる表現などの重要な情報が動作記述で再現されていない場合があることが検索精度が下がった全般的な原因であると考えられる．TF–IDF で低下した原因は，複文を単文群に分割しているので，同じ単語が複数回出るようになり，全体的に単語の出現回数が多くなり，IDF が小さくなったためであると考えられる．Elasticsearch で精度が低下した原因は，複文を単文群に分割しため，文長が長くなったためであると考えられる．つまり，動作記述において余分な情報を削除することよりも単文に分割する際の主語の複製の影響が大きかったと考えている．なお，true_tag システムの結果を BERT 利用で検索した場合の検索精度が向上したことから，動作記述への書き換えを学習する際には正しい動作ラベルを用いることが有効であることが再確認された．

[4]https://www.elastic.co/jp/products/elasticsearch

表 3: ラベルの使用法の違いによる動画検索精度

	norm_cap	no_tag	true_tag	rand_tag
TF–IDF	**0.951**	0.682	0.932	0.931
BERT	0.785	0.414	**0.829**	0.821
Elastic	**0.939**	0.628	0.919	0.917

5 おわりに

本論文では，キャプションの書き換えにより動画における動作に対する記述を生成する手法を提案し，その内的・外的評価について議論した．内的評価の結果 (表 1, 表 2)，および，外的評価の結果 (表 3) を比べると，書き換えシステムの良さの順がともに true_tag > rand_tag > no_tag と一致した．これは，提案指標により良いと評価される動作記述が動画検索という応用においても優れている可能性を示すとともに，動作ラベル利用の有効性を示す．一方で，外的評価である動画検索における動作記述の有用性は現時点では限定的であることが示唆され，さらに動作記述生成の精度を上げる必要があることが確認された．

謝辞

本研究は JSPS 科研費 (17H01831) の助成を受けた．

参考文献

[1] R. Krishna *et al.*, Dense-captioning events in videos., in: ICCV 2017, pp. 706–715.

[2] J. Gu *et al.*, Incorporating copying mechanism in sequence-to-sequence learning, CoRR abs/1603.06393.

[3] A. See *et al.*, Get to the point: Summarization with pointer-generator networks, arXiv preprint arXiv:1704.04368.

[4] S. Narayan *et al.*, Split and rephrase, arXiv preprint arXiv:1707.06971.

[5] C. Chu *et al.*, An empirical comparison of domain adaptation methods for neural machine translation, in: ACL 2017, pp. 385–391.

[6] F. Caba Heilbron *et al.*, Activitynet: A large-scale video benchmark for human activity understanding, in: CVPR 2015, pp. 961–970.

[7] M. A. Sultan *et al.*, Back to basics for monolingual alignment: Exploiting word similarity and contextual evidence, in: ACL 2014, pp. 219–230.

[8] M. A. Sultan *et al.*, Fast and easy short answer grading with high accuracy, in: NAACL 2016, pp. 1070–1075.