連想対話モデル: 発話文から連想した視覚情報を用いた応答文生成

石橋　陽一†　　宮森　　恒†

† 京都産業大学 コンピュータ理工学部　〒 603–8555 京都府京都市北区上賀茂本山
E-mail: †{g1445539,miya}@cc.kyoto-su.ac.jp

あらまし　本研究では, 画像入力のない対話システムにおいてより人間らしい理解に基づく応答文生成を実現するため, 言語情報から視覚情報を連想し, それを言語情報と融合的に活用することで応答文を生成する連想対話モデルを提案する. ニューラル機械翻訳の分野では, 画像と文の両方を用いて翻訳文を生成する研究があり, これらの研究では視覚情報が翻訳文生成に有益に働くことが示されている. しかし多くの対話システムでは発話内容がテキスト形式でのみ与えられるため, 画像を用いた文生成アルゴリズムを対話の応答文生成に直接利用することができない. そこで, 我々は, テキスト入力された発話文から視覚情報を生成 (連想) し, 連想された視覚情報と文の言語情報を融合した意味表現を用いて応答文を生成する手法を提案する. 連想を用いないモデルとの比較実験では, 提案手法が文に関係する視覚情報を連想することによって, 言語情報のみでは生成が困難と思われる内容を含んだ文を生成できていることを確認した. さらに, 連想した視覚情報を解析した結果, 対話の内容をより豊かにするために役立つ視覚情報を生成 (連想) していることが分かった.

キーワード　対話システム, マルチモーダル学習, 転移学習, ニューラルネットワーク, アテンションメカニズム, Encoder-Decoder モデル

1. はじめに

言語は人間が思考や意思を素早く正確に伝達するために必要な記号体系である. 人と機械の間における情報伝達の効率を考えると, 人間と同水準で対話が可能な機械 (対話エージェント) を作ることには産業的に非常に大きな価値がある. 古くは, ELIZA [1] に代表される入力発話に対するパターンマッチを用いたシステムや SHRDLU [2] を始めとする概念理解に基づいて対話を行うシステムが存在したが, 文法等の知識体系を明示的に与える必要性があった. 近年, 人間の対話から文法を学習し応答文を生成することができる Encoder-Decoder モデルが研究されている [3]. Encoder-Decoder モデルの構造は, 翻訳元文を意味表現ベクトルに符号化する Encoder と, その意味表現を用いて翻訳先文の生成を行う Decoder から構成されている. Encoder-Decoder モデルは [3] によって機械翻訳のアルゴリズムとして提案されたが, [4] は, 入力として発話文を, 出力として応答を与え学習させることで文法を獲得し, 学習データ中の知識を用いた対話が可能であることを示した. 例えば, “who is skywalker ?” という問に対して, “he is a hero .” と応答したことが報告されている [4].

しかし彼らの対話モデル (NCM) は視覚情報が必要な発話文に対して, 適切な応答ができないという問題点があった. 例えば, “how many legs does a spider have ?” という問に対して, “three , i think .” と応答したことが報告されている [4]. また, 映像には文より詳細な情報が含まれていることがある. 例えばニュースにおいて, “マラソン大会で優勝した” という文とマラソン選手が金メダルを持っている映像が提示されているとき, 文中には金メダルの情報はないが映像中には存在している. 我々はこのような詳細な情報を画像から抽出できれば “金メダル” を含む有益な文の生成が行えるのではないかと考えた.

近年, Encoder-Decoder モデルで符号化した意味表現に画像特徴量も加え翻訳文を生成した研究がある [5] [6] [7] [8] [9]. これらの研究では視覚情報が翻訳文生成に効果的に働くことが示された.

しかし, 実用的な多くの対話システムでは視覚情報を考慮せず, 入力はテキストのみである場合が多い. したがって画像入力ありの文生成アルゴリズムを対話に応用する場合, 画像等の視覚情報を入力することは不可能である. では視覚情報入力なしで視覚情報を用いるにはどうすれば良いか？

これらの問題の解決を目指し, 入力として文のテキスト情報に加え, そのテキスト情報から連想した視覚情報も用いて応答文を生成する連想対話モデルを提案する.

2. 連想対話モデル

2.1 概　　要

図 1 にモデルの概要を示す. 本研究の最終目的は, Decoder が視覚情報に基づく意味表現を必要に応じて適宜参考にすることで, テキスト情報だけでは得られない, より適切な応答文の生成を可能にすることである. ただし, 本タスクでは, 入力として与えられる情報がテキスト情報 (文) のみである. そこで, 我々は以下のアプローチを取ることでこの問題の解決を目指した.

- はじめに, 発話文が持つテキスト情報から連想によって, 文と関連した視覚情報を生成する.
(例 “マラソン大会が開かれました.” という文からテキスト情報を得て “マラソン大会で金メダルを獲得した少女” という視覚情報を生成する 図 1.)
- 次に, テキスト情報と連想した視覚情報を融合することで, 両者を必要に応じて反映した情報を得る. (例 “マラソン大

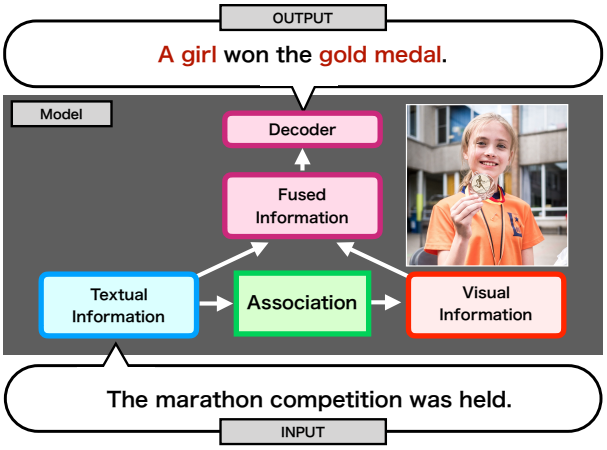

図 1: 連想を用いた文生成の仕組み．発話文を文の意味表現の符号化し，文の意味表現から対応する視覚情報を生成する．そしてそれら 2 つの意味表現を融合し Decoder に入力することで応答を生成する．

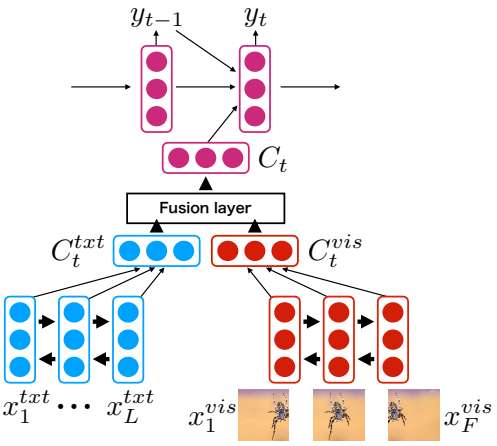

図 2: ステップ 1: テキストと動画の意味表現の学習および抽出に用いるモデル．このモデルが動画 $X_{vis} = (x_1^{vis}, ..., x_F^{vis})$ と文 $X_{txt} = (x_1^{txt}, ..., x_L^{txt})$ から C^{txt} と C^{vis} を学習した後，このモデルを用いて C^{txt} and C^{vis} を抽出する．

会”というテキスト情報と “少女・金メダル”という視覚情報を融合し “マラソン大会・少女・金メダル”という融合情報を生成する)

- 最後に，融合した情報を元に応答文を生成する．

(例 融合情報を元に，Decoder で “少女が金メダルを獲得した”という文を生成する 図 1.)

我々のアプローチはシンプルである．文と動画を入力し文生成を行う場合，Visual Encoder と Textual Encoder を用いて動画の意味表現ベクトルと文の意味表現ベクトルを獲得し文生成に利用すればよい．しかし我々の扱うタスクでは，入出力とも文であるので動画を入力することができない．そこで Visual Encoder をなくし，そのかわりに動画の意味表現ベクトルを生成する機構に付け替えるという発想である．この連想対話モデルの構成はシンプルで，Attention Mechanism を備えた Encoder と Decoder に加え，連想 Encoder からなる．連想 Encoder は，文の意味表現 (テキスト情報) から動画の意味表現 (視覚情報) を生成する RNN である (2.2.2 参照)．我々は Encoder，Decoder，そして連想 Encoder に LSTM を用いた [10]．連想対話モデルは，文の意味表現ベクトルと生成された動画の意味表現ベクトルを融合した融合意味表現を Decoder に入力することで応答文を生成する．

2.2 学習方法

学習は以下の 3 ステップからなる．最終的にステップ 3 で連想対話モデルの学習を行うために，ステップ 1 とステップ 2 において 2 つのモデルの学習をする必要がある．

ステップ 1 ： テキスト情報と視覚情報の対応関係の学習と抽出

図 2 はこのステップで使用するネットワークである．発話文 X_{txt} と，発話文に対応する動画 X_{vis} を入力とし，応答文 Y を出力する End-to-End のモデルの学習を行う．この学習では，以下の 4 つの機構を同時に学習する．

- X_{txt} を入力とし，文の意味表現ベクトル C^{txt} を出力する Textual Encoder
- X_{vis} を入力とし，動画の意味表現ベクトル C^{vis} を出力する Visual Encoder
- C^{txt} と C^{vis} を入力とし，融合意味表現 C を出力する意味表現融合層
- C を入力とし，応答文 Y を出力する Decoder および Attention

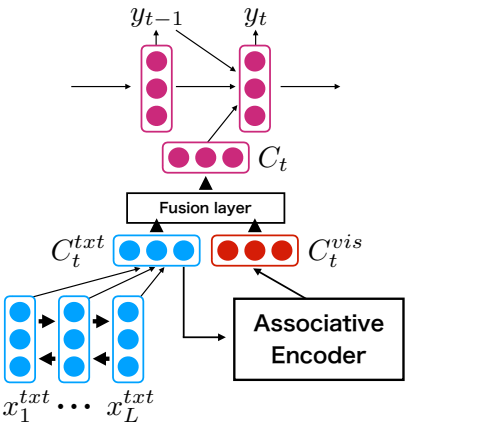

図 3: ステップ 3: 連想対話モデル．このモデルは動画を学習に用いず，代わりに連想を用いて応答文を学習する

最後に，学習を終えたこのモデルを用いて文の意味表現ベクトル C^{txt} と動画の意味表現ベクトル C^{vis} を抽出する．

ステップ 2 ： 連想の学習

このステップでは，ステップ 1 で学習済みの Textual Encoder で生成した発話文の意味表現ベクトル C^{txt} を入力とし，発話文に対応する動画の意味表現ベクトル C^{vis} を出力する連想 Encoder の学習を行う．

ステップ 3 ： 連想を用いた応答文生成

ステップ 3 ではステップ 1 の Visual Encoder をステップ 2 で学習した連想 Encoder に入れ替えたネットワークで学習を行う (図 3)．このモデルは，X_{txt} を入力とし，Textual Encoder で生

成した C^{txt} と連想生成 Encoder で生成した C^{vis} を意味表現融合層で生成した融合意味表現ベクトル C から応答文 Y を出力する応答文生成モデルである．つまり連想対話モデルの構成は Visual Encoder の代わりに連想 Encoder を用いている点以外，ステップ 1 のネットワークと同じである．意味表現融合層と Decoder および Attention はステップ 1 で学習した重みを初期化し再度学習を行う．Textual Encoder と Visual Encoder はステップ 1 で学習したモデルの重みを共有し，重み更新を行わない．なお，このモデルでは Decoder と Attention，意味表現融合層のみ学習を行う．

2.2.1 ステップ 1：テキスト情報と視覚情報の対応関係の学習と抽出

このステップで利用する学習データは，発話文，発話文に対応する動画，および，発話文に対する応答文である．テキスト情報には動作を表す表現があるので画像ではなく動画を用いた．発話文と動画が対応しているデータとして，我々は TV ニュースのデータを独自で作成し，利用した．まず，TV ニュースの動画データから，発話文 X_{txt} として，字幕データから抽出した台詞を獲得した．同様に，TV ニュースの動画データから，発話文に対応する動画 X_{vis} として，台詞が発せられた一連のシーンを画像系列として獲得した．データセットに関する詳細は 3.1 を参照．ここで，発話文 X_{txt} は，単語 x^{txt} の系列によって表現される．また，発話文に対応する動画 X_{vis} は，画像特徴ベクトル x^{vis} の系列によって表現される．我々は画像を直接入力するのではなく，代わりに学習済み CNN によって獲得しフラットにした画像特徴量を用いる．したがって，この事前学習モデルへの入力は，単語ベクトルの系列 $X_{txt} = (x_1^{txt}, ..., x_L^{txt})$ と画像特徴ベクトルの系列 $X_{vis} = (x_1^{vis}, ..., x_F^{vis})$ である．また，出力は単語ベクトルの系列 $Y = (y_1, ..., y_T)$ である．ここで，L，F，T はそれぞれ発話文の長さ，入力画像の枚数，応答文の長さを表す．

ステップ 1 のモデルは Textual Encoder，Visual Encoder，Attention を含む文の Decoder から構成されるマルチモーダル Encoder-Decoder モデルである [11]．ステップ 1 では発話文と発話文に対応する動画を Textual Encoder と Visual Encoder に入力しそれぞれの意味表現 C_t^{txt} と C_t^{vis} を得る．そして意味表現を融合した C_t を Decoder に入力し応答単語を生成する．

$$P(y \mid s_{t-1}, C_t) = softmax(W_s s_{t-1} + W_c C_t + b_s) \quad (1)$$

$$s_t = LSTM(C_t, s_{t-1}, y_{t-1}) \quad (2)$$

s_t は t 番目の出力単語を生成したときの Decoder の隠れ層である．また，W_s，W_c とバイアス b_s はモデルによって学習されるパラメータである．なお意味表現の抽出には Attention mechanism を使う [11]．Attention mechanism では応答文の単語に対応する系列の一部分の情報を強く反映した意味表現を抽出できる．例えば，文中の “蜘蛛” という単語に注目し，動画中の蜘蛛の画像に注目したとき，蜘蛛という言葉の意味表現と蜘蛛の画像の意味表現が生成される．この場合，文の意味表現と動画の意味表現には対応関係があるので，ステップ 2 で文の意味表現から動画の意味表現を予測する連想 Encoder を学習する．ステップ 1 の目的は，ステップ 2 の学習に使用するためにこれら文の意味表現と動画の意味表現を抽出することである．一般的な Encoder-Decoder モデルの Attention は式，，によって意味表現 C_t を計算する．h_i は入力系列を受けて計算された双方向 LSTM の中間ベクトル $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$ である．

$$C_t^{att} = \sum_{i=0}^{T} \alpha_{t,i} h_i \quad (3)$$

$$\alpha_{t,i} = softmax(e_{t,i}) \quad (4)$$

$$e_{t,i} = w_e^{\mathrm{T}} tanh(W_e s_{t-1} + V_e h_i + b_e) \quad (5)$$

$e_{t,i}$ は Encoder の隠れ層状態 h_i と Decoder の隠れ層状態 s_{t-1} の関連強度，$\alpha_{t,i}$ はそれを確立で表現している．また，w_e，W_e，V_e とバイアス b_e はモデルによって学習されるパラメータである．一方，マルチモーダル Encoder-Decoder モデルは意味表現 C_t^{txt} と C_t^{vis} の 2 つの意味表現を獲得する．したがってマルチモーダル Encoder-Decoder モデルは式 (3)(4)(5) の代わりに以下の式を用いる．

$$C_t^{txt} = \sum_{i=0}^{L} \alpha_{t,i}^{txt} h_i^{txt} \quad (6)$$

$$C_t^{vis} = \sum_{j=0}^{F} \alpha_{t,j}^{vis} h_j^{vis} \quad (7)$$

$$\alpha_{t,i}^{txt} = softmax(e_{t,i}^{txt}) \quad (8)$$

$$\alpha_{t,j}^{vis} = softmax(e_{t,j}^{vis}) \quad (9)$$

$$e_{t,i}^{txt} = w_{e_{txt}}^{\mathrm{T}} tanh(W_{e_{txt}} s_{t-1} + V_{e_{txt}} h_i^{txt} + b_{e_{txt}}) \quad (10)$$

$$e_{t,j}^{vis} = w_{e_{vis}}^{\mathrm{T}} tanh(W_{e_{vis}} s_{t-1} + V_{e_{vis}} h_j^{vis} + b_{e_{vis}}) \quad (11)$$

対話において視覚情報が必要ではない応答も存在する．よって Decoder 側は文の意味表現と動画の意味表現のどちらをどの程度強く参考にするかという，“尺度” を持つ必要がある．重み行列を掛けることによって動画の意味表現ベクトル C_t^{vis} と文の意味表現ベクトル C_t^{txt} の情報の参考具合を学習することが可能である (式 (12))．よって最終的に Decoder へ渡される意味表現は，文の意味表現 (テキスト情報) と動画の意味表現 (視覚情報) の参考具合を持った融合意味表現ベクトル C_t となる．

$$C_t = W_{txt} C_t^{txt} + W_{vis} C_t^{vis} + b_c \quad (12)$$

W_{txt}，W_{vis}，b_c はここで学習されるパラメタとバイアスである．この融合意味表現ベクトル C_t を用いて Decoder で次の単語を予測する．我々はこの層を意味表現融合層と呼ぶ．ステップ 1 での損失関数は以下の式で与えられる．

$$L(\Theta, \Omega, K, \Lambda, H) = -\sum_{i=1}^{B} \sum_{j=1}^{T} y_{i,j} \log \hat{y}_{i,j} \quad (13)$$

ここで，Θ は Decoder のパラメタ，Ω は意味表現融合層の

パラメタ, K は Attention の FFNN のパラメタ, Λ は Textual Encoder のパラメタ, H は Visual Encoder のパラメタで, B は学習回数の総数, そして $\hat{y}_{i,j}$ は正解ラベルが 1 の one-hot ベクトルである. マルチモーダル Encoder-Decoder モデルの訓練が完了した後, 再度訓練データを入力し文の意味表現と動画の意味表現を保存する. なお, このモデルは意味表現を抽出する際も学習する際も teacher forcing を行なっている.

2.2.2 ステップ 2 : 連想の学習

ステップ 2 では連想 Encoder に文の意味表現から動画の意味表現を予測させる学習を行う. 連想 Encoder には再帰型ニューラルネットワーク (Recurrent Neural Network: RNN) 用いる. 連想 Encoder は入力が文の意味表現ベクトルの系列 $C^{txt} = (C_1^{txt}, ..., C_t^{txt}, ..., C_T^{txt})$, 教師が動画像の意味表現ベクトル系列 $C^{vis} = (C_1^{vis}, ..., C_t^{vis}, ..., C_T^{vis})$ の回帰問題を解くモデルである (式 (14)). ここで T は応答文の長さを表す.

$$\hat{C}_t^{vis} = RNN(C_t^{txt}) \tag{14}$$

ここで $\hat{C}_t^{vis}$ は文の意味表現 C_t^{txt} から予測された動画の意味表現である. また, 損失関数は以下の式で与えられる.

$$L(\Gamma) = \sum_{i=0}^{T} (C_i^{vis} - RNN(C_i^{txt}))^2 \tag{15}$$

ここで Γ は RNN が学習するパラメータを表す. なお, 我々は RNN として長期依存関係を学習することが可能である LSTM を用いた.

2.2.3 ステップ 3 : 連想を用いた応答文生成

ステップ 3 では, 連想対話モデルが動画像のかわりに連想を用いて応答文を学習する (図 3). 連想対話モデルへの入力は単語の系列 $X_{txt} = (x_1^{txt}, ..., x_L^{txt})$ である. また出力は単語の系列 $Y = (y_1, ..., y_T)$ である.

連想対話モデルではステップ 1 の事前学習モデルの Visual Encoder を連想 Encoder に変更する. よって連想対話モデルのアーキテクチャは, ステップ 1 での式 (7) を 式 (14) に置き換えたものである. つまりステップ 1 で行っていた言語と視覚情報を融合するという仕組みは変えず, 発話文の意味表現から動画像の意味表現を連想 Encoder によって予測し, 連想 Encoder で生成した連想表現と発話文の意味表現を Decoder に入力し Decoder, Attention そして意味表現融合層 (式 (12)) を学習させる. なお, Decoder, Attention そして意味表現融合層の重みは初期化を行い, ステップ 1 の事前学習モデルの重みを利用しない. Textual Encoder と連想 Encoder では重み更新を行わずステップ 1 の事前学習モデルのパラメータを利用する. このように Decoder と Attention と意味表現融合層を再学習する理由は, 連想した意味表現がステップ 1 で生成された動画の意味表現とは異なった性質を持つためである. したがってステップ 3 での損失関数は以下の式で与えられる.

$$L(\Theta, \Omega, K) = -\sum_{i=1}^{B} \sum_{j=1}^{T} y_{i,j} \log \hat{y}_{i,j} \tag{16}$$

我々は, 意味表現の符号化以外の部分を再学習することで, 連想した視覚情報を用いた応答文生成が可能になると考えた.

3. 実　　験

この章で, 我々は連想対話モデルと連想を用いないモデルとの比較結果について述べる. ベースラインには文 X_{txt} を入力として文 Y を生成する Attention 付きの Encoder-Decoder モデルを用いる [11]. つまりベースラインモデルは, 提案モデルから連想の機構を除去したモデルといえる. 我々は提案モデルとベースラインモデルの両方に同じデータを学習させ, 連想が文生成にどのような効果をもたらすのか調べた. 人による生成文の評価実験 (3.3 参照) では, 連想の有無によって有益または文脈に依存した文生成の精度に差があることが判明した. さらに, 生成文の比較 (3.4 参照) を行い, 連想によって生成された文の定性的評価を行った. その結果, 連想が具体的な情報を持つ文の生成に効果的に働いていることが示唆された. また, 連想された視覚情報を可視化し, 連想対話モデルが文生成に有効な視覚情報を獲得しているかを調べた (3.5 参照). その結果, 我々の提案モデルは発話文に関係した視覚情報を連想していることが分かった.

3.1 データセット

我々は独自に収集した以下のデータを用いてモデルの学習を行った.

- TV ニュースの字幕と, その字幕が表示された場面の映像 (動画)

なお, 句点 “。”や “!”, “?” までを 1 つの文と仮定し, この区切り方式で文に区切り 1 つの文として扱った. ステップ 1 のモデルの Visual Encoder は画像の系列を符号化する. 画像の系列はフレームレート 5fps で動画から切り出した画像の集合である. 次に, それらの画像を学習済み畳込みニューラルネットに入力し, 最後の畳み込み層の出力を画像特徴量として用いることとした. この画像特徴量は畳込みニューラルネットに VGG16 を用いて抽出した [12]. 獲得した画像特徴量の次元は 25088 次元である. ニュースは 2016 年 12 月から 2017 年 2 月までに取得した日本語のニュース 163 番組. 対話数は 38K 対話, 語彙は 19K 単語である. データは学習データ用に 34K 対話, テストデータ用に 4K 対話に分けられた.

3.2 パラメータ設定

ステップ 1 でのマルチモーダル Encoder-Decoder モデルの構成は, 2 つの Encoder および Decoder に 1 層の LSTM を用いた. Encoder の隠れ層次元数を 512 次元に, 融合意味表現の次元を 1024 次元に, Decoder 隠れ層次元数を 1024 次元に設定した. またバッチサイズは 64 に設定した.

ステップ 2 での連想 Encoder は 4 層の LSTM を用いた. 隠れ層次元数は 4 層すべて 1024 次元, 最適化手法は adam, バッチサイズは 64 に設定した.

ステップ 3 での連想対話モデルは, 文の Encode の隠れ層次元数を 512 次元に, 融合意味表現の次元を 1024 次元に, Decoder の隠れ層次元数を 1024 次元に設定した. またバッチサイズは 64 に設定した. また, ステップ 3 において, Textual Encoder の重みとバイアスはステップ 1 での重みを用いて, 重みを更新しないようにしている. 連想 Encoder も同様に, 重みを更新しな

表 1: ニュース文テストデータでの精度

Model	Accuracy (News test data) [%]
Seq2Seq	8.228
Seq2Seq-Assc	8.109

いようにしている．また, Decoder や Attention や融合意味表現を獲得する意味表現融合層の重みとバイアスはステップ 1 での重みを用いず, 重みを初期化して学習を行った (式 (2), (1), (10), (11), (12) 参照).

ベースラインモデルの構成は, 1 つの Encoder と 1 つの Attention つき Decoder である [11]．ベースラインモデルは連想 Encoder も Visual Encoder も含まない．Encoder と Decoder ともに 1 層の LSTM を用いた．Encoder の隠れ層次元数を 512 次元に, Decoder 隠れ層次元数を 1024 次元に設定した．

ベースラインのステップ 1 と, 提案手法のステップ 1 とステップ 3 では最適化手法として adagrad を用いた．またバッチサイズは 64 とした．また, 単語を 512 次元の埋め込み表現にして Textual Encoder に入力した．

3.3 定量的評価

提案モデル (*Seq2Seq-Assc*) とベースラインモデル (*Seq2Seq*) のニュース文テストデータにおける精度 (生成単語と正解単語の平均正答率) は, 提案モデル (*Seq2Seq-Assc*) が 8.109%, ベースラインモデル (*Seq2Seq*) が 8.228%であった (Tab. 1)．しかし対話にはあらゆる応答が考えられ, テストデータの応答文のみが正しいとは限らないため, テストデータの精度からそのモデルが一貫性のある文または発話者にとって有益な文を生成したかどうかは評価できない．そこで我々は, 提案モデル (*Seq2Seq-Assc*) とベースラインモデル (*Seq2Seq*) の比較のため, 6 人の評価者による生成文の評価を行った．評価手法として NTCIR13 STC-2 タスクでの評価方法を用いた [13]．この評価方法では, 複数の評価者によって 4 つの観点 (流暢性・一貫性・文脈依存性 ・有益性) に 0 から 2 のスコアが振られる．

そして最終的に以下の方法によってラベル L0,L1,L2 が付与される．なお, NTCIR13 STC-2 では最終的なラベルが付与するために Rule-1 と Rule-2 の 2 つの方法が用いられたが, 我々は Rule-1 を用いてラベル付与を行った [13] (Listing 1).

Listing 1: Rule-1

```
IF fluent & coherent = 1
        IF context-dependent & informative = 2
                THEN L2
        ELSE L1
ELSE
        L0
```

[13] と同じく以下の式に基づき精度 $Acc_G@k$ を求めた.

$$Acc_G@k = \frac{1}{nk}\sum_{r=1}^{k}\sum_{i=1}^{n}\delta(l_i(r) \in G) \tag{17}$$

$l_i(r)$ は i 番目の発話者が r 番目の応答候補に付与されたラベルである．n は 1 応答文に評価者らが付与したラベルの数である ($n = 7$). G は評価対象のラベルである ($G = \{L2\}$ または $G = \{L2, L1\}$). k は 1 発話あたりの応答候補文の数である．本実験ではモデルが 1 発話あたり 1 つの応答を生成するため, $k = 1$ である．したがって $Acc_{L2}@1$ は, ある文の応答候補 1 番目の応答文に付与された $L2$ ラベルの平均数を表す．この実験では, 評価用データとしてニュース字幕と一般的な事実を問う質問 (例: ドナルド・トランプ大統領の髪の毛の色は?) を含む対話文をそれぞれ 50 文用いた．

また, 提案モデル (*Seq2Seq-Assc*) が生成した文の精度とベースラインモデル (*Seq2Seq*) が生成した文の精度に差はないと帰無仮説を立て, 有意水準 $\alpha = 0.05$ で二群の平均値の差の t 検定 (両側検定) を行った．

表 (2) は評価結果, 表 (3) は 2 標本 t 検定の結果である．提案モデル (*Seq2Seq-Assc*), ベースラインモデル (*Seq2Seq*) ともにニュースデータでの精度が対話文データでの精度を上回っている (表 (2))．これは, 対話文とニュース文で用いられる表現は異なるため, ニュース文から学習された 2 つのモデルはいずれも対話文の流暢性を満たすような文を生成することが困難であるという事実が原因であると考えられる．ニュース文での精度は従来手法よりも提案手法が良い．$MeanAcc_{L2}@1$ では, 提案手法は従来手法より高い精度を示した．さらに 2 標本 t 検定の結果 (表 (3)), 提案手法と従来手法の $MeanAcc_{L2}@1$ に差はあると予想される．以上の結果は, 有用性と文脈依存性のある多くの応答文が提案手法により生成されたことを示している．一方, 対話文に対する $MeanAcc_{L1,L2}@1$ は提案手法よりも従来手法の方が高かったが, $MeanAcc_{L2}@1$ は両モデル共にほぼ 0.0 であった．これは連想によってニュース文への過学習が起きた結果, 一貫性を満たす文を生成できなかったことが原因として考えられる．

3.4 定性的評価

図 (4) は未学習データに対して提案手法とベースラインが生成した文と連想された視覚情報に類似している画像を示している．この結果は, 提案手法がベースラインよりも, 有益な情報を持つ文を生成していることを示している．例えば, 図 (4) の左の例では, 発話文 “14 日と 15 日は大学入試センター試験です.” に対して, 提案手法は “雪” という具体的な天気予想を生成している．なお, これは事実であり, ニュースデータの中にはこの試験の会場で雪が降っている映像がある．ここで重要な点は, この雪という単語は発話文のみからでは容易に生成できず, 雪の画像を連想して初めて生成された単語であるという点である．さらに “広がっ” という単語を生成したときには, 実際にこの試験会場で雪が降っている映像が連想されていた．一方で, ベースラインは “西日本や東日本にもなる, も大規模な火災ができます.” と, 火災等の誤った情報を含んでいる．図 (4) の右の例では, 発話文 “さあ, きょうまずは, フィギュアスケート全日本選手権.” に対して, 提案手法は “金メダル” という単語を含んでいる．

この例の他にも, 連想を持つ提案手法が, 連想を持たないベー

(注1)：出典: 左画像: “NHK ニュース 7”, NHK, 2017 年 1 月 11 日, 右画像: “NHK ニュース 7”, NHK, 2017 年 2 月 23 日

表 2: 人による, 提案モデル (*Seq2Seq-Assc*) とベースラインモデル (*Seq2Seq*) の生成文の評価結果

Model	**News**		**Dialog**	
	Mean Acc_{L2}@1	Mean $Acc_{L1,L2}$@1	Mean Acc_{L2}@1	Mean $Acc_{L1,L2}$@1
Seq2Seq	0.066	0.429	**0.01**	**0.251**
Seq2Seq-Assc	**0.109**	**0.503**	0.00	0.180

表 3: 提案モデル (*Seq2Seq-Assc*) とベースラインモデル (*Seq2Seq*) の平均精度の差の t 検定 (両側検定, 有意水準 $\alpha = 0.05$) の結果

データ	**News**		**Dialog**	
検定対象	Mean Acc_{L2}@1	Mean $Acc_{L1,L2}$@1	Mean Acc_{L2}@1	Mean $Acc_{L1,L2}$@1
結果	棄却	採択	採択	棄却

入力文	14日と15日は、 大学入試センター試験です。	さあ、きょうまずは、フィギュアスケートの全日本選手権。
生成文 **(Baseline)**	西日本や東日本にもなる、 も大規模な火災ができます。	女子シングルで4連覇を狙うで、 日本選手権の選手が、 大会に出場しました
生成文 **(ACM)**	雪の風の予想が 広がっています。	女子シングルで3した**金メダル** を獲得している選手。
入力文から連想された視覚情報と類似度の高い画像		
連想を用いて生成された単語	**雪**	**金メダル**
Cos 類似度	**0.333**	**0.344**

図 4: 提案手法と従来手法の比較結果の一例 (注1). 図中の画像は連想対話モデルが発話文から生成した連想表現ベクトル $\hat{C}^{vis}$ と最もコサイン類似度の高い意味表現ベクトル C^{vis} に対応する画像 ($x^{vis}_{d,l}$) である. 右の例では, 発話文に “フィギュアスケート”, “選手権”が含まれているため, ある選手がスケートの選手権で金メダルを獲得した場面を連想している. しかしフィギュアスケートの選手権をスピードスケートの選手権と誤解して連想している.

スラインよりも具体的な情報を持つ有益な文を生成している例を発見した. 我々はこれらの結果から, 連想が具体的な情報を持つ文の生成に効果的に働いていると推測した. これを実証するため, 我々は連想 Encoder がどのような視覚情報を生成しているについての解析を行った.

3.5 連想の解析

図 (4) は連想の解析結果である. これらの結果は, 提案モデルが発話文に関係する視覚情報を連想していることを示している. 図中の上の画像は, 発話文から提案手法が連想によって生成した連想表現と, 最も類似度が高い実際の画像である. 我々は, ステップ 1 のモデルに学習データの動画 (画像の系列) を入力し, Visual Encoder によって生成された動画の意味表現 C^{vis} と連想表現 $\hat{C}^{vis}$ をコサイン類似度で測った (式 (18)). ここで $C^{vis}_{i,j}$ はモデルの Visual Encoder に i 番目の学習データ $X^{vis}_i = (x^{vis}_{i,1}, ..., x^{vis}_{i,F})$ を入力し,j 番目の出力単語 $y_{i,j}$ を得る際に計算された動画の意味表現ベクトルを表す.

$$similarity(\hat{C}^{vis}, C^{vis}_{i,j}) = \frac{\hat{C}^{vis} \cdot C^{vis}_{i,j}}{\|\hat{C}^{vis}\| \cdot \|C^{vis}_{i,j}\|} \tag{18}$$

そして我々は, 計算された類似度の中で最も類似度が高い動画の意味表現 $C^{vis}_{d,t}$ と対応する入力画像 $x^{vis}_{d,l}$ を, 連想対話モデルが連想した視覚情報と仮定した. したがって図 (4) 中の画像は, 文から連想した視覚情報を学習データ中の画像を用いて可視化したものである. 学習データの識別番号 d および出力単語番号 l の決定方法は式 (19)(20) として定式化できる.

$$(d, t) = \arg\max_{i \in \Phi}(similarity(\hat{C}^{vis}, C^{vis}_i)) \tag{19}$$

$$l = \arg\max_{i \in [1,F]}(\alpha^{txt}_{d,t,i}) \tag{20}$$

上の式における (19) 中の Φ は学習データの識別番号の集合 $A = \{d \mid 1 \leqq d \leqq D\}$ と出力単語数の集合 $B = \{t \mid 1 \leqq t \leqq T\}$ の直積集合 $A \times B = \{(d, t) \mid d \in A, t \in B\}$ である. ここで D は学習データの総数, T は応答文の長さ (出力単語の数) を表す. また, 式 (20) 中の $\alpha^{txt}_{d,t,i}$ は d 番目の学習データを入力し,t 番目の出力単語 $y_{d,t}$ を得る際に計算された Visual Encoder への Attention 荷重のうち, k 番目の隠れ状態 h^{vis}_k への荷重を表す.

図 (4) の結果は, 連想対話モデルが発話文の内容と一致する視覚情報の連想に成功していることを示している. 例えば, 図 (4) の右の例はフィギュアスケートをスピードスケートと誤解しているものの, “スケートの選手権”の話題に対して, スケート選手が金メダルを獲得している場面を連想している. また, その連想結果から “金メダル”という単語を生成している. 連想された画像には金メダルをもつ選手と, その後ろにスケートをしている選手らが写っている. 図 (5) は単語 “金メダル”が生成された際の Attention の荷重を可視化したものである. “フィギュア”という単語には注目されず “スケート”と “選手権”に注目されていることから, 図 (4) のような連想結果になったと考えられる.

我々は, このような, 発話文と一致する画像が連想された例を複数発見した. それらの例の一部は, 付録 (7. 参照) で紹介している. また, それらの例はどれも連想なしでは生成が困難であると思われる単語が生成できていることを示している. 例え

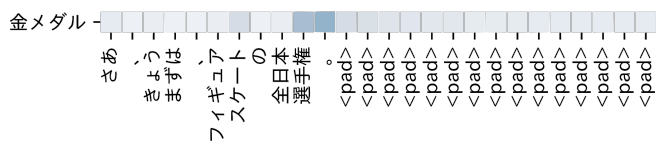

図 5: 単語 “金メダル”が生成された際の Attention の荷重 (出力単語への入力単語の関連確率) を可視化したヒートマップ. 下の入力単語中の色の濃い箇所 (“スケート”, “選手権”, “。”) に注意が払われ図 (4) 右のような画像が連想された結果, 単語 “金メダル”が生成されたことを表している.

ば, “14 日と 15 日は大学入試センター試験です.” という文からは “雪” という単語を生成することは, 実際に雪が降ったという連想がなければ生成することは難しい.

4. 関 連 研 究

[14] は共有された画像についてのソーシャルメディア上の対話を想定し, 画像と文から自然な対話を行うモデルを提案した. 我々は対話を行う際には画像を用いないケースを想定している. 文生成アルゴリズムでは, 視覚情報とテキスト情報を同時に与えることで有益な応答文を得ようとする試みがある [5] [6] [7] [8] [9]. 訓練時にのみ視覚情報を用いるアプローチは [5] や [7] のアプローチがあり, 我々のアプローチと似ている.

[5] は, テキストと画像から潜在変数をモデリングし, その潜在変数を用いて文を生成するモデルを提案した. 彼らのモデルは通常の Attention 付き Encoder-Decoder に加え, 潜在変数を生成する機構がある. 文を Encoder に入力し, Decoder で翻訳文を予測さるとき, ソース文・ターゲット文・画像から生成した潜在変数を Decoder の隠れ層に使ってターゲット単語を生成する. 次に潜在変数を Decoder に組入れ, 最後にターゲットを出力する.

[7] は, 文生成と画像特徴生成の 2 種類の出力を持つマルチタスク型学習フレームワークを用いて意味表現を学習し, Decoder の翻訳文生成に使うことで精度が上がることを示した. 彼らのモデルの構造は, Textual Encoder と文の Decoder, 画像の Decoder からなり, 文を入力し文の Decoder で翻訳文を予測させる. 同時に, Encoder の隠れ層を平均して, 画像の Decoder に入力し, 画像特徴を予測させる.

我々のアプローチと彼らのアプローチ大きな違いは 3 つある. 第一に, 彼らのアプローチでは訓練時の視覚情報として画像を与えているが, 我々のアプローチではそれを拡張し動画を与えている. この拡張によって我々は, モデルが動作を表す語と視覚情報との対応付けを学習することが可能になると考えている. 第二に, 我々は翻訳タスクではなく対話タスクを扱っている. 第三に, 我々の扱うタスクでは, 一般的な対話のような, 文とそのときの視覚情報が必ずしも対応していないデータから知識抽出等の学習を行うことを想定している. このため, 我々のアプローチは視覚情報とテキスト情報の使用度合を学習するノイズ除去機構を備えている式 (12) .

[7] とのもう 1 つの違いは, 我々の手法ではマルチタスク学習を採用していないため視覚情報を入力している点である. したがって訓練時に視覚情報への Attention を行うことが可能であるので, 応答単語に応じて視覚情報を抽出できる.

5. 課　　題

本研究では, 連想によって有益な文の生成が行われていることを示したが, 実験結果より幾つかの課題も示唆された.

まず, 提案モデルは発話文を誤解して視覚情報を連想していることがあった (3.5 参照). 例えば先程の例では, フィギュアスケートについての話題に対して, スケート選手が金メダルを獲得している場面を連想している. これについては学習データを増やすことで改善できるだろう.

また, 対話文を入力した場合に, 提案手法よりも従来手法が $MeanAcc_{L2,L1}@1$ が高かった (3.3 参照). これは提案モデルが流暢性および一貫性を満たさない文を多く生成したことを示している. この原因として, 連想を用いる場合, 連想された意味表現 C^{vis} をテキスト意味表現 C^{txt} と融合する意味表現融合層において, ニュース文および連想された意味表現への過学習が起きた結果, 一貫性を満たす文を生成できなかったことが考えられる. 過学習の原因はデータ量の少なさによるものと思われるため, 学習データを増やすことで改善できると考えられる.

6. 結　　論

ニューラル機械翻訳の分野において画像と文の両方を用いて翻訳文を生成する研究では画像が翻訳文生成の際に効果的に働くことが示されていた. 近年, テキスト入力-テキスト出力タイプの翻訳文生成モデルは対話モデルとしても利用されている一方で, 多くの対話システムでは発話内容がテキスト形式でのみ与えられるため, 上述したような画像を用いた翻訳文生成アルゴリズムを対話の応答文生成に直接利用することができないという問題点があった. そこで我々は, 発話文から視覚的意味表現ベクトルを生成 (連想) し, 連想された視覚的意味表現ベクトルと文の意味表現ベクトルを融合した意味表現ベクトルを用いて応答文生成を行う手法を提案した. 提案手法 (連想あり) と従来手法 (連想なし) との比較実験では, 連想を用いないモデルよりも連想を用いる提案モデルの方が言語情報のみでは生成が困難と思われる内容を含んだ文を生成できていることを確認した. また, 連想対話モデルが生成 (連想) した視覚情報を解析した結果, 対話の内容をより豊かにするために役立つ視覚情報を生成していることが分かった.

文　　献

[1] Joseph Weizenbaum. ELIZA - a computer program for the study of natural language communication between man and machine. *Commun. ACM*, Vol. 9, No. 1, pp. 36–45, 1966.

[2] Terry Winograd. Procedures as a representation for data in a computer program for understanding natural language. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE PROJECT MAC, 1971.

[3] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Con-*

ference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pp. 3104–3112, 2014.

[4] Oriol Vinyals and Quoc V. Le. A neural conversational model. *CoRR*, Vol. abs/1506.05869, , 2015.

[5] Joji Toyama, Masanori Misono, Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. Neural machine translation with latent semantic of image and text. *CoRR*, Vol. abs/1611.08459, , 2016.

[6] Iacer Calixto, Qun Liu, and Nick Campbell. Doubly-attentive decoder for multi-modal neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pp. 1913–1924, 2017.

[7] Desmond Elliott and Ákos Kádár. Imagination improves multimodal translation. *CoRR*, Vol. abs/1705.04350, , 2017.

[8] Hideki Nakayama and Noriki Nishida. Zero-resource machine translation by multimodal encoder-decoder network with multimedia pivot. *Machine Translation*, Vol. 31, No. 1-2, pp. 49–64, 2017.

[9] Amrita Saha, Mitesh M. Khapra, Sarath Chandar, Janarthanan Rajendran, and Kyunghyun Cho. A correlational encoder decoder architecture for pivot based sequence generation. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pp. 109–118, 2016.

[10] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[11] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, Vol. abs/1409.0473, , 2014.

[12] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, Vol. abs/1409.1556, , 2014.

[13] Lifeng Shang, Tetsuya Sakai, Hang Li, Ryuichiro Higashinaka, Yusuke Miyao, Yuki Arase, and Masako Nomoto. Overview of the NTCIR-13 short text conversation task. In *Proceedings of NTCIR-13*, 2017.

[14] Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P. Spithourakis, and Lucy Vanderwende. Image-grounded conversations: Multimodal context for natural question and response generation. *CoRR*, Vol. abs/1701.08251, , 2017.

7. 付　　録

図 (6)(7) は連想を行う提案モデル (*ACM*) と連想を行わない従来モデル (*Baseline*) に同一の発話文を入力したときに生成された応答文の比較結果と提案モデル (*ACM*) が連想した視覚情報 $\hat{C^{vis}}$ と類似度の高い画像を示したものである. 図 (6) 左の例では, 発話文から横綱を連想し出力単語として “横綱” を生成しているが, 画像の人物を同じ横綱である “白鵬” と誤解している. 図 (6) 右の例では, 発話文から野球選手 (投手) を連想し, 選手という単語を生成してるが, 発話文中にある “大谷” 選手ではない選手を連想している. 図 (7) 左の例では, 日本全域が高気圧に覆われている場面を連想しており, 出力単語として “晴れ” を生成している. 図 (7) 右の例では, 自転車に乗った人が道路付近で強い風に煽られている場面を連想しており, 出力単語として “交通” と “暴風” を生成している.

入力文	大相撲初場所は2日目です。	投手では、 日本ハムから大谷投手を含む3人が 選ばれています。
生成文 **(Baseline)**	1位は1メートルでの 1位の1位です。	今シーズンはプロに野球 が出ています。
生成文 **(ACM)**	きょうは初優勝を目指す **横綱**・白鵬です。	こちら の スポーツ **選手** 。
入力文から連想された視覚情報と類似度の高い画像		
連想を用いて生成された単語	**横綱**	**選手**
Cos 類似度	**0.336**	**0.297**

図 6: スポーツに関する応答と連想された画像 (注2)

入力文	本州付近は、移動性高気圧に広く 覆われる見込みです。	発達中の低気圧の影響で、 あすの朝にかけては、西日本や東日本の 日本海側を中心に、 所によって吹雪となる見込みです。	
生成文 **(Baseline)**	では、あすの各地の天気です。	気象庁は、暴風や高波に警戒 が必要です。	
生成文 **(ACM)**	あすの朝には、西日本、東日本から 西日本の日本海側では、 **晴れ**ている所が多くなりそう	気象庁は、**暴風**や猛吹雪、 吹きだまりによる**交通**への 影響も注意してください。	
入力文から連想された視覚情報と類似度の高い画像			
連想を用いて生成された単語	**晴れ**	**交通**	**暴風**
Cos 類似度	**0.749**	**0.338**	**0.280**

図 7: 天気に関する応答と連想された画像 (注3)

(注3) : 出典:
左画像: “NHK ニュース 7”, NHK, 2017 年 1 月 13 日放送
, 右画像: “ニュースウォッチ 9”, NHK, 2017 年 2 月 17 日放送
(注3) : 出典:
左画像: “ニュースウォッチ 9”, NHK, 2017 年 2 月 27 日放送
, 右画像: “ニュースチェック 11”, NHK, 2017 年 2 月 20 日放送