シチュエーションに応じたテキストのインタラクティブ入力支援手法の評価

土屋 潤一郎[†1]　入江 英嗣[2]　坂井 修一[2]

概要：本研究では，マルチモーダルなモデルを活用して，画像と関連するテキストの柔軟な入力支援を行うことができることを示す．情報技術の発達により，瞬間的な感情や思考の，個人の自己表現としてのマルチメディアでの即時的な情報の記録・発信が増加している．個人から発する自由な表現を損ねない柔軟性を持ちながら，非言語メディアによって高精度化されたテキスト入力支援があればこの情報交換を促進できる．本研究では，畳み込み型ニューラルネットワークによって画像の特徴量を取り出し，それを利用した再帰型ニューラルネットワークを言語モデルとしてインタラクティブに利用し，システム使用者の表現の選択に対して柔軟性を確保した．Windows 用のキーボードを用いることを想定した実験では，本研究が提案する画像情報を入力とする入力支援システムがキーストロークを最大 16%超削減した．

Input Method for Text According to Situations

JUNICHIRO TSUCHIYA[†1]　HIDETSUGU IRIE[2]　SHUICHI SAKAI[2]

1. はじめに

情報技術の発達により，同じ事象に関連する画像や動画といった非言語情報によるメディアとテキストとの作成が大衆化している．さらにモバイル端末やウェアラブル端末が従来のデジタルデバイスよりも人々の生活に密着しはじめており，情報の記録も発信も即時的に行うことが可能になった．これらの技術の発達は，個人の刹那的な感情や思考を記録する機会を促進し，それらはミニブログや SNS などを通じて自由な自己表現として共有され，他者との交流に利用されるようになった．モバイル端末やウェアラブル端末で，対象の客観的な記述でない，個人から発される自己表現の一部であるテキストを入力する頻度は今後も増加することが見込まれる．このようなテキストの入力を高速化・省力化できるような柔軟性の高い支援があれば，人々の自己表現の発信コストをさらに低下させ，他者との交流を刺激することになるであろう．

そこで本研究は，このような，ある非言語情報とそれにに対する客観的説明でない要素を含む言語情報との作成に関して，非言語情報の中から画像を入力として言語情報の作成を高速化・省力化するテキスト入力支援を目指す．

画像を用いた言語モデルの構築については，画像キャプションの自動生成の分野が，近年，ニューラルネットワークを取り入れて成果を上げている．画像キャプションの自動生成は，テキストのフレーズ単位の予測とみなすこともできるが，画像について計算機が正確に説明できるかどうかを主眼とされるため，自動生成文（とそのための学習データ）は極めて説明的であり，かつ出力は文の単位で行われるものがほとんどである．したがって，従来の画像キャプションの自動生成は，そのままでは本研究の目的である自由な自己表現に対する支援を満たせないため，本研究ではこのようなモデルのインタラクティブな利用を新たに提案する．

本研究の提案するシステムは，単語の予測と補完とによって打鍵回数の削減を主とするテキストの入力支援を行う．単語の補完とは，使用者が単語を入力するときに，システムが次の単語として完成すると想定しうる単語を提案するタスクであり，本研究では使用者が 1 文字を入力する前に次の要素（文，フレーズ，単語など）の候補を提示する予測とともに，本研究が提案する入力支援で用いられる．

1　東京大学工学部電子情報工学科
2　東京大学大学院情報理工学系研究科
†1　現在，東京大学大学院情報理工学系研究科

また，使用者が単語の選択について明確な意思を持っていた場合，単語の予測と補完とは，文の単位での予測・補完よりも，使用者の意図する入力をより柔軟に，あるいはより完全に反映させることができる．本研究は，テキスト入力の高速化・省力化と入力支援システムが自己表現としてのテキストに求められる柔軟性を損なわないこととを両立するため，単語の単位で予測・補完を行うものとする．

非言語情報を入力とする言語モデルを利用すれば，言語だけに基づくモデルを利用した際よりも精度が高く，インタラクションによりシステムを使用して生成されるテキストに柔軟性を確保できる．このような利点は，自由な自己表現以外にも，医療，介護，教育などの流動的なシチュエーションにおける自動生成の難しい記録文書などの作成の効率化にも資するであろう．

以下，第 2 章では画像キャプションの自動生成と，ニューラル言語モデルを用いた単語の予測・補完について述べ，第 3 章では人間の表現と自動生成キャプションとに関する事前調査と，本研究の提案手法とについて述べる．第 4 章では提案手法の評価を行い，最後に第 5 章で本研究についてまとめ，今後について述べる．

2. 関連研究

2.1 ニューラルネットワーク

ニューラルネットワークとは，脳の神経細胞とその結合とを有向グラフとして数学的に定式化したものである．複数のユニットを組み合わせることで入出力の複雑な関係を学習できるため．近年さまざまな分野で利用されている．ここでは，画像処理の分野で成功を収めている畳み込みニューラルネットワーク，並びに自然言語などの系列データの処理に用いられる再帰型ニューラルネットワークを用いた言語モデル，及び再帰型ニューラルネットワークでよく用いられるユニットである Long Short-Term Memory について説明する．

2.1.1 畳み込みニューラルネットワーク

いらないくさい畳み込みニューラルネットワーク (Convolutional Neural Network, CNN) は，それぞれのニューロンが次の層のニューロンにそれぞれ接続されている全結合層の他に，畳み込み層と，プーリング層からなるニューラルネットワークである．畳み込み層では，あるサイズのフィルタを入力領域全体に対してスライドさせながらフィルタと入力との内積をとる．プーリング層では，入力をサブサンプリングする．多くの場合全体ではなくサイズを指定した窓の中で最大の値をとることが行われる．ここでは，本研究でその応用を行う [20] で用いられている，ResNet-50[5] についてそのネットワーク構成の概略図を図 1 に示す．ResNet は，通常のネットワークのようにあるいくつかの層で構成されるブロックによる変換 $\boldsymbol{F}(\boldsymbol{x})$ を次の層にそのまま渡すのではなく，ブロックへの入力 $\boldsymbol{x}$ を

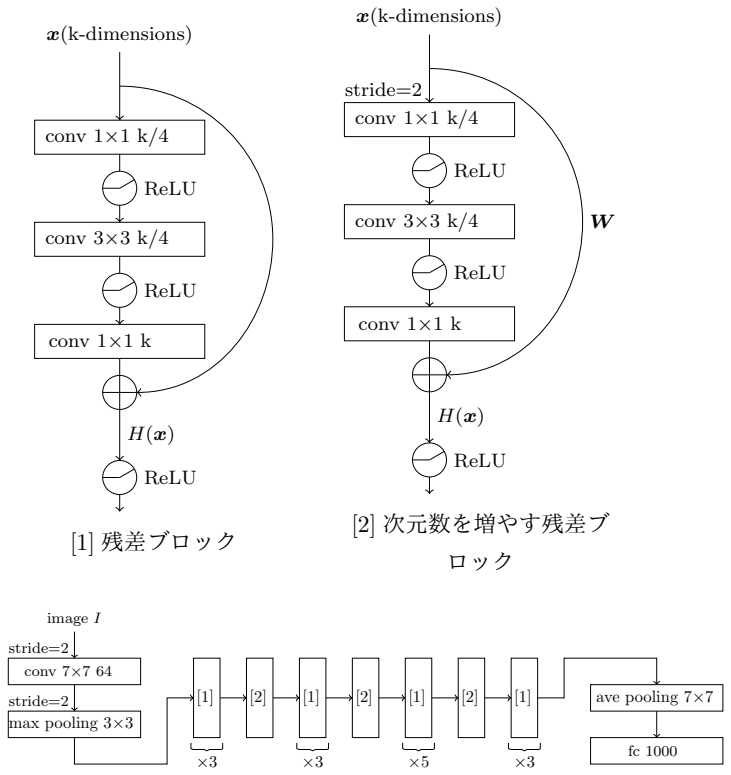

図 1 ResNet-50 に用いられている残差ブロックとネットワーク概略図

ショートカットし，$\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{x}$ を次の層に渡す，という処理が入るブロックから構成される（このブロックを残差ブロックと呼ぶ）．ブロックの入出力で次元が違う際は，線形写像 $\boldsymbol{W}$ を用いて $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}) + (\boldsymbol{x})$ とする．ショートカットを行うことで勾配が保存され次のブロックに伝わるため，誤差逆伝播法 [15] における勾配消失問題 [2] に対して有効で，非常に層数の多いネットワークを構築できるようになった．

2.1.2 再帰型ニューラルネットワーク

再帰型ニューラルネットワーク (Recurrent Neural Network, RNN) はネットワーク中にフィードバックのためのループ構造を有するニューラルネットワークである．これにより，時間的な情報を文脈として利用でき，かつ固定長でない任意の長さの情報を取り扱うことができる．隠れ層のパターンをフィードバックに使用する単純な方法で，RNN を用いた言語モデル [12] で利用されている ElmanNet[3] を図 2 に示す．時刻 t での入力 $\boldsymbol{x}_t \in \mathbb{R}^{I\times 1}$ から，出力 $\boldsymbol{y}_t \in \mathbb{R}^{O\times 1}$ を，

$$\boldsymbol{s}_t = f\left(\boldsymbol{U}\boldsymbol{x}_t + \boldsymbol{V}\boldsymbol{s}_{t-1} + \boldsymbol{b}\right) \quad (1)$$

$$\boldsymbol{y}_t = g\left(\boldsymbol{W}\boldsymbol{s}_t + \boldsymbol{c}\right) \quad (2)$$

とする．但し，$\boldsymbol{s}_t \in \mathbb{R}^{H\times 1}$ は隠れ層の状態ベクトル，$\boldsymbol{U} \in \mathbb{R}^{H\times I}$，$\boldsymbol{V} \in \mathbb{R}^{H\times H}$，$\boldsymbol{W} \in \mathbb{R}^{O\times H}$ はそれぞれ重み行列であり，$\boldsymbol{b} \in \mathbb{R}^{H\times 1}$ と $\boldsymbol{c} \in \mathbb{R}^{O\times 1}$ とはバイアスベク

トル，f と g とは活性化関数である．

2.1.3 Long Short-Term Memory

Long Short-Term Memory（LSTM）[6] は RNN を構成するユニットとしてよく用いられるものであり，単純な RNN で問題であった誤差逆伝播法 [15] における勾配消失問題 [2] に対して有効な対処法として提案された．ここでは，本研究でその応用を行う [20] で用いられている，[6] に忘却ゲートを追加した LSTM[4] を図 3 に示す．LSTM ベースの RNN では，中間層を図 3 のような構造に置換する．時刻 t での入力 $\boldsymbol{x}_t \in \mathbb{R}^{I\times 1}$ から，出力 $\boldsymbol{y}_t \in \mathbb{R}^{O\times 1}$ を，

$$\boldsymbol{i}_t = \sigma\left(\boldsymbol{W}_{ix}\boldsymbol{x}_t + \boldsymbol{W}_{iy}\boldsymbol{y}_{t-1} + \boldsymbol{b}_i\right) \tag{3}$$

$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{W}_{fx}\boldsymbol{x}_t + \boldsymbol{W}_{fy}\boldsymbol{y}_{t-1} + \boldsymbol{b}_f\right) \tag{4}$$

$$\boldsymbol{o}_t = \sigma\left(\boldsymbol{W}_{ox}\boldsymbol{x}_t + \boldsymbol{W}_{oy}\boldsymbol{y}_{t-1} + \boldsymbol{b}_o\right) \tag{5}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tanh\left(\boldsymbol{W}_{cx}\boldsymbol{x}_t + \boldsymbol{W}_{cy}\boldsymbol{y}_{t-1} + \boldsymbol{b}_c\right) \tag{6}$$

$$\boldsymbol{y}_t = \boldsymbol{o}_t \odot \boldsymbol{c}_t \tag{7}$$

とする．但し，$\boldsymbol{i}$，$\boldsymbol{o}$，$\boldsymbol{f}$ はそれぞれ入力，出力，忘却ゲートで，$\boldsymbol{c}$ はメモリセルの内容である．$\boldsymbol{W}$ はそれぞれの層の重み行列，$\boldsymbol{b}$ はそれぞれの層のバイアスベクトルである．また，σ はシグモイド関数で，演算子 $\odot$ は要素ごとの積である．入力ゲートは時刻 t の状態の候補から反映する部分を選択し，忘却ゲートは時刻 $t-1$ の状態から反映する部分を選択する．さらに出力ゲートは時刻 t のメモリセルから出力に反映する部分を選択する．

2.2 画像キャプションの自動生成を行うニューラルネットワーク

画像に対してキャプションを自動的に生成する試みは，画像に対する情報検索の利便化などを目的に発展し，近年ではニューラルネットワークを用いた手法が数多く提唱されている．これらの多くは畳み込みニューラルネットワーク（CNN）で画像の特徴を抽出し，それを RNN（LSTM ベースのもの含む）などに入力してキャプションとなる文を出力する．このような手法としては，[22]，[8]，[23] などが挙げられるが，ここでは，本研究でその応用を行う [20] について説明する．

Tsutsui らが多言語で単一のモデルを共有する手法のために使用したモデル [20]（図 4）は，画像の特徴抽出のための CNN として ResNet-50[5] を用いている．画像から抽出した特徴を前節で説明した LSTM ベースの RNN に入力し，さらにその画像のキャプションを入力して，そのキャプションの生成確率を最大化するように RNN を学習する．LSTM ベースの RNN は，画像と全ての先行する単語を考慮して次の単語を予測するように訓練される．

予測モデルを時間的に展開すると，時刻 t において，画像 I と正しいキャプションの one-hot ベクトル表現の系列である $\boldsymbol{S} = \boldsymbol{S}_0, \boldsymbol{S}_1, ..., \boldsymbol{S}_N$ から，次の単語の確率分布 p_{t+1} は

$$\boldsymbol{x}_{-1} = \mathrm{CNN}\left(I\right) \tag{8}$$

$$\boldsymbol{x}_t = \boldsymbol{W}_{embed}\boldsymbol{S}_t,\ t \in \{n \in \mathbb{Z} | 0 \leq n \leq N-1\} \tag{9}$$

$$\boldsymbol{y}_t = \mathrm{LSTM}\left(\boldsymbol{x}_t\right),\ t \in \{n \in \mathbb{Z} | 0 \leq n \leq N-1\} \tag{10}$$

$$p_{t+1} = \mathrm{Softmax}\left(\boldsymbol{y}_t\right) \tag{11}$$

となる．但し，S_0，S_N はそれぞれ文頭と文末を表す仮想単語であり，$\boldsymbol{W}_{embed}$ は単語埋め込みのための行列である．また，Softmax はソフトマックス関数である．

各時刻の確率分布からの文の生成は，ビームサーチを使用して行われる．時刻 t までの系列の候補のうち，最良の k 個について次の時刻の確率分布 $p_t + 1$ を考慮し，長さ $t+1$ の系列を生成することを繰り返す方法である．

2.3 自動評価尺度

画像キャプション自動生成モデルの自動評価尺度については，多くの場合，その出力の類似性から，機械翻訳システムの機械翻訳文の正しさを定量的に把握するために考案された尺度を活用している．ここではその中から BLEU とその発展である BLEUS について説明する．

BLEU[14] は元来機械翻訳文の正しさを定量的に把握するために考案された，用意された正解翻訳文と機械翻訳文とを比較する手法で，その基本は正解翻訳文と機械翻訳文との n-gram の一致率が高ければ良い機械翻訳文を出力しているという思想である．画像キャプションの自動生成の分野では，正解翻訳文を正解キャプションに，機械翻訳文を自動生成キャプションに置き換えて比較し，自動生成の精度の評価によく用いられている．以下，N-gram までを用いる BLEU を BLEU(N) として，その式を示す．

$$\mathrm{BLEU}\left(N\right) = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} \frac{1}{N} \log p_n\right) \tag{12}$$

$$\mathrm{BP} = \min\left\{1, \exp\left(1 - \frac{\sum_k {}^{\text{正解翻訳文 } k \text{ の中の機械翻訳文 } k \text{ に単語数が一番近い文の単語数}}}{\sum_k {}^{\text{機械翻訳文 } k \text{ の単語数}}}\right)\right\} \tag{13}$$

$$p_n = \frac{\sum_{k=1}^{K} m_n(k)}{\sum_{k=1}^{K} c_n(k)} \tag{14}$$

$$c_n(k) = \text{機械翻訳文 } k \text{ の } n\text{-gram 数} \tag{15}$$

$$m_n(k) = \sum_x \min\left\{{}^{n\text{-gram}x \text{ が正解翻訳文 } k \text{ に現れる回数}}, {}^{n\text{-gram}x \text{ が機械翻訳文 } k \text{ に現れる回数}}\right\} \tag{16}$$

但し，x は機械翻訳文中に出現する n-gram を走るものとする．BP は Brevity Penalty で，システムが出力した文が正解文よりも短い場合に課されるペナルティである．BLEU はコーパス全体に対して計算するが，本研究で扱うような文ごとの評価では，n-gram の n が大きくなると

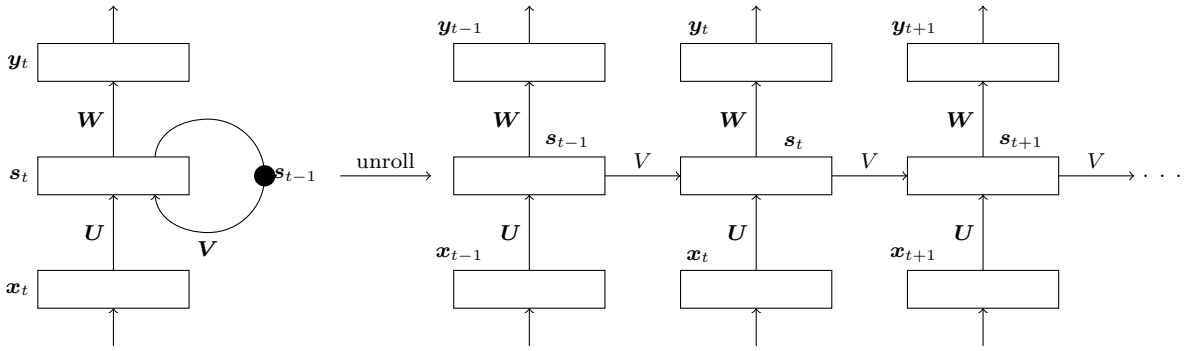

図 2 基本的な RNN の構造とその時間的展開

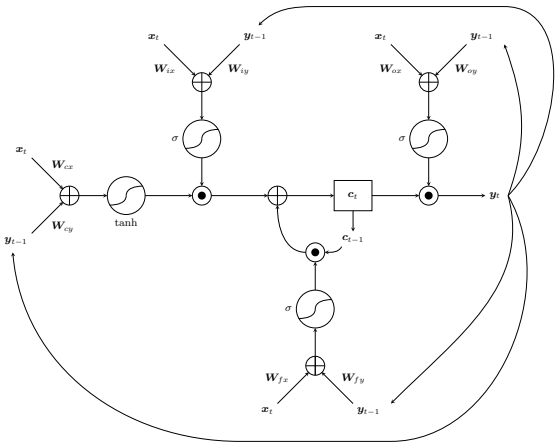

図 3 忘却ゲート付き LSTM

n-gram が 1 つも一致せず，p_n が 0 になってしまうからである．

そこで，これに平滑化を施した BLEUS[10] も良く用いられている．上記の（14）式に換えて次式の p_n を用いることで，n の大なるときも適合率が 0 となることを防ぐ．

$$p_n = \begin{cases} \frac{\sum_{k=1}^{K} m_n(k)}{\sum_{k=1}^{K} c_n(k)} & \text{if } n = 1 \\ \frac{\sum_{k=1}^{K} m_n(k)+1}{\sum_{k=1}^{K} c_n(k)+1} & \text{otherwise} \end{cases} \tag{17}$$

BLEU は 0 から 1 の値をとり，値が大きいほど正解翻訳文と機械翻訳文が近く，システムの評価が良いということになる．

2.4 日本語における展開

画像に対してキャプションを自動的に生成する試みでは，多種多様な画像を認識し, それらに対して適切にキャプションを付与する必要がある．したがって，画像とキャプション文のペアを大量に学習する必要があり，これには，MS-COCO[11] などの英語のデータセットが用意されてきた．しかし英語以外の言語をキャプションとしたデータセットは少なく，機械翻訳により英語キャプションを翻訳する方法や，[13] のような工夫が行われてきた．

近年では，Yoshikawa らによって MS-COCO に対して独自に日本語のキャプションを付した STAIR Captions[24] が整備され，既存手法 [8] を適応するだけで日本語キャプションを生成できることが確認されており，本研究でも活用を予定している．

2.5 RNN による単語の予測と補完

Spithourakis らが，ニューラルネットワークを用いた言語モデルによる単語の予測と補完を行っている [17]．これは，臨床領域において，患者の状態を報告する文章を作成することを想定した単語の予測と補完とを実験したものである．その補完に関するアルゴリズムを以下アルゴリズム 1 に示す．但し，$\mathcal{V}$ はシステムが持つ語彙で，*scorer* は評価関数である．Spithourakis らは，[16] で提案された文中に現れた数値情報に対して敏感なモデルの有効性を確かめるために，[12] をベースラインとしてこの実験を行い，高度に訓練されたモデルは単語の予測と補完とにおいても実用可能性があることを示した．

以上のように，言語モデルとしての RNN による単語の予測と補完は研究されてきた．しかしこの分野において，シチュエーションからもたらされる即時的な情報は，その活用が充分に検討されてきたとは言い難い．そこで本研究では，CNN+RNN のアーキテクチャを用いて，画像を利用する単語予測・補完を行うシステムを提案する．このマルチモーダルなモデルに対する応用可能性の調査は，我々の知る限り行われてこなかった．

3. 画像を利用する単語の予測と補完の提案

3.1 事前調査

本研究では，提案システムの評価を行うためにアンケートを取得し，画像 10 枚に対し，1 枚につき 7 つの文を回答として得た（詳細は 4.1 節にて後述する）．まず事前調査として，画像から自動生成されたキャプションと人間による表現との差異を明らかにする．

本研究で使用するモデル（詳細は 4.1.1 節にて後述する）が「人間が入力したい文」をどれだけ捉えられているかの参考にするために，本研究で使用するモデルから自動生成

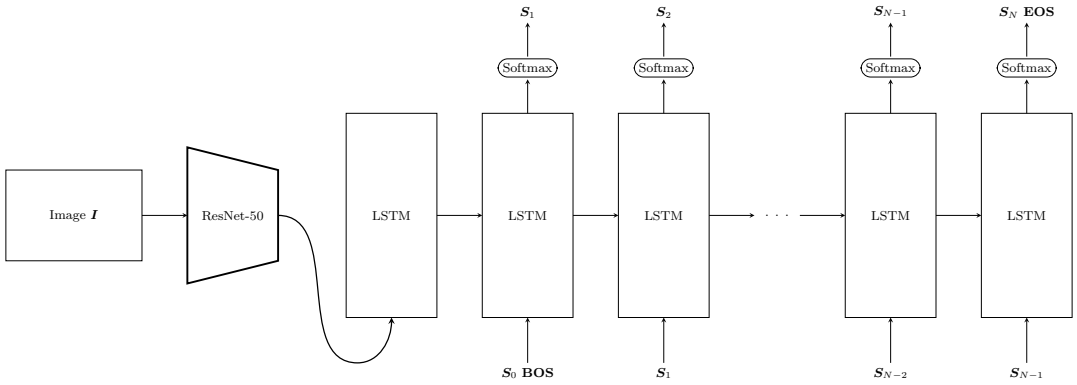

図 4 画像キャプション自動生成モデルの基本構造

Algorithm 1 Word Completion

Require: $\mathcal{V}$, *scorer*
Ensure: w_{next}
  function *complete_word*($\mathcal{V}$, *scorer*)
    $inputted\_char := Null$
    $lexicon := \mathcal{V}$
    loop
      $lexicon :=$
       $\{token \in lexicon | token$ is starting with $inputted\_char\}$
      $best\_word := \arg\max_{token \in lexicon}$ *scorer* $(token)$
      Display($best\_word$)
      $next\_char :=$ char readed by input
      if $next\_char == TAB$ **then**
        return $best\_word$
      else if $next\_char == WHITESPACE$ **then**
        return $inputted\_char$
      else
        $inputted\_char := inputted\_char + next\_char$
      end if
    end loop
  end function

されるキャプションと，アンケート回答によるテキスト（以下，入力文）との間で 2.3 で述べた BLEUS を計算した．ここでは自動生成文として，ビームサイズ 5 のビームサーチを用いた自動生成文を用いる．入力文を正解キャプションとして，モデルをキャプション自動生成システムとして用いた場合の自動生成されたキャプションとの BLEUS の値を表 1 に示す．

BLEUS(1)	BLEUS(2)	BLEUS(3)	BLEUS(4)
0.3656	0.2159	0.1776	0.1670

表 1 入力文と自動生成文との BLEUS スコア

使用モデルの BLEU スコアはモデルの公開ページに示されている (BLEU(1): 0.657, BLEU(2): 0.471, BLEU(3): 0.327, BLEU(4): 0.228) が，BLEUS スコアは BLEU スコアより高く出るにも関わらず，計算結果は使用モデルの BLEU スコアと比較しても大きく低い．画像キャプション自動生成システムが生成することを目指すキャプションは，画像と言語を併用した人間による表現とは異なることがわかった．

3.2 提案手法の概要

提案システムを用いる使用者は，画像と密接に関連したテキストの入力を補助される．対象の画像が CNN+RNN のシステムに認識されるとまず単語の予測候補が表示され，次いで使用者が入力を意図する単語の最初の文字を入力すると，補完候補が表示される．使用者はこれを繰り返し，素早く意図した表現によるテキストを完成させることができる．

本研究の基本設計では，計算量や環境に対する汎用性からインタラクティブ入力としてキーストロークやウェアラブル端末へのジェスチャ入力などのキャラクターレベルでの入力を仮定するが，提案手法は，音声入力など他の入力手段を補助する手法として，その本質を共有できることが期待される．

3.3 モデル側の設計

提案手法のイメージ図を図 5 に示す．モデルは予め学習されており，使用者はまず対象とする画像をシステムに入力することから始める．該当の画像は CNN への入力とされる．取り出された特徴量が RNN の最初の入力となり，以下，時刻 $t = 1$ からは使用者が決定した単語が式 9 での時刻 t への入力 $\boldsymbol{S}_t$ となる．

表示される候補の決定には，次単語の確率分布 p_{t+1} を用いる．確率の上位 k 個が予測候補として表示され，1 文字が入力されて語彙内を単語として入力中の文字列が変化するたびに前方一致検索して語彙の部分集合を作成し，その中から確率の上位 k 個を補完候補として表示する．したがって時刻 t における基本的なアルゴリズムはアルゴリズム 2 のようになる（出力された w_{next} が S_{t+1} として使用される）．

ただし，提案手法では，インタラクティビティのために，3.4 節で述べるように Backspace キーを用いることで DEL 文字が入力されることが想定されている．入力中のテキス

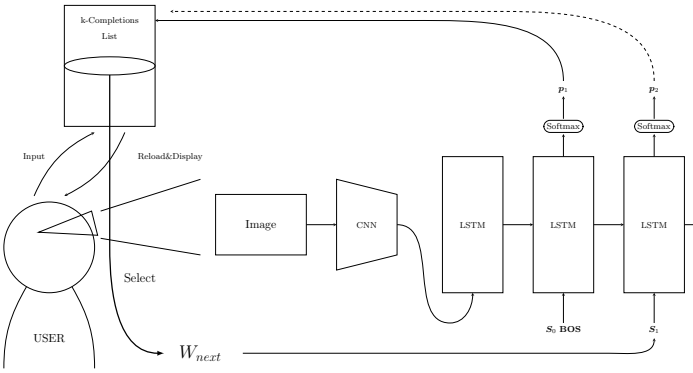

図 5 提案システムのイメージ

トの末尾の文字が単語間をまたがって削除されていくと，モデルを 1 単語（またはそれ以上）前の状態にロールバックし，その後の入力の変化に対応せねばならないという，画像キャプションの自動生成では通常あまり意識されない問題がある．

3.4 テキスト入力の設計

提案する入力支援システムは，本研究では英語に関して実験を行うため，米国圏で通常用いられる 101 キーボードでその機能を充分に果たすことを想定し以下のように動作する．

- アルファベットを入力された場合
  - 候補単語を選択していなければ，現在入力中の文字列の直後にその文字が入力される．
  - 候補単語を選択していれば，その単語の直後に接尾してその文字が入力される．
- Space キーまたは Enter キーを入力された場合
  - 候補単語を選択していなければ，現在入力中の文字列が次単語として言語モデルに入力され，半角空白が現在入力中の文字列の直後に入力される．
  - 候補単語を選択していれば，現在入力中の文字列の最後の空白の次の文字から先がその単語に置換され，その単語が次単語として言語モデルに入力され，半角空白が入力される．
- Backspace キーを入力された場合
  - 候補単語を選択していなければ，現在入力中の文字列の最後の 1 文字を削除する．
  - 候補単語を選択していれば，現在入力中の文字列の最後の空白の次の文字から先がその単語の最後の 1 文字を削除した文字列に置換される．
- tab キーを入力された場合
  - 候補単語を選択していなければ，候補単語第 1 位が選択される．
  - 候補単語を選択していれば，次の候補単語に選択が移る．

Algorithm 2 Proposal system

Require: $\mathcal{V}$, $\boldsymbol{S}_0, ..., \boldsymbol{S}_t$
Ensure: w_{next}
  function *complete_word*($\mathcal{V}$, S_t)
    $inputted\_char := Null$
    $lexicon := \mathcal{V}$
    $\boldsymbol{x}_t := \boldsymbol{W}_{embed}\boldsymbol{S}_t$
    $\boldsymbol{y}_t := \mathrm{LSTM}(\boldsymbol{x}_t)$
    $p_{t+1} := \mathrm{Softmax}(\boldsymbol{y}_t)$
    loop
      $lexicon :=$
       $\{token \in lexicon | token$ is starting with $inputted\_char\}$
      $best\_kwords := \{token \in lexicon | token$ whose probability is in the top k of p_{t+1} $\}$
      Display($best\_kwords$)
      $next\_input :=$ char readed by input or selected word
      if $next\_input == WORD$ **then**
        return $next\_input$
      else if $next\_char == WHITESPACE$ **then**
        return $inputted\_char$
      else
        $inputted\_char := inputted\_char + next\_input$
      end if
    end loop
  end function

3.5 実装

本研究では，CNN に学習済みの ResNet-50[5] を用い，RNN は LSTM ベースの RNN[12] を用いたモデルを使用する．

また，3.3 節の最後に述べた問題を解決するために，系列モデルの可遡性を確保する必要がある．そこで，単語が確定されると，そのときの LSTM の状態を保存する．1 単語以上遡って書き直す場合，LSTM を保存された状態にまでロールバックすることで，別単語が入力された場合もそこまでの入力系列の記憶を保ったまま再開し，その後の入力系列にも対応することができる．

語彙内に無い単語が入力された場合は，モデルに対しては未知語として入力を行う．

予測・補完候補は複数個表示することも試行する．本研究では PC において tab キーを用いて候補選択を行うことを想定するが，主にモバイル端末では，予測・補完候補を複数個の中から選択する際でもタッチ数は 1 回で済むため，その有効性をより良く発揮することを視野に入れるためである．

4. 提案システムの評価

4.1 評価方法と環境

本研究では，提案システムの評価を行うため，まず，flickr 8k[7] からランダムに 10 枚の画像を抽出した．被験者にこれらの画像 10 枚を提示し，「その画像をライフログとして記録しておくときに（自分ならば）付する文」を考えてもらった．10 名から回答を得て，一つの画像につき 10 個の

入力したい文（以下，入力文とする）が揃った．

この画像と入力文に対する言語モデルとしての評価のためにパープレキシティ及び iPP[25] の計算と，入力支援システムとしての評価のためにタイピング方法を想定しキーストロークセービングの計算を行った．

4.1.1 使用モデル

使用したモデルは Tsutsui ら [20] がすでに公開している[*1]英語のもので，詳細は以下である．

- 英語は空白で区切って分かち書きする．
- 実装には chainer[18] を用いている．
- LSTM の隠れ層は 512 次元である．
- 学習には MSCOCO[11] のうち YJ Captions 26k[13] に用いられている 26,500 枚を，22,500 枚を訓練用，2,000 枚を検証用, 2,000 枚を検証用に分割した．
- 学習にはデフォルトハイパーパラメータの Adam[9] を用いた．
- バッチサイズ 128 で 40 エポック訓練し，テストセットに対する CIDEr[21] スコアが最も良いモデルを選択した．

なお，本研究では，比較のために同条件で学習した LSTM ベースの RNN だけの言語モデルも用いる．図中では提案システムを CNN+RNN，LSTM ベースの RNN だけの言語モデルを用いたものを RNN と表記する．

4.1.2 パープレキシティと iPP

パープレキシティ（以下，PP とする）は言語モデルの評価に用いられる指標である．単語ごとのパープレキシティはモデル M とテストセット内のテキストの集合 D に対して以下のように定義される．

$$\mathrm{PP}\left(S|M\right) = 2^{\mathrm{H}(D|M)} \tag{18}$$

$$\mathrm{H}\left(D|M\right) = -\frac{1}{\sum_{s\in D} l_s}\sum_s\sum_{t=1}^{l_s}\log_2 P_M\left(w_t|w_1w_2...w_{t-1}\right) \tag{19}$$

但し，l_s はテキスト s の単語数で，それぞれのテキスト s は形態素 $w_1...w_{l_s}$ からなる．$P_M\left(w_t|w_1w_2...w_{t-1}\right)$ はモデル M の下で，単語列 $w_1w_2...w_{t-1}$ の次に w_t が生起する確率である．

PP は言語モデルでテストセット内のテキストを予測した際の単語の平均分岐数を表す指標であるが，テキスト入力において候補単語を表示する際には，使用者が 1 文字を入力するたびに候補が絞り込まれる．したがって本研究では，中村らによって提案された，次の単語を途中まで任意の文字数入力した状況における平均単語分岐数を表す iPP も併せて用いる．

入力文字数 l における iPP(l) はモデル M とテストセット内のテキストの集合 D に対して以下のように定義される．

$$\mathrm{iPP}\left(l\right) = 2^{\mathrm{H}(D|M)} \tag{20}$$

$$\mathrm{H}\left(D|M\right) = -\frac{1}{\sum_{s\in D} l_s}\sum_s\sum_{t=1}^{l_s}\log_2 P_M\left(w_t|w_1w_2...w_{t-1};w_t(:l)\right) \tag{21}$$

$$P_M\left(w_t|w_1w_2...w_{t-1};w_t(:l)\right) = \begin{cases}\frac{P_M(w_t|w_1w_2...w_{t-1})}{\sum_{w\in V(w_t(:l))}P_M(w|w_1w_2...w_{t-1})} & w\in V\left(w_t(:l)\right)\\ 0 & w\notin V\left(w_t(:l)\right)\end{cases} \tag{22}$$

但し $w_t(:l)$ は w_t の前から l 文字目までを表し，$V\left(w_t(:l)\right)$ はモデルの語彙内で最初の l 文字が $w_t(:l)$ に一致する語の集合である．

入力文字数 $l=0$ のときの iPP は PP と一致する．

4.1.3 キーストロークセービング

キーストロークセービングとは，レター・バイ・レターの打鍵回数に対して，システムの使用によって削減された打鍵回数の割合である．

Trnka ら [19] はキーストロークセービングについて，システム使用による値とともに理論上の限界と語彙上の限界を示すことを提案している．理論上の限界とは，システムが必ず使用者が次に入力したい単語を候補単語第 1 位として表示していると仮定した場合の打鍵回数を用いたキーストロークセービングの値である．語彙上の限界は，使用者が次に入力したい単語がシステムが用いている言語モデルの語彙に含まれている際は必ず候補単語第 1 位として表示しており，それ以外については全文字入力する必要があると仮定した場合の打鍵回数を用いたキーストロークセービングである．これらの限界を同時に示すことで，システム使用による値の実際の評価の一助とすることができる．本研究でもこれらの値を示す．

また，本研究ではキーストロークセービングを計算する際，使用者が以下のようなタイピングをすることを想定した．

- 表示されている候補単語の中に入力したい単語または入力したい単語の前方部分文字列がある場合，以下のうちから打鍵回数を少ない方を選択する．
  - その候補単語を選択状態にするための tab キーの打鍵回数
  - その候補単語を入力しきるのに必要な残りの文字入力回数
- 入力したい単語が候補単語の中の前方部分文字列となっている場合，以下のうちから打鍵回数を少ない方を選択する．
  - 候補単語を選択状態にするための tab キーと，その後の余分な文字列を削除するための Backspace キー

[*1] https://github.com/apple2373/chainer-caption 2018/2/5 閲覧

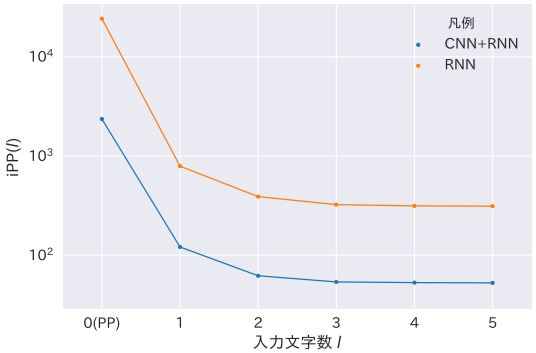

図 6　テストセット iPP

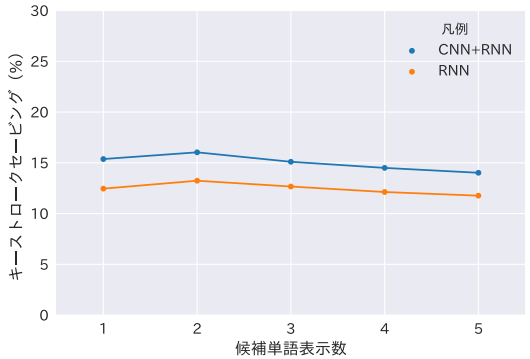

図 7　候補単語表示数を変化させたときのキーストロークセービング

とを加えた打鍵回数

– 入力したい単語を入力しきるのに必要な残りの文字入力回数

4.2　結果と評価

4.2.1　PP と iPP

単語ごとの入力文字数 l を 0〜5 で変化させ，テストセットに対して iPP(l) を計算した結果を図 6 に示す．$l = 0$ のとき，iPP(0) は PP に一致する．

PP の値は大きいが，前方文字列の入力により iPP が大きく減少し，特に 1 文字入力するだけで iPP が 3 桁（121.21）まで落ちており，2 文字目に 2 桁（62.17）となった．これは言語モデルの平均単語分岐数として良好な値である．また，提案システムは常に RNN だけのモデルを用いたシステムよりも平均単語分岐数が少なく，予測・補完が高精度である．

4.2.2　キーストロークセービング

4.1.3 節で述べた方法で，本研究で提案するシステムに対する理論上の限界，語彙上の限界及び，想定されたタイピングを行う使用者のキーストロークセービングを計算した．想定されたタイピングを行う使用者のキーストロークセービングについては，候補単語ウィンドウのサイズ（候補として表示する単語数）を 1〜5 で変化させて計測した．表 2 と図 7 にその結果を示す．

画像について人間の入力したい文を完成させるのに必要な打鍵回数を，画像を用いない方法よりも削減できる可能性があり、提案手法の効果を確認した．

4.3　議論

本研究は，画像情報と言語情報の組み合わせによる自由な自己表現のための，テキスト入力の高速化・省力化と柔軟性とが両立したテキスト入力支援システムを目的とする．4.2 節を振り返ると，文字入力による iPP の大幅な減少によって，提案手法が使用者の入力意思に対して柔軟性を持つことが確認され，キーストロークセービングスによって実際に打鍵回数の 16%以上を削減できる可能性があり，またそれぞれにおいて画像を用いない方法よりも良い結果を得たため，テキスト入力の高速化・省力化を目指し画像を入力とする提案手法の有効性が示された．

しかし以下のような点を考慮に入れることで，提案手法のより正確な評価を行うことが可能である．

第一に，モデルに言語モデルとして改良の余地がある．PP の値は悪く，キーストロークセービングの値は理論上の限界からも語彙上の限界からも遠い．画像キャプションの自動生成タスクのために一般的に学習されるようなデータセットではなく，より表現の幅が広いデータセットを用いれば，改善が期待される．

第二に，想定されたタイピングが単純であり，提案手法のキーストロークはさらに改善する余地がある．理想的な使用者のもとではキーストロークセービングは候補ウィンドウサイズに対して単調増加しなければならない．4.1.3 節で述べたような使用者は，自分の選択したい単語がウィンドウの比較的下位に現れた際に，追加で途中まで文字を入力して選択したい単語をウィンドウの上位に引き出すという動作を行わない．ウィンドウサイズが大きければ自らの選択したい単語が表示される可能性が高いため，結果的に打鍵回数が増えてしまうことになる．

5.　おわりに

本研究では，瞬間的な感情や思考の，自己表現としてのマルチメディアでの即時的な記録・発信の増加を背景に，マルチモーダルなモデルを活かした，使用者の入力意思に対しての柔軟性を確保しつつテキスト入力を高速化・省力化するテキスト入力支援について，画像情報を用いる例を提案し，その手法の評価を行った．提案手法はフレーズ単位の予測・補完ではなく単語単位の予測・補完を行うことで使用者の入力意思に対して柔軟性を確保することを目指し，系列モデルに可遡性を確保することで入力が削除される可能性に対応することで，単語単位の予測・補完というインタラクションを可能にした．結果として，打鍵回数の

想定されたタイピング（ウィンドウサイズ 2）（%）	理論上の限界（%）	語彙上の限界（%）
16.03	60.47	46.18

表 2 キーストロークセービング

削減の比較によって，提案手法が画像情報と関連するテキスト入力を高速化・省力化できることが示され，その有効性が確認された．

今後の方針として, 日本語対応及びシステム使用者に対するアンケート調査や文字入力速度による比較などを考えている．また，（声に限らない）環境音を情報とする入力支援，動画の内容を受け取ることでモバイル端末・ウェアラブル端末を用いての眼前の光景を背景とした即時的な文章編集のための入力支援など，画像以外のメディアについても拡大されることを期待する．

謝辞 本論文の研究は一部, 共同研究「VR・MR 環境のための文字入力インタフェースの研究」（TIS 株式会社）によります．

参考文献

[1] Kenneth C. Arnold, Krzysztof Z. Gajos, and Adam T. Kalai. On suggesting phrases vs. predicting words for mobile text composition. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pp. 603–608, New York, NY, USA, 2016. ACM.

[2] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, Vol. 5, No. 2, pp. 157–166, Mar 1994.

[3] Jeffrey L. Elman. Finding structure in time. *COGNITIVE SCIENCE*, Vol. 14, No. 2, pp. 179–211, 1990.

[4] Felix A. Gers, Jürgen A. Schmidhuber, and Fred A. Cummins. Learning to forget: Continual prediction with lstm. *Neural Comput.*, Vol. 12, No. 10, pp. 2451–2471, October 2000.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, June 2016.

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

[7] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, Vol. 47, No. 1, pp. 853–899, May 2013.

[8] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3128–3137, June 2015.

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, Vol. abs/1412.6980, , 2014.

[10] Chin-Yew Lin and Franz Josef Och. Orange: a method for evaluating automatic evaluation metrics for machine translation. August 2004.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. *Microsoft COCO: Common Objects in Context*, pp. 740–755. Springer International Publishing, Cham, 2014.

[12] Tomas Mikolov, Martin Karafiát, Lukás Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, pp. 1045–1048, 2010.

[13] Takashi Miyazaki and Nobuyuki Shimizu. Cross-lingual image caption generation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1780–1790, Berlin, Germany, August 2016. Association for Computational Linguistics.

[14] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[15] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. In James A. Anderson and Edward Rosenfeld, editors, *Neurocomputing: Foundations of Research*, pp. 696–699. MIT Press, Cambridge, MA, USA, 1988.

[16] Georgios P. Spithourakis, Isabelle Augenstein, and Sebastian Riedel. Numerically grounded language models for semantic error correction. *CoRR*, Vol. abs/1608.04147, , 2016.

[17] Georgios P. Spithourakis, Steffen E. Petersen, and Sebastian Riedel. Clinical text prediction with numerically grounded conditional language models. *CoRR*, Vol. abs/1610.06370, , 2016.

[18] Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton. Chainer: a next-generation open source framework for deep learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.

[19] Keith Trnka and Kathleen McCoy. Evaluating word prediction: Framing keystroke savings. In *Proceedings of ACL-08: HLT, Short Papers*, pp. 261–264, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[20] Satoshi Tsutsui and David Crandall. Using Artificial Tokens to Control Languages for Multilingual Image Caption Generation. In *CVPR Language and Vision Workshop*, 2017.

[21] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4566–4575, June 2015.

[22] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3156–3164, June 2015.

[23] Q. Wu, C. Shen, L. Liu, A. Dick, and A. v. d. Hengel. What value do explicit high level concepts have in vision to language problems? In *2016 IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), pp. 203–212, June 2016.

[24] Yuya Yoshikawa, Yutaro Shigeto, and Akikazu Takeuchi. Stair captions: Constructing a large-scale japanese image caption dataset. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 417–421, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[25] 中村明, 速水悟, 津田裕亮, 松本忠博, 池田尚志. 複数モデルの統合による lda トピックモデルの高精度化とテキスト入力支援への応用. 情報処理学会論文誌, Vol. 50, No. 4, pp. 1375–1389, apr 2009.