一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
NLC2021-39(2022-03)

生成的自然言語推論のためのデータ収集とモデル構築

榎本　　悠†　嶋田　和孝†

† 九州工業大学大学院 情報工学府 〒820–8502 福岡県飯塚市川津 680–4
E-mail: †enomoto.haruka137@mail.kyutech.jp, shimada@ai.kyutech.ac.jp

あらまし　我々は日常生活の中で常識を用いた推論をよく行っている．例えば，「目が覚める」というイベントの後に起きるイベントを考えたとき，「顔を洗う」や「朝ご飯を食べる」といったものが一般的に考えられる．このような自然言語推論は自然言語処理分野における主要なタスクの 1 つであり，盛んに研究が行われている．自然言語推論はそのほとんどが分類タスクとして解かれている．しかし，分類モデルは与えられた選択肢以外の回答を出力することができない．そのため，推論モデルは真の意味で推論を解いているといえない．これに対して，より柔軟に，真の意味で推論を解くことのできるモデルが生成モデルである．生成モデルは分類モデルと比べて，モデルの出力数を調整することで大量の推論結果を得ることが可能である．また，任意の入力に対して，生成モデルは対応するイベントを生成することができる．そこで本論文では，生成モデルを用いた自然言語推論を行う．しかし，生成モデルを用いた推論の研究は少なく，データも不十分である．そのため，Twitter データを用いて前後関係を持った大量のイベント文のペアを収集する．その後，収集したデータを用いて生成モデルである Seq2Seq や T5 を学習させ，生成実験を行う．しかし，出力結果が正しいものであるかを人手で判断することは難しい．そこで，編集距離を用いた評価方法の提案を行い，その結果について考察する．

キーワード　自然言語推論，常識的推論，文生成

Generating the next action from sequence event pairs automatically acquired from Twitter

Haruka ENOMOTO† and Kazutaka SHIMADA†

† Kyushu Institute of Technology 680–4 Kawadu, Iizuka-shi, Fukuoka, 820–8502 Japan
E-mail: †enomoto.haruka137@mail.kyutech.jp, shimada@ai.kyutech.ac.jp

Abstract　Recently, natural language inference have been actively studied and it's mainly solved as a classification task. However, classification models can only select answers that are included in the given choices. Therefore, it doesn't truly solve the inference task. In this paper, we propose a method to generate the next actions from a previous action. Generally, we need a large number of training data to learn a generation model. However, there are no data for the generation task of commonsense reasoning. Therefore, we automatically collect event-sentence pairs from Twitter. and we conduct a generation experiment. We apply the extracted data to neural network-based models; seq2seq with attention and T5. One problem of this task is to evaluate the applicability of the generated sentences. In this paper, we also propose an evaluation measure based on Levenshtein distance. Through the measure, we compare the outputs from two generation methods.

Key words　Natural Language Inference, Commonsense Reasoning, Text Generation

1. ま え が き

我々は日々の日常生活の中で常識を用いた推論をよく行っている．例えば，以下のような質問について考えてみる．

- 「目が覚める」というイベントの後に起きるイベントは何？

上記のような問題に対する回答として，「顔を洗う」や「朝ご飯を食べる」といったものが一般的に考えられる．このような自然言語推論は自然言語処理分野における主要なタスクの 1 つである．自然言語推論は様々な分野に応用することができる．汎用的な推論モデルが実現すれば，対話システム [1] や情報抽出 [2] といったタスクに役立つと考えられている．そのため，近年では自然言語推論の研究が盛んに行われている．SNLI [3] や MNLI [4] などの含意関係認識といった簡単なタスクを始め

として，SQuAD [5] といった質問応答データセット，SWAG [6] や HellaSwag [7] といった困難性の高い問題設定を有したデータセットなど様々なタスクが提案されている．それに伴いモデルの性能も向上している．ニューラルネットワークを用いたモデルが現在の主流となっているが，その中でも BERT [8] を始めとする事前学習済みモデルは様々なタスクで高精度を記録している．

しかしながら，これらはすべて分類タスクとして取り組まれている．分類タスクでの自然言語推論はモデルにいくつかの選択肢を与え，入力として与えられた情報を元に選択肢の中から適切な回答を探すというものである．そのため，分類モデルは選択肢中の適切な回答の数までしか出力することができない．また，与えられた選択肢以外の回答を出力することができない．このことから，分類モデルでは真の意味で推論問題を解くことはできないといえる．

これに対して，真の意味で推論を解くことのできるモデルが生成モデルである．ここでの生成モデルとは，何らかのデータセットを与えることで，そのデータセットに対して学習を行い，それらのデータと似たような新たなデータを生成するモデルのことである．例えば，（目が覚める，顔を洗う）のような前後関係のあるイベントペアを用意し，学習することで，任意の入力（例えば，雨が降る）に対して，生成モデルは対応するイベント（例えば，傘をさす）を生成することができる．加えて，生成モデルの出力数を調整することで，大量の推論結果を得ることも可能である．

そこで，本論文では「出来事を示した文（イベント文）を与えることで，その次に起きるであろうイベントを推論する」という問題設定のもと，生成モデルを用いた自然言語推論を行う．具体的には，「目が覚める」というイベント文を与えることで，「顔を洗う」や「朝ご飯を食べる」といった複数のイベント文を生成してくれることを期待する．生成モデルを学習させるためには大量のデータが必要となる．しかし，生成モデルを用いた自然言語推論の研究は数が少なく，十分なデータが存在するとはいえない [9], [10]．そのため，（目が覚める，顔を洗う）のような大量の前後関係を持ったイベント文のペアデータを収集する．この論文では Twitter をその収集対象とする．その後，収集したデータを用いてモデルを学習させ，構築したモデルに対して生成実験を行う．しかし，入力として与えられた文のみから出力結果が適切なイベントを記した文となっているかを判断することは人手でも難しい．そこで，入出力のペアから 1 つの文を人手で作成し，編集距離を用いて評価を行う方法を提案する．また，実際に評価を行うことでその結果について考察をする．

2. データセット

本節ではデータセットの構築について述べる．対象となるのは Twitter データである．Twitter データから自動収集した各 Tweet に対して，いくつかの手がかり表現をもとに，前後関係を持ったイベントを抽出する（これをイベントペアと呼ぶ）．はじめに，2.1 節で Twitter データからイベントペアデータセットを作成する手法をまとめ，2.2 節で実際に作成したイベントペアの一部を例として示す．

2.1 データセット作成手法

Twitter データを用いて前後関係を持ったイベントペアデータセットを作成する．Twitter データは山元ら [11] が TwitterAPI を利用して収集したものを用いる．64.5 万ユーザの計 4.1 億 Tweet を対象に，イベントペアデータセットを作成する．

イベントペアを作成するためには，前後関係を持った 2 つのイベントを見つける必要がある．例えば，「入力作業してたら肩より腰にきた」という文章について考える．なお，この文章は使用する Twitter データに実際に含まれている Tweet 本文である．この文章から「入力作業をする」と「腰にくる」という前後関係を持った 2 つのイベントを見つけることができる．これは，2 つのイベント間を繋ぐ「たら」という助動詞の働きによるものである．そこで，このように前後関係を示す助動詞や接続助詞といったフレーズを設定し，そのフレーズをもとに Tweet 本文を 2 文に分割することで前後関係を持った 2 つの候補を見つける．設定したフレーズの種類は以下である．

たら，だら，て，で，と，ば，後

分割したそれぞれの文からイベント文を獲得し，前後関係を持ったイベントペアを作成する．イベント文の獲得には係り受け情報を用いる．使用した係り受け解析器は CaboCha [12] である．手順を以下に記す．また，その処理の例を図 1 に示す．

（1） Tweet 本文に対して，設定したフレーズが含まれているかを探す．

（2） フレーズが含まれている場合，そのフレーズをもとに文章を分割する．

（3） 分割した各文から動詞を取得する（原形に変換）．

（4） 係り受け情報をもとに，取得した動詞を含むチャンクとリンクしているチャンク（内容語）を取得する．

（5） 取得したチャンクとリンクしているチャンク（修飾語）を取得する．

（6） 取得できるチャンクがなくなるまで（5）を繰り返す．

2.2 ペアデータセット作成結果

2.1 節において獲得したイベントペアデータセットは合計で 347,783 ペアである．作成したイベントペアデータセットの一部を表 1 に示す．作成したイベントペアデータセットの中から無作為に選んだものを載せている．表 1 より，一部において非文を含んではいるものの，（薬飲む，しっかり寝る）や（今やってる原稿終わる，ゲーム始める）など事象の前後関係を持ったイベントのペアをしっかりと取得できていることが分かる．

また，「前半戦振り返ってる」や「太鼓の達人久しぶりにやる」などのように助詞の欠けがいくつか見られる．SNS という手軽に文章を投稿できる性質であるためか，得られた表現は話し言葉に近い傾向にあると考えられる．

3. モデルの構築

2. 節において作成したイベントペアデータセットを用い

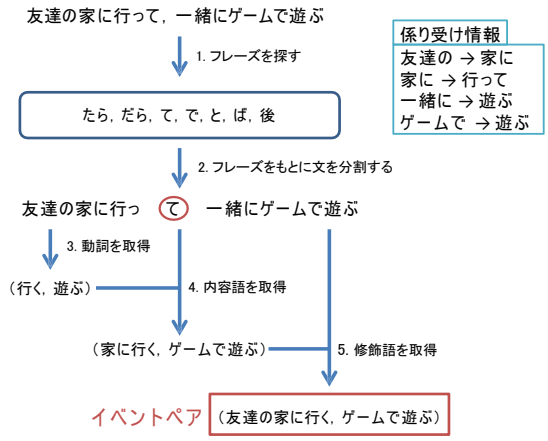

図 1　ペアデータ獲得手順の概要図

Fig. 1　Schematic diagram of paired data acquisition

表 1　収集したイベントペアデータの例

Table 1　Event pair data example

（イベント文（前），イベント文（後））
（ネ燃やす，11 時前には寝る）
（パクソノ本気でデビューする，変な笑い出る）
（薬飲む，しっかり寝る）
（前半戦振り返ってる，２０分くらいから寝落ちする）
（太鼓の達人久しぶりにやる，ロキとかある）
（今やってる原稿終わる，ゲーム始める）

て，生成モデルを構築する．構築した生成モデルはニューラルネットワークモデルである Sequence-To-Sequence [13] と事前学習済みモデルである T5 [14] の 2 種類である．3.1 節で Sequence-To-Sequence の説明を行い，3.2 節では T5 について説明する．

3.1　Sequence-To-Sequence

Sequence-To-Sequence モデルはその名の通り，あるシークエンスを受け取ることで別のシークエンスへと変換するモデルである．例えば文章を与えるとモデル内部のエンコーダによって文章ベクトルへと変換される．変換された文章ベクトルはデコーダへと与えられ，デコーダは与えられたベクトルをもとに別の文章を生成する．そのため，文全体を考慮した出力を行うことができる．しかし，デコーダへと与える情報はエンコーダの各隠れ層の和であるため，何も工夫をせずに渡してしまうと分散表現を与えるだけになってしまい，参照すべき情報が正しく伝わらないという問題点を抱えている．この問題の解決策として，Luong ら [15] は Attention 機構を提案している．

Attention 機構とは，与えられた情報のうちどの値を使用するべきかを学習するためのものである．そのため，必要な情報に重みを付けて学習することができ，性能が向上すると報告されている．本論文では，Sequence-To-Sequence に Attention 機構を組み合わせた Sequence-To-Sequence + Attention（以下，Seq2Seq）モデルを実装する．

3.2　Text-To-Text Transfer Transformer (T5)

T5 は Text-To-Text Transfer Transformer という名の通り，Text-To-Text と Transer，そして Transformer という特徴を兼ね備えたモデルである．Text-To-Text とは入力として文章を与えることで，そのまま文章が出力されるモデルのことを指す．また，Transfer とは転移学習（Transfer Learning）のことを指している．T5 は大規模データによる事前学習済みモデルであるため，転移学習を用いることで様々なタスクに対して高い性能を持つという特徴を持っている．最後の Transformer とは Vaswani ら [16] が提案した Self-Attention と呼ばれる Attention 機構の発展形のみを用いた機械学習モデルのことであり，回帰型ニューラルネットワークや畳み込みニューラルネットワークを一切使わないモデルでもある．T5 はこの Transformer の技術をベースに構築されたモデルである．

T5 には様々なタスクの問題をそのまま入力として与えることができ，T5 の出力も入力として与えた問題に対する答えがそのまま返る．そのため，翻訳や質疑応答，要約といったあらゆる自然言語処理タスクを 1 つのモデルで解くことができる．

本論文では，日鉄ソリューションズ株式会社が提供している大規模な日本語コーパスで事前学習された日本語用 T5 [1] を生成モデルとして利用した．

4.　生 成 実 験

本節では，構築した生成モデルを用いて生成実験を行う．4.1 節で実験設定について説明し，その実験結果と考察について 4.2 節で述べる．

4.1　実 験 設 定

本節では実験設定について詳しく説明する．実験を行うにあたり，結果の出力用として各モデルに入力として与えるための文（以下，入力文とする）を，筆者が人手で作成した．モデルにはイベントペアデータセットを学習データとして渡し，学習したモデルに対して各入力文を与えることでその答えとなる文を生成させる．モデルが生成する文の数は各入力文につき 5 文とする．また，T5 はモデルの学習時に開発データを必要とするため，イベントペアデータセットを 9 : 1（転移学習データ : 開発データ）に分割する．

性能実験を行うにあたり Seq2Seq と T5 に設定したパラメータを以下に示す．

Seq2Seq

- Batch Size : 32
- Epoch 数 : 40
- 学習率 : 5e-3
- 最適化関数 : Adam

T5

- Batch Size : 16
- Epoch 数 : 8
- 学習率 : 3e-4
- 最適化関数 : AdamW

（注1）：https://github.com/sonoisa/t5-japanese

4.2 実験結果と考察

各入力文に対して生成モデルが出力した結果の一部を表 2 に示す．

表 2 より，Seq2Seq と T5 の結果を比べると，Seq2Seq に対して，T5 は例えば，「家を出る」という入力に対して「雨が降ってくる」や「雨降ってくる」など似た表現を出力していることが見てとれる．このことから，T5 に比べて Seq2Seq の方が出力の多様性は高いといえる．

一方で，T5 の出力はそれ一文で完結したイベント文となっているのに対し，Seq2Seq の出力は「を忘れる」のように目的語の抜けた非文になっていたり，「腐る」などのように主語のない，曖昧なイベント（具体性を欠いたイベント）文が生成されていることが分かる．このことから，Seq2Seq に比べて T5 の方が曖昧性のない文法的に正しいイベント文を出力する傾向があることがわかる．

また，「友達と会う」という入力文に対して Seq2Seq では「枯渇なる」，T5 では「お菓子もらう」などのように，両モデルの出力文に助詞の欠けが見られる．これは，学習に用いたデータセットの影響であると考えられ，2.2 節で述べたように，助詞の欠けが見られるデータがイベントペアデータセット内に多く存在するためだと考えられる．

各モデルの出力文が入力文に対して適切なイベント文かどうかについても見ていく．「友達と会う」という入力文に対して Seq2Seq の「懐かしくなる」や T5 の「全力で抱きしめる」など，適切そうなイベント文が出力できていることが確認できる．一方で，「大会で優勝する」というイベント文に対して Seq2Seq の「たらす」など，何を指しているのか分からず，一見して不適切そうなイベント文が生成されていることも確認できる．しかし，見る人によっては「大会で優勝して金メダルを首からたらす」といったように，上手く情報を補完することで適切だと判断する場合も考えられる．このように，単純に入力と出力を見比べただけでは判断する人によって評価が変わるため，人手によって出力結果が入力に対するイベント文として適切かどうかを判定することは難しいといえる．

5. 評 価 実 験

4. 節では，作成したイベントペアデータセットを用いて実装した生成モデルに対して，その生成実験を行った．4.2 節で述べたように，モデルの出力結果について，入力文に対する適切なイベント文であるか 2 つの文を見比べるだけでは判断が難しいものも多い．見比べるだけで適切と判断できる入出力のペアについて考えてみると，例えば，4.2 節で挙げた（友達と会う，懐かしくなる）というペアにおいては，「友達と会うと懐かしくなる」のように，簡単な接続詞（あるいは，助詞・助動詞）を挿入するだけで文章が完成する．不適切な入出力ペアである場合は，付与だけではなく，書き換えが必要になる場合も考えられる．このことから，作成した文章を用いて編集距離を測定することで，出力結果の適切度合いを評価できると仮定する．

そこで，編集距離を用いた入力文に対するモデル出力文の評価方法を検討し，実際に評価を行う．

表 2　各生成モデルの出力結果
Table 2　Output results for each generative model

入力文	Seq2Seq	T5
家を出る	四国を見てしまう いい加減を眺める 何気なくを待つ を忘れる 物干してなる	雨が降ってくる 雨降ってくる 5 分で着く 車に積む 電車に乗り遅れる
友達と会う	枯渇なる 懐かしくなる 下品垂れてくる 腐る 年下になる	めっちゃ笑われる 全力で抱きしめる お菓子もらう なんか泣けてくる 色々話してる
大会で優勝する	強打する 延々とにする 流石に下りる たらす アビリティになる	お菓子貰える 優勝できる自信ある 今度こそは優勝する 賞金あげる お菓子もらう

5.1 合成文作成

評価を行うため，生成実験における入出力ペアを用いて人手による合成文の作成を行う．文の作成者（以下，ワーカー）は筆者と同じ研究室の学生 6 名である．文を作成するため，ワーカーには入力文とそれに対するモデルの出力文のペアを与える．文を作成するにあたり，ワーカーにはいくつかの手順を伝えている．手順に沿ってワーカーは文作成を行い，その手順で文が作成できなければ，次の手順を参照する．また，文を作成するにあたり，句読点は自由に使ってもよいことにしている．その後，作成した文に対して文章判定を行う．具体的には，許容できる書き換えであるかの判定を行い，許容できる書き換えであれば Accept，許容できない書き換えであれば UnAccept と付与する．合成文作成の詳しい手順を以下に示す．

（手順 1 ） 文同士が自然につながるように，各文の末尾のみ変化させてもよい（「入力文」と「出力文」を全て使用）．

（手順 2 ） 手順 1 に加えて，出力文側に単語を挿入してもよい（「入力文」と「出力文」を全て使用，単語の削除・書き換えは禁止）．

（手順 3 ） 手順 1，2 に加えて，出力文側の名詞や助詞を書き換えてもよい（「入力文」と「出力文の動詞」を使用）．

（手順 4 ） 手順 1，2 に加えて，出力文側の動詞に対して書き換え・削除してもよい（「入力文」と「出力文の目的語部分」を使用）．

（手順 5 ） 「入力文」のみを使用して文を作成する．つまり，出力文を無視してよい．

合成文作成の対象となるデータは，生成実験用に用意した入力文とそれに対して生成モデルが出力した文のペアである．生成実験用に用意した入力文が 20 文であり，各入力文に対して 2 つの生成モデルがそれぞれ 5 文生成しているため，各モデルにつき 100 ペア，合計して 200 ペアのデータに対して合成文生成を行ってもらう．

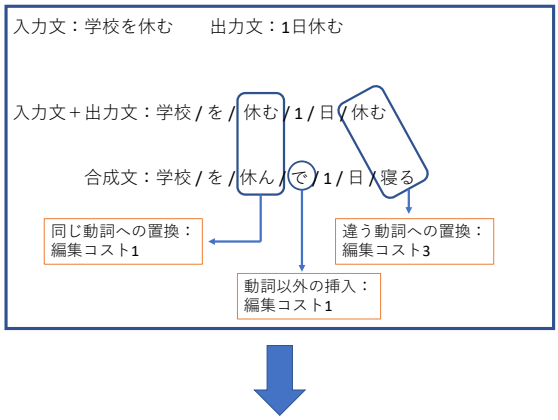

図 2　編集距離の算出例

Fig. 2　Example of edit distance calculation

5.2　評 価 方 法

入出力ペアに対して，入力と出力を単純につなげただけの文章と合成文とで編集距離を求める．編集距離にはいくつかの考え方があるが，本論文においてはレーベンシュタイン距離を用いる．レーベンシュタイン距離とは，2 つの文に対して，1 文字の挿入・削除・置換によって，一方の文字列をもう一方の文字列に変形するのに必要な手順の最小回数のことである．

編集距離を求めるにあたり，文字単位ではなく単語単位で計算を行う．これは，文字単位で編集距離を求めてしまうと，同じ内容語に対する置換にもかかわらず内容語の文字数の差だけ編集コストが算出されてしまうからである．また，置換や挿入といったそれぞれの編集コストにも重みをつける．これは，挿入が入出力ペアに対して情報を付与しているだけであることに対して，置換や削除は直接手を加えているということを示すためである．その他にも，本論文の問題設定がイベントに関していることから，動詞を編集する際のコストをより重く設定した．具体的な編集コストの内訳は以下である．また，編集コストの算出例を図 2 に示す．

- 編集コスト 1 : 動詞以外の挿入，名詞・動詞以外の削除，同じ動詞への置換（末尾の変化）
- 編集コスト 2 : 動詞以外の置換，動詞の挿入，名詞や動詞の削除
- 編集コスト 3 : 異なる動詞への置換

5.3　評価結果と考察

5.2 節で，編集距離を用いた評価方法について説明した．実際に評価方法を適用した結果を表 3 および表 4 に示す．表 3 は Seq2Seq の生成結果，表 4 は T5 の生成結果に対する評価をワーカーごとにまとめたものである．Dist (All) はワーカーが作成した合成文，および合成文を作成する際のもととなる入力文と出力文を単純にくっつけただけの文とで測った編集距離全体の平均のことを指している．Accept Num は文章判定において，ワーカーによって許容できる書き換えであると判断された合成文の数を示している．また，Dist (Accept) は文章判定によって許容できる書き換えだと判断された合成文に対する平均編集距離，Dist (UnAccept) は文章判定によって許容できない書き換えだと判断された合成文に対する平均編集距離を表している[2]．Dist (Accept) はワーカーによって許容された合成文がどれほど書き換えられているかを示しているため，小さいほどより望ましい．加えて，Dist (UnAccept) との差が大きくなるほど，Dist (Accept) / Dist (UnAccept) の関係と合成文の判定結果（許容できる / 許容できない）の関係が一致しているといえる．

表 3　Seq2Seq の出力に対する評価結果

Table 3　Evaluation results for Seq2Seq output

	Dist (All)	Accept Num	Dist (Accept)	Dist (UnAccept)
worker 1	5.08	85	4.35	9.20
worker 2	5.70	66	4.11	8.79
worker 3	6.18	56	4.54	8.27
worker 4	5.99	66	4.82	8.26
worker 5	3.44	84	3.02	5.63
worker 6	6.70	57	5.30	8.56
Avg.	5.52	69.0	4.27	8.28

表 4　T5 の出力に対する評価結果

Table 4　Evaluation results for T5 output

	Dist (All)	Accept Num	Dist (Accept)	Dist (UnAccept)
worker 1	4.83	87	4.30	8.38
worker 2	5.90	63	3.06	10.73
worker 3	6.28	53	4.23	8.60
worker 4	5.91	77	4.92	9.22
worker 5	3.56	76	2.74	6.17
worker 6	7.04	51	5.71	8.43
Avg.	5.59	67.8	4.10	8.72

また，表 3 より全てのワーカーに対して，Dist (Accept) は Dist (UnAccept) を大きく下回っている．これは，表 4 にも同様のことがいえる．ワーカーが許容できると判断した時の平均距離は低く，ワーカーが許容できないと判断した時の平均距離は高いことから，平均距離が低ければ低いほど，ワーカーも許容できる文章であるということがいえる．そのため，今回用いた単語単位における重み付き編集距離はワーカーが主観的に判定していた，合成文が許容できる書き換えであるか，という判定を数値的に表すことができると考えられる．

最後に，6 人のワーカーが行った文章判定の結果全体（200 文）について Fleiss' kappa [18] を用いて一致度（κ 値）を調べた．κ 値は 0.4269 という値を示した．κ 値の評価には様々な考え方が存在するが，Landis ら [19] によると，κ 値が 0.4 から 0.6 の範囲を指し示す場合，適度に一致しているとみなすことができる．6 人のワーカー全員による一致度であることを考えると，算出された κ 値はそれほど悪いものではなく，人手による評価でもそこそこ一致するということがわかる．これは，単純に入

(注2)：ただし，各モデルの Dist (All)，Dist (Accept)，Dist (UnAccept) について Welch 検定 [17] を用いて有意差検定（有意水準は 0.05）を行ったが，全てに関して算出された値は有意水準を上回っていたため，2 つのモデルの平均編集距離に有意差はないことが確認されている．

出力のペアを見比べるだけで判定を行うよりも，合成文の作成を行うことで 2 つのイベント文に対して上手く情報を補完できるようになったため，一致度が上がったのだと考えられる．

6. おわりに

本論文では，生成モデルを用いた自然言語推論として，まずはじめに Twitter を利用して前後関係を持ったイベント文のペアを獲得し，データセットを構築した．

その後，作成したデータセットを用いて Seq2Seq と T5 の 2 種類の生成モデルを学習させ，生成実験を行った．生成結果を確認すると，出力の多様性という点では Seq2Seq の方が高く，文法的に正しい文の出力という点においては T5 の方が優れていることがわかった．また，各モデルの出力文が入力文に対して適切なイベント文を生成することができたか確認したところ，単純に入出力のペアを見比べるだけでは，判断する人によって評価の変わる出力が存在したため，人手による評価は難しい．

そこで，編集距離を用いた入出力ペアに対する評価方法を提案し，実際に評価を行った．各モデルの平均編集距離について有意差検定を行ったところ，2 つのモデルに有意差は見られなかった．しかしながら，文章判定において許容できる書き換えだと判断された平均編集距離と許容できない書き換えだと判断された平均編集距離を比べたところ，両モデルの全てのワーカーにおいて，大きな差が見られた．このことから，平均距離が低ければ低いほどワーカーが許容できる文章であるとみなすことができる．そのため，本研究で提案した編集距離を用いた評価方法はワーカーが主観的に判定していた，合成文が許容できる書き換えであるか，という判定を数値的に表すことが可能であるといえる．また，文章判定の κ 値の結果より，人手による評価でもそこそこ一致していることが分かった．これは，文章判定を行う前に合成文を作成することで，情報を上手く補完することができるようになったため，一致度が高い値を示したものと考えられる．

今後の課題としては，各生成モデルの出力結果の改善が挙げられる．Seq2Seq や T5 の出力文において助詞の欠落が多く見てとれた．これは，学習に用いたデータセットの影響であると考えられる．そのため，イベント文ペアの収集方法について，係り受け情報を利用する際に名詞を含むチャンクには必ず助詞を含むように制約をかける，などの改善を行い，より質の高いデータを収集することが望ましい．また，T5 では似たようなイベント文の出力が見られた．そのため，得た出力リストに対してランキング化を行うなどして，似た表現の出力を抑えつつ，多様な回答を複数獲得することが必要である．

文　献

[1] Y. Zhang, Z. Ou, and Z. Yu, “Task-oriented dialog systems that consider multiple appropriate responses under the same context,” Proceedings of the AAAI Conference on Artificial Intelligence, vol.34, pp.9604–9611, 2020.

[2] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, “Supervised learning of universal sentence representations from natural language inference data,” Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp.670–680, 2017.

[3] S.R. Bowman, G. Angeli, C. Potts, and C.D. Manning, “A large annotated corpus for learning natural language inference,” Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp.632–642, 2015.

[4] A. Williams, N. Nangia, and S. Bowman, “A broad-coverage challenge corpus for sentence understanding through inference,” Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp.1112–1122, 2018.

[5] P. Rajpurkar, R. Jia, and P. Liang, “Know what you don’t know: Unanswerable questions for SQuAD,” Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp.784–789, 2018.

[6] R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi, “SWAG: A large-scale adversarial dataset for grounded commonsense inference,” Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp.93–104, 2018.

[7] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi, “HellaSwag: Can a machine really finish your sentence?,” Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp.4791–4800, 2019.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp.4171–4186, 2019.

[9] M. Boratko, X. Li, T. O’Gorman, R. Das, D. Le, and A. McCallum, “ProtoQA: A question answering dataset for prototypical commonsense reasoning,” Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pp.1122–1136, 2020.

[10] B.Y. Lin, W. Zhou, M. Shen, P. Zhou, C. Bhagavatula, Y. Choi, and X. Ren, “CommonGen: A constrained text generation challenge for generative commonsense reasoning,” Findings of the Association for Computational Linguistics: EMNLP 2020, pp.1823–1840, 2020.

[11] K. Yamamoto and K. Shimada, “Acquisition of periodic events with person attributes,” Proceedings of the 2020 International Conference on Asian Language Processing, pp.229–234, 2020.

[12] 工藤　拓，松本裕治，“チャンキングの段階適用による日本語係り受け解析，” 情報処理学会論文誌，vol.43，no.6，pp.1834–1842，2002.

[13] I. Sutskever, O. Vinyals, and Q.V. Le, “Sequence to sequence learning with neural networks,” Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, p.3104–3112, 2014.

[14] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P.J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” Journal of Machine Learning Research, vol.21, no.140, pp.1–67, 2020.

[15] T. Luong, H. Pham, and C.D. Manning, “Effective approaches to attention-based neural machine translation,” Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp.1412–1421, 2015.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Proceedings of the 31st International Conference on Neural Information Processing Systems, p.6000–6010, 2017.

[17] B.L. Welch, “The significance of the difference between two means when the population variances are unequal.,” Biometrika, vol.29, no.3-4, pp.350–362, 1938.

[18] J.L. Fleiss, “Measuring nominal scale agreement among many raters.,” Psychological Bulletin, vol.76, pp.378–382, 1971.

[19] J.R. Landis, “The measurement of observer agreement for categorical data,” Biometrics, vol.33, no.1, pp.159–174, 1977.