事態の一貫性推定に基づく雑談対話応答選択モデル

Conversational Response Selection Model Based on Event Coherency Estimation

田中翔平 [1*] 吉野幸一郎 [1,2] 須藤克仁 [1] 中村哲 [1]
Shohei Tanaka[1] Koichiro Yoshino[1,2] Katsuhito Sudoh[1] Satoshi Nakamura[1]

[1] 奈良先端科学技術大学院大学
[1] Nara Institute of Science and Technology
[2] 科学技術振興機構 さきがけ
[2] PRESTO, Japan Science and Technology Agency

Abstract: Dialogue systems need to attract users' interest by maintaining coherency of system responses. In this paper, we propose novel methods to select coherent responses for a given dialogue context. The methods improve coherency and dialogue continuity using related event pairs, such as "be stressed out" and "relieve stress." We used two re-ranking methods to estimate coherency. The first method estimates coherencies of event pairs by matching with event causality pairs, which are extracted from a large-scale corpus statistically. The second method estimates coherencies of responses to their dialogue contexts using Coherence Model. Experimental results showed that the method based on event causality pairs can select responses in the higest coherency and dialogue continuity.

1 はじめに

Neural Conversational Model (NCM) を始めとする，ニューラルネットワークで対話のクエリ-応答ペアを学習する対話モデルが盛んに研究されている．しかし，こうした対話モデルはしばしば対話の文脈や論理を考慮せず，どのような場合にでも当てはまる単純な応答を生成してしまうという問題が知られている．そこで本論文では，文脈や論理を考慮した応答を，対話モデルの生成する応答候補からリランキングにより選択する手法を提案する．リランキングは，質問応答システムや対話システムなどの言語生成タスクにおいて様々な要素を考慮した候補の選択に用いられる．提案手法は応答候補と対話履歴に存在する事態の一貫性に基づきリランキングを行う．

本研究では,「ストレスが溜まる」と「発散する」など，関連すると認められる事態ペアが対話履歴と応答候補の間に存在する場合，対話中の事態の一貫性が高いと考える．この事態間関係の一つとして，因果関係がある．因果関係とは 2 つの事態間に原因と結果の関係が成立することと定義され，この定義に従い,「ストレスが溜まる」が原因,「発散する」が結果，のように認定する．因果関係はこれまで質問応答システムなどで利用されており，質問と応答の間に成立する因果関係を考慮することで，質問に対する適切な応答を生成できることが示されている [1]．雑談対話システムにおいても因果関係を考慮することで，文脈に沿った応答を生成できることが示されている [2]．本研究では一貫性に加え，雑談対話システムにとって重要な対話を継続する働き（対話継続性）が向上することも期待する．

また一貫性推定に関する研究として，Coherence Model [3] がある．このモデルは文書中に出現する単語の品詞情報や文の分散表現をもとに，ある文の文書における一貫性を推定する．対話においてもこの一貫性推定は有効であることが知られており [4]，これを用いて応答候補の対話履歴に対する一貫性を推定することを考える．

本論文で提案する手法は，NCM によって生成された N-best 応答候補より，一貫した，対話継続性の高い応答を選択するものである．提案手法では，対話履歴に対し一貫した応答を選択するために，事態の一貫性を考慮したスコアの計算を行い，これに基づいて応答候補から応答を選択する．事態の一貫性の考慮を行うため，大規模コーパスから統計的に獲得された因果関係ペア [5] を用いる．この際，単純にこれらのペアを用いるとカバレージの問題が生じるため，Role Factored Tensor Model (RFTM) [6] を用いた事態の分散表現によって汎化を行う．本論文では，事態を述語と付随する格要素のペアと定義する．また上述の事態の一貫性のみを考慮したリランキングでは応答全体の一貫性が

*連絡先： 奈良先端科学技術大学院大学
奈良県生駒市高山町８９１６番地の５
E-mail: tanaka.shohei.tj7@is.naist.jp

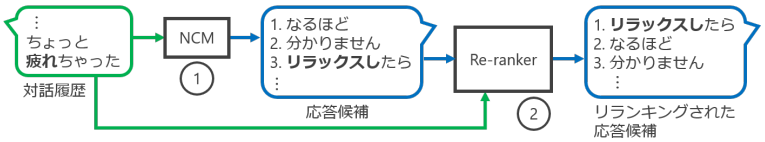

図 1: Neural Conversational Model+ リランキング; 「疲れる」と「リラックスする」が関連した事態であるという知識に基づき応答を選択.

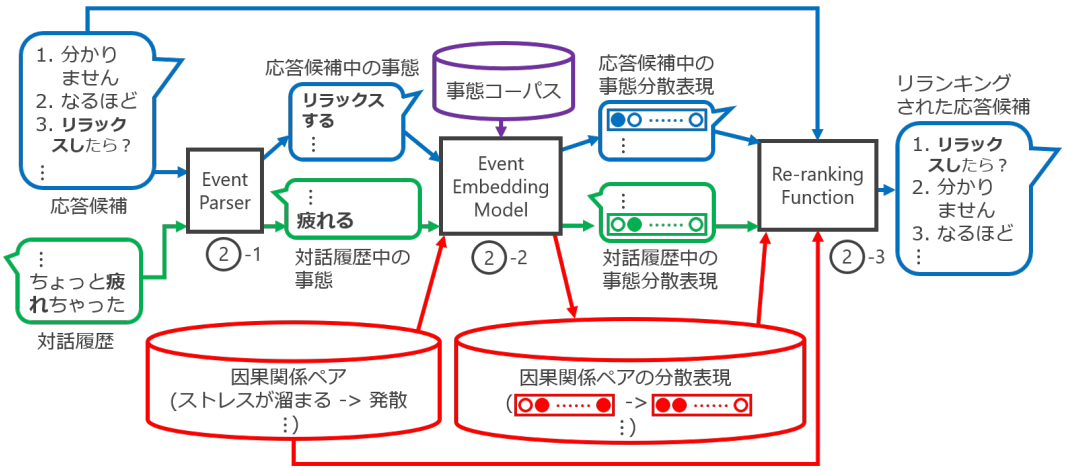

図 2: 因果関係ペアを用いたリランキング; 「疲れる」→「リラックスする」という因果関係が対話履歴との間に成立する応答がリランキングにより選択される.

低下する可能性があるため，異なるリランキング手法として Coherence Model に基づく応答候補の一貫性推定を提案する．自動評価及び人手評価の結果，因果関係ペアを用いたリランキングにより応答の一貫性，対話継続性が最も向上することが示された．

2 事態の一貫性に基づく応答のリランキング

図 1 に提案手法の概要を示す．提案手法は大きく分けて 2 つのパートから構成される．まず対話履歴をもとに既存の NCM モデルから N-best 応答候補を生成する（図 1 ①）．次に応答候補を事態の一貫性に基づきリランキングする（図 1 ②）．このリランキングのために，本研究では 2 つの異なる手法を用いる．1 つ目の手法は事態の一貫性に関する外部知識として，統計的に獲得された因果関係ペアを用いるリランキングである（2.1 節）．2 つ目の手法は事態間の関係のみでなく，Coherence Model によって対話全体の一貫性も評価するリランキングである（2.2 節）．

2.1 因果関係ペアを用いるリランキング

このリランキング手法では，まず対話履歴と応答候補に含まれる事態（述語項構造）を事態パーサーを用いて抽出する（図 2 ②-1）．この事態パーサーには KNP[1] を用いる．その後，抽出した事態及び因果関係ペアを事態埋め込みモデルを用いて分散表現に変換する（図 2 ②-2; 2.1.2 節）．事態埋め込みモデルとして，RFTM を利用する．最後に応答候補を因果関係に基づきリランキングする（図 2 ④; 2.1.1, 2.1.3 節）．

表 1: 因果関係の一例

述語 1	項 1	述語 2	項 2	$lift$
溜まる	ガ:ストレス	発散	-	10.02

2.1.1 因果関係ペア

柴田ら [5] が提案した，共起情報と格フレームに基づき自動獲得された因果関係ペアデータセットをリランキングに用いる．このデータセットは約 16 億文の Web テキストから抽出された約 42 万件の因果関係知識で構成されている．表 1 に因果関係ペアの一例を示す．各事態は述語項構造により表現され，述語 1 及び項 1 は原因となる事態を，述語 2 及び項 2 は結果となる事態を表す．ここで各事態は述語を必ず含むが，項（ガヲニデ格のいずれか）は含まない場合もある．$lift$ は 2 つの事態の自己相互情報量であり，事態間の因果関係としての結びつきの強さを表す．$lift$ を用い，リランキングのためのスコアの計算を次のように定義する．

$$score(\mathrm{h}, \mathrm{r}) = \max_{<\mathrm{e}_h, \mathrm{e}_r>} \frac{\log_2 \mathrm{P}(\mathrm{r}|\mathrm{h})}{\left(\log_2 lift(\mathrm{e}_h, \mathrm{e}_r)\right)^{\lambda}}. \quad (1)$$

[1] http://nlp.ist.i.kyoto-u.ac.jp/?KNP

P(r|h) は対話履歴 h を与えたときに NCM が生成した各応答候補 r の事後確率であり，λ は因果関係の重みを決定するハイパーパラメータである．$lift(\mathrm{e}_h, \mathrm{e}_r)$ は対話履歴中の事態 e_h と応答候補中の事態 e_r との間の $lift$ の値である．この事態ペアが因果関係ペアに含まれない場合，$lift$ の値は 2 とする．ただし $lift(\mathrm{e}_h, \mathrm{e}_r)$ は値域が広い $(10 < lift(\mathrm{e}_h, \mathrm{e}_r) < 10,000)$ ため，対数をとった値を使用する．応答候補と対話履歴との間に複数の因果関係ペアが含まれる場合，$lift(\mathrm{e}_h, \mathrm{e}_r)$ の値が最も大きい因果関係のみを考慮する．このモデルを “Re-ranking (Pairs)” と呼ぶ．

2.1.2 Role Factored Tensor Model (RFTM) に基づく事態分散表現

大規模テキストから抽出した大規模因果関係ペアデータセットであっても，あらゆる因果関係ペアを網羅できるわけではないため，これのみを用いて対話履歴と応答候補に存在する全ての因果関係を考慮することは難しい．そこで因果関係ペア，および発話中に含まれる事態を分散表現に変換し，ベクトル空間中で因果関係知識と対話中に出現した因果関係との類似度に基づくマッチングを行うことで，表層の一致しない因果関係に対するマッチングを実現する．

本論文では，事態を述語，もしくは述語と付随する格要素のペアと定義して用いる．格要素 a は GloVe によりベクトル v_a へと変換される．述語 p は predicate embedding によりベクトル v_p へと変換される．predicate embedding は Skip-gram をもとにした単語分散表現である．図 3 に predicate embedding モデルの概要を示す．このモデルは Skip-gram が与えられた単語の周辺単語を予測するよう学習を行うのと同様に，与えられた述語に付随する格要素を予測するよう学習を行う．

v_p および v_a より事態の分散表現を得る手法として，Weber らが提案した RFTM [6] を利用する．RFTM は述語と項を次式により事態分散表現 e へと変換する．

$$\mathrm{e} = \sum_a W_a T(\mathrm{v}_p, \mathrm{v}_a). \quad (2)$$

述語と付随する格要素の関係は 3 階パラメータテンソル T, パラメータ行列 W_a により計算される．述語が格要素を持たない場合，e は v_p により代替される．RFTM では学習の目標として連続して起こる事態を予測する．これは単語における分布仮説同様，似た文脈に出現する事態が似た意味を持つことを仮定するものである．これにより，似た文脈を持つ事態を潜在空間上の近い位置に埋め込むことが出来る．

2.1.3 事態分散表現を用いた因果関係のマッチング

図 4 に事態分散表現による事態のマッチングを示す．提案手法は事態分散表現に基づき，応答候補と対話履歴中の発話との間の事態ペアに対し，最も高いコサイン類似度を持つ因果関係を因果関係ペアより選択する．ここで 2 つの事態間の $lift_{emb}$ を次の式のように定義する．

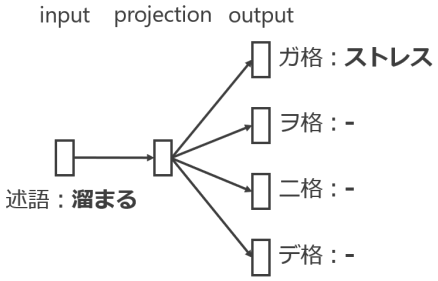

図 3: Predicate Embedding

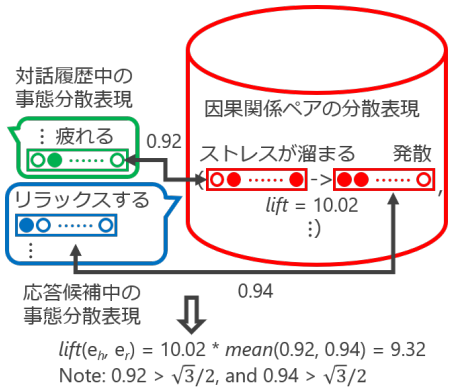

図 4: 因果関係のマッチング; 「疲れる」→「リラックスする」という因果関係の $lift$ は最もコサイン類似度が高い因果関係である「ストレスが溜まる」→「発散」の $lift$ から計算される．

$$lift_{emb}(\mathrm{e}_h, \mathrm{e}_r) = lift(\mathrm{e}_c, \mathrm{e}_e) * mean\big(sim(\mathrm{e}_h, \mathrm{e}_c), sim(\mathrm{e}_r, \mathrm{e}_e)\big). \quad (3)$$

e_h は対話履歴中の事態，e_r は応答候補中の事態であり，e_c と e_e はそれぞれ因果関係ペア中の原因となる事態，結果となる事態である．sim はベクトル間のコサイン類似度である．提案手法では対話履歴中の事態が結果，応答候補中の事態が原因となる場合も考慮する．ただし「風邪を引く」と「目が覚める」を同一とみなすなど，事態を過剰に汎化してしまうことを避けるために，各 sim はしきい値を設定する．式 (1) の $lift(\mathrm{e}_h, \mathrm{e}_r)$ を $lift_{emb}(\mathrm{e}_h, \mathrm{e}_r)$ で更新することで，事態分散表現を用いたリランキングスコアは次のように定義される．

$$score(\mathrm{h}, \mathrm{r}) = \max_{<\mathrm{e}_h, \mathrm{e}_r>} \frac{\log_2 \mathrm{P}(\mathrm{r}|\mathrm{h})}{\big(\log_2 lift_{emb}(\mathrm{e}_h, \mathrm{e}_r)\big)^{\lambda}}. \quad (4)$$

このモデルを “Re-ranking (RFTM)” と呼ぶ．

2.2 Coherence Model を用いるリランキング

因果関係の定義の難しさや，RFTM が事態を過汎化する可能性があることから，“Re-ranking (RFTM)” で

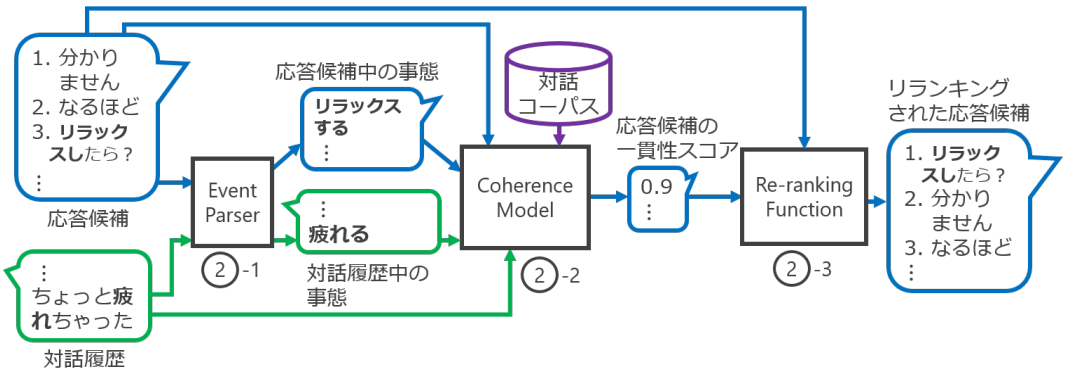

図 5: Coherence Model を用いたリランキング; 「疲れる」「リラックスする」という事態ペアの一貫性に加え，対話全体の一貫性も考慮して応答を選択.

用いられる因果関係は必ずしも正確ではない．また因果関係ペアを用いたリランキングは応答候補中に出現する事態ペアの一貫性のみに着目しているため，選択された応答候補全体が持つ意味が対話履歴に対して一貫していないことも考えられる．そこで事態ペアのみでなく，応答全体の一貫性も評価するリランキングを実現する．このリランキングでは因果関係ペアを用いたリランキングと同様，はじめに対話中の事態を事態パーサーにより抽出する（図 5 ②-1）．次に，抽出した事態，対話履歴，応答候補に基づき，応答候補の対話履歴に対する一貫性スコアを推定する（図 5 ②-2; 2.2.1 節）．一貫性スコアの推定には Coherence Model を利用する．最後に応答候補を一貫性スコアに基づきリランキングする（図 5 ②-3; 2.2.1 節）.

2.2.1 Coherence Model による対話の一貫性推定

応答候補の対話履歴に対する一貫性を推定するために，Xu らが提案した Coherence Model [3] を利用する．Xu らの研究において，このモデルは Web テキストなどの文書と文のペアの一貫性推定に使用された．学習に用いる正例は文書とその文書に連続する一文のペアであり，負例は文書とその文書に連続しない一文のペアである．これに対し本研究では，Coherence Model を応答候補の対話履歴に対する一貫性推定に用いる．モデルの学習に用いる正例，負例の例を図 6 に示す．正例は対話履歴と対話履歴中の発話に対し因果関係を持つ応答候補のペアである．因果関係のマッチングには “Re-ranking (Pairs)” と同じく因果関係ペアデータセットを用いる．負例はこれを学習データに出現する他の発話と入れ替えたペアである．これにより，含まれる事態と全体の意味の両方が対話履歴に対して一貫している応答候補のみ，一貫性スコアを高く見積もることが期待できる．

図 7 にモデルの概要を示す．このモデルはまず対話履歴 h，応答候補 r を BERT を用いて分散表現に変換する．またこれに加えて，対話履歴，応答候補から抽出された事態ペアを RFTM を用いて分散表現に変換

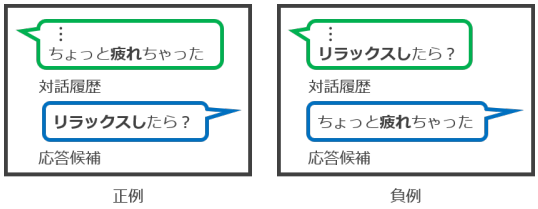

図 6: Coherence Model の学習に用いる正例と負例; 負例の応答に含まれる事態は対話履歴に対して一貫しているが，応答全体の意味は一貫していない.

する．このとき事態の一貫性を保証するために，対話履歴中の事態 e_h，応答候補中の事態 e_h のコサイン類似度はしきい値以上のもののみを用いる．RFTM は同じ文脈を持つ事態を分散表現上の近い点に埋め込むため，連続して起こりやすい，一貫した事態ペアのコサイン類似度は高くなると考えられる．得られた分散表現に基づき，応答候補の一貫性スコア coh を次式のように計算する.

$$coh(\mathrm{h},\mathrm{r},\mathrm{e}_h,\mathrm{e}_r) = \sigma\big(W\mathrm{v}+b\big). \tag{5}$$

$$\mathrm{v} = [\mathrm{h};\mathrm{r};\mathrm{h}-\mathrm{r};|\mathrm{h}-\mathrm{r}|;\mathrm{h}*\mathrm{r} ;\mathrm{e}_h;\mathrm{e}_r;\mathrm{e}_h-\mathrm{e}_r;|\mathrm{e}_h-\mathrm{e}_r|;\mathrm{e}_h*\mathrm{e}_r] \tag{6}$$

ここで σ はシグモイド関数を，W, b はそれぞれパラメータ行列，パラメータバイアスを表す．また $[;]$ はベクトルの結合を表し，$*$ は要素積を表す．対話履歴，応答候補から抽出される事態ペアが複数存在する場合，最も高いコサイン類似度を持つ事態ペアのみを用いる．この一貫性スコアの計算には図 7 のように Multi Layer Perceptron (MLP) を使用する．応答候補の一貫性スコア $(0 \leq coh \leq 1)$ が 0.5 以上であった場合，その応答候補は対話履歴に対し一貫していると判定し，リランキングスコアを次式のように計算する.

$$score(\mathrm{h},\mathrm{r}) = \big(1-coh(\mathrm{h},\mathrm{r},\mathrm{e}_h,\mathrm{e}_r)\big)\log_2 \mathrm{P}(\mathrm{r}|\mathrm{h}). \tag{7}$$

このモデルを “Re-ranking (Coherence)” と呼ぶ.

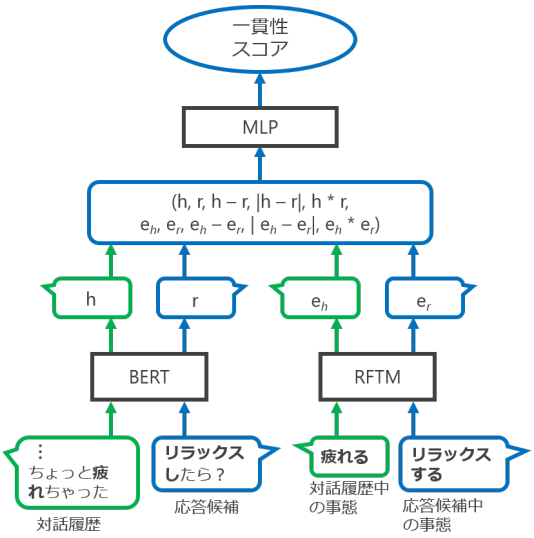

図 7: Coherence Model による一貫性スコアの推定

3 実験

提案手法による事態間関係を用いた対話応答リランキングの有効性を検証するため，リランキングの有無による対話応答の違いを自動評価，人手評価する実験を行った．実験では NCM として Encoder-Decoder with Attention (EncDec) と Hierarchical Recurrent Encoder-Decoder (HRED) を用いる．HRED のモデルは，単純な Encoder-Decoder などのモデルと比較して対話履歴を考慮した応答を生成しやすいと期待される一方で，出力結果のバリエーションが対話履歴により制約され，N-best 応答候補のリランキングには不向きである可能性もある．

対話モデルの学習及びテストに用いるコーパスとしてマイクロブログ (Twitter) から収集した 2,072,893 対話を使用した．平均対話長は 13.50 ターン, 平均発話長は 22.52 文字である．語彙サイズを削減しモデルの学習を促進するために，絵文字などはあらかじめ発話から除外した．対話コーパスを学習データ，バリデーションデータ，テストデータとしてそれぞれ 1,969,626 対話，51,573 対話，51,694 対話に分割した．RFTM が利用する GloVe, predicate embedding の学習には日本語 Wikipedia ダンプデータ[2]を用い，RFTM の学習には因果関係ペア，毎日新聞 2017 データ集[3]に加え，対話モデルの学習に用いたものと同様の対話データを用いた．Coherence Model が用いる BERT モデルには事前学習済みの公開されているモデル[4]を用い，Coherence Model の学習には対話モデルの学習に用いたものと同様の対話データを用いた．

[2]2018 年 11 月 2 日時点の最新版
[3]http://www.nichigai.co.jp/sales/mainichi/mainichi-data.html
[4]http://nlp.ist.i.kyoto-u.ac.jp/index.php?BERT 日本語 Pre-trained モデル

3.1 自動評価

自動評価として，まず提案手法が有効な範囲を測るためリランキングされた応答候補の割合 (“re-ranked”) を用いた．また，応答候補の reference に対する類似度を測るため，reference に対する BLEU, NIST, また greedy, average, extrema と呼ばれる 3 つの分散表現を用いた指標 [7] を用いた．また評価指標として，dist-n, Pointwise Mutual Information (PMI) も用いた．dist-n, PMI はそれぞれ応答の多様性，一貫性を測るために用いた．応答と対話履歴の PMI は次式のように計算される．

$$\mathrm{PMI} = \frac{1}{\#wr} \sum_{wr}^{\#wr} \max_{wh} \mathrm{PMI}(wr, wh). \qquad (8)$$

wr, wh はそれぞれ応答中，対話履歴中の単語を表す．

表 2 に全テストデータに対するリランキング前後の自動評価による比較を示す．表 2 の手法名は左から順に，用いた NCM，用いたリランキング手法を示している．“1-best” はリランキングを行わない，ベースラインの NCM を表す．“Re-ranking (Pairs)”, “Re-ranking (RFTM)”, “Re-ranking (Coherence)” はそれぞれ因果関係ペアのみを用いたリランキング，因果関係ペアと RFTM を用いるリランキング，Coherence Model を用いるリランキングを表す．

リランキングされる応答候補の割合は “Re-ranking (Pairs)”, “Re-ranking (Coherence)” の場合多くとも 10% 前後に留まり，リランキングの効果が限定的となる．これに対して，RFTM による分散表現で汎化を行ったモデルでは，リランキングの割合が 30% 程度に上昇している．また NIST, dist-2 および PMI は “Re-ranking (RFTM)” により最も上昇しており，語彙の組み合わせが多様かつ対話履歴と関連したものになっていることが分かる．

3.2 人手評価

自動評価のみで対話システムの性能を評価することは困難である．そこで，ベースラインモデルと提案モデルを人手評価により比較することで，提案モデルにより選択された応答の一貫性，対話継続性を測った．ベースラインとして HRED を用い，提案するそれぞれのリランキング手法と比較した．評価者の負担を軽減するため，内容を理解するために外部知識を必要する対話は評価対象から取り除いた．人手評価にはクラウドソーシングを用い，10 人のクラウドワーカーに 2 つのシステムの応答を比較し，次に挙げる 2 点の指標をより満たすものを 2 つの応答のどちらか，もしくは「どちらでもない (neither)」の 3 択より選んでもらった．1 番目の指標は「どちらの応答に含まれる単語がより対話履歴に関連しているか (word coherency)」であり，これはシステム応答が一貫しているかを計測するため

表 2: 全テストデータに対する自動評価のスコア

Method NCM	re-ranking	Evaluation re-ranked (%)	BLEU	NIST	greedy	average	extrema	dist-1	dist-2	PMI
reference	-	-	-	-	-	-	-	0.09	0.43	2.26
EncDec	1-best	-	1.24	0.27	**0.46**	**0.56**	**0.46**	0.07	0.12	1.45
	Re-ranking (Pairs)	3,058 (8.86)	1.28	0.27	0.45	0.55	0.45	0.07	0.13	1.50
	Re-ranking (RFTM)	**11,354(32.90)**	1.46	0.42	0.44	0.54	0.44	**0.08**	0.16	1.73
	Re-ranking (Coherence)	2,667 (7.73)	1.24	0.31	**0.46**	**0.56**	**0.46**	0.07	0.13	1.51
HRED	1-best	-	**1.58**	2.64	0.44	**0.56**	0.45	**0.08**	0.19	1.60
	Re-ranking (Pairs)	2,608 (7.56)	1.56	2.62	0.44	**0.56**	0.45	**0.08**	0.19	1.63
	Re-ranking (RFTM)	11,247 (32.59)	1.57	**2.73**	0.44	**0.56**	0.45	**0.08**	**0.20**	**1.75**
	Re-ranking (Coherence)	3,245 (9.40)	1.53	2.61	0.45	**0.56**	**0.46**	**0.08**	0.19	1.64

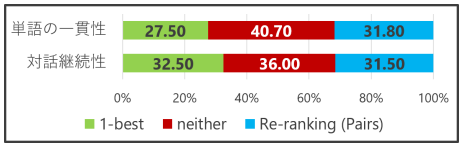

図 8: 1-best v.s. Re-ranking (Pairs)

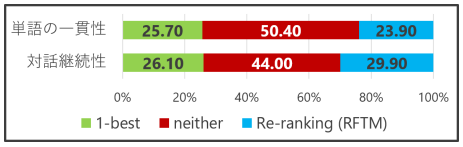

図 9: 1-best v.s. Re-ranking (RFTM)

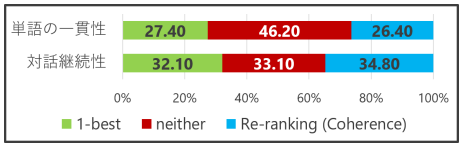

図 10: 1-best v.s. Re-ranking (Coherence)

に用いる．2 番目の指標は「どちらの応答により返答したいと思うか (dialogue continuity)」であり，これはシステム応答の対話継続性が高いかを計測するために用いる．これらの指標は Alexa Prize を参考に決定した．また，各比較で対象とした対話数はそれぞれ 100 である．なお，これらの評価は各手法により 1-best と異なる応答が選択されたケースのみを評価として用いたので，評価の対象となったサンプルがそれぞれ異なる．このため，異なる図間のスコアを直接比較できないことに注意されたい．

人手評価の結果を図 8 から 10 に示す．単語の一貫性は “Re-ranking (Pairs)” で上昇している一方，“Re-ranking (RFTM)” では減少している．これは因果関係ペアでもともと因果関係と認められているものは一貫性の改善に役立つものの，汎化された因果関係には因果関係と認めにくいものが多く含まれてしまうからだと考えられる．また “Re-ranking (Coherence)” でもわずかに一貫性が減少しているが，これは “Re-ranking (Coherence)” がリランキング対象とする対話は 1-best の時点である程度高い一貫性を持つためだと考えられる．

これに対し，対話継続性は “Re-ranking (RFTM)”, “Re-ranking (Coherence)” において向上しており，単純でつまらない応答の割合が減少していることがわかる．

4 おわりに

本論文では，ニューラル雑談対話モデル (NCM) により生成された N-best 応答を，連続する事態の一貫性に基づきリランキングする手法を提案した．提案手法は述語項構造で表現された事態に基づいているため，関連する事態ペアの外部知識を用意すれば構文解析器を持つあらゆる言語に適用可能である．実験の結果，事態の一貫性に基づくリランキングにより，一貫した対話継続性の高い応答が選択できることを確認した．今後は一貫した対話中の事態を生成した上で応答生成を行う生成的アプローチについて検討していく．

謝辞

本研究で使用した因果関係ペアをご提供頂いた京都大学黒橋研究室の黒橋禎夫教授，柴田知秀博士に感謝いたします．

本研究は JST さきがけ (JPMJPR165B) の支援を受けた．

参考文献

[1] Jong-Hoon Oh, Kentaro Torisawa, Canasai Kruengkrai, Ryu Iida, and Julien Kloetzer. Multi-Column Convolutional Neural Networks with Causality-Attention for Why-Question Answering. In *Proceedings of the 10th Association for Computing Machinery International Conference on Web Search and Data Mining (WSDM)*, pp. 415–424, 2017.

[2] 佐藤祥多, 乾健太郎. 因果関係に基づくデータサンプリングを利用した雑談応答学習. 言語処理学会 第 24 回年次大会 発表論文集 (ANLP), pp. 1219–1222, 2018.

[3] Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. A Cross-Domain Transferable Neural Coherence Model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 678–687, 2019.

[4] Alessandra Cervone, Evgeny Stepanov, and Giuseppe Riccardi. Coherence Models for Dialogue. In *Proceedings of INTERSPEECH 2018 (INTERSPEECH)*, 2018.

[5] Tomohide Shibata, Shotaro Kohama, and Sadao Kurohashi. A Large Scale Database of Strongly-Related Events in Japanese. In *Proceedings of the 9th International Conference on Language Resources and Evalu ation (LREC)*, 2014.

[6] Noah Weber, Niranjan Balasubramanian, and Nathanael Chambers. Event Representations with Tensor-Based Compositions. In *Proceedings of the 32nd Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence (AAAI)*, 2018.

[7] Chia-Wei Liu, Ryan Lowe, Iulian V. Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. How NOT to Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.