

積極的・消極的な性格を持つ自由対話システムの構築に向けて

吉田 寛太 白井 清昭

北陸先端科学技術大学院大学 先端科学技術研究科
{s2110184,kshirai}@jaist.ac.jp

概要

本論文は、ペルソナを含まない対話コーパスから積極的あるいは消極的な性格を持つ自由対話システムを構築する手法について述べる。Twitter から収集された擬似的な発話と応答の組からなるコーパスを用意し、応答の極性を中立、肯定、否定のいずれかに分類し、その分類結果に基づいて発話と応答の組を3つのサブコーパスに分割する。次に、大規模な中立のサブコーパスを用いて対話モデルを学習後、小規模な肯定または否定のサブコーパスを用いて対話モデルを再学習することで、積極的または消極的な応答を返す自由対話システムを獲得する。

1 はじめに

自由対話システムは、様々な話題についてユーザーと雑談するシステムであり、自然言語処理研究分野における重要な研究課題のひとつである。近年では、ウェブやソーシャルメディアから大量の対話の例を収集し、それを元にユーザーの発話に対してシステムの応答を返す系列変換モデルを学習する研究が多い。このようなアプローチは自由対話システムの品質を飛躍的に向上させた一方、応答の個性や一貫性が欠けやすいという問題がある [1]。例えば、自由対話システムと長い時間雑談をすると、システムの性格が変わるような印象を受けたり、少し前の発話と矛盾するような発言をしたりすることがある。これは、自動収集された対話データには様々な性格や個性を持つ不特定多数のユーザーの対話が混在していること、多くの先行研究では文脈を考慮せずにユーザーの1つの発話に対して応答を生成するシステムを学習するために矛盾を生じる発言が生成されやすいことなどが原因として考えられる。

このような問題を解決するために、ペルソナを用いて個性や一貫性を持つ自由対話システムを実現した研究がある [2, 3]。ペルソナとはシステムが想定

する話者のパーソナリティに関する知識である。あらかじめ用意したペルソナに沿うようにユーザーに対話させたデータを用いて対話モデルを学習する研究が多いが、その作成コストの高さが問題となる。また、どのような知識をペルソナとして用意しておくべきかは難しい課題である。

本研究は、ペルソナを参照しない方法で個性を有する自由対話システムを学習する手法を提案する。個性には様々なものがあるが、本研究では「積極的な性格」または「消極的な性格」に着目する。積極的な性格を持つ対話システムとは、ユーザーの応答に対してそれを支持したり励ますような発言をするシステムを、消極的な性格を持つ対話システムとは、否定したり賛意を示さない発言をするシステムを想定する。例えば、「沖縄旅行に行きたい」というユーザーの発話に対し、前者は「いいですね。私も行きたいです。」と応答し、後者は「私は出不精なので行きたくないです。」と応答する。積極的な対話例や消極的な対話例を自動収集し、これを訓練データとして系列変換モデルを学習することで、積極的・消極的な性格を持つ対話システムを実現する。

2 関連研究

ペルソナを用いて自由対話システムによる応答に個性や一貫性を与える手法が研究されている。Liらは、深層学習に基づく対話システムの応答に一貫性を持たせるためにペルソナを利用する手法を提案した [1]。Long Short-Term Memory (LSTM) による Encoder-Decoder モデルをベースとし、Decoder の入力に一つ前の時刻の単語埋め込みに加えて話者埋め込みを与えることで、話者に応じて一貫した応答を生成させた。対話モデルは話者の ID が付与された対話データから学習した。

Zhang らは、クラウドワーカーにプロフィール文の集合によって設定された人物を演じながらテキストチャットを行わせて収録した PERSONA-CHAT

というデータセットを構築し、それから対話モデルを学習することでシステム応答に個性を与える手法を提案した [2]. 対話モデルとして、Key-value memory network などを利用して複数の発話候補の中から最適な応答を選択するランキングモデルと、プロフィール文を参照する Memory network などを利用して応答文を出力する生成モデルを学習した. 被験者実験の結果、流暢性、(対話としての) 魅力性、一貫性のいずれにおいても、PERSONA-CHAT から学習したモデルは Twitter や映画の字幕などの対話データから学習したモデルよりも優れていることを示した.

Xu らはユーザとの長期にわたる雑談においても対話システムの個性を維持する手法を提案した [3]. ボット役とユーザ役の二人のクラウドワーカーにボット・ユーザのペルソナを参照しながら対話することを依頼し、その対話を収録した. さらに、対話における応答がペルソナを参照しているか、参照しているときはどのペルソナの文を参照しているかをアノテーションした. 次に、対話履歴からペルソナの文を抽出してメモリとして保存するモジュール、応答を生成するときにペルソナのメモリから文の生成に用いるプロフィール文を抽出するモジュール、このプロフィール文を参照しつつ系列変換モデルによって応答を生成するモジュールから構成される PLATO-LTM と呼ぶ対話システムを設計し、収録した対話コーパスを用いてこれを学習した. 被験者実験の結果、ペルソナのメモリ機能を導入することで会話の一貫性と魅力性の評価が向上したと報告している.

先行研究ではペルソナにしたがってユーザに対話を行わせたコーパスを用いて自由対話システムを学習しているが、そのようなコーパスの作成コストは高い. 1 節で述べたように、本研究では、設定する性格は積極的、消極的の 2 つに限定するが、ペルソナの情報を参照して構築したコーパスを用いずに個性を持つ対話システムを実現することを目指す.

3 提案手法

提案手法の概要を図 1 に示す. 図中の用語の正確な定義は 3.1 項以降で後述する. まず、Twitter におけるツイートとそれに対するリプライの組を対話における発話と応答の組とみなし、これを自動的に収集し、Twitter 対話コーパスとする. 次に、応答のツイートの極性判定を行い、「肯定」「否定」「中立」の

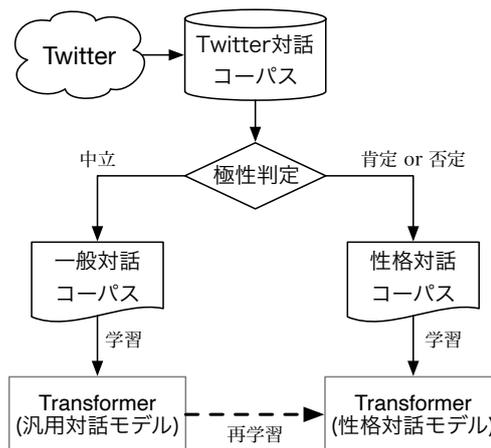


図 1 提案手法の概要

いずれかに分類する. 応答が「肯定」または「否定」と判定された発話対を抜粋して性格対話コーパスを作成する. 同時に、応答が「中立」と判定された発話対を抜粋して一般対話コーパスを作成する.

自由対話システムのための系列変換モデルを学習する. まず、一般対話コーパスを用いて発話に対する応答を返す汎用対話モデルを学習する. 次に、汎用対話モデルを初期のパラメタとして、性格対話コーパスを訓練データとして系列変換モデルを再学習することで、性格対話モデルを得る. 肯定的な応答を含む性格対話コーパスからは積極的な性格を持つ対話システムが、否定的な応答を含む性格対話コーパスからは消極的な性格を持つ対話システムが得られる.

3.1 Twitter からの対話コーパスの収集

Twitter から収集された対話コーパス [4] を利用する. 同コーパスは Twitter におけるツイートとリプライの連鎖を擬似対話として収集したデータであり、簡単なヒューリスティクスによって不適切な対話を除外している. 3 つ以上の発話からなる対話を含むが、本研究ではこれを 2 つの発話の組に分割して使用する. ツイートに対する前処理として、URL、スクリーンネーム、ハッシュタグ、絵文字、顔文字などを除去する. さらに、形態素数が 3 以下または記号、空白、数字のみからなるツイートを含む発話対を除去する. 最終的に 642,722 の発話対から構成されるコーパスを作成した. 本論文ではこれを「Twitter 対話コーパス」と呼ぶ.

3.2 応答ツイートの極性判定

発話対は発話とそれに対する応答から構成されるが、応答に相当するツイートの極性を判定する。極性の分類クラスは肯定、否定、中立の3つとする。本研究では oseti[5] を用いて極性を判定する。同ツールは、日本語評価極性辞書(用言編)[6](名詞編)[7] を使用し、文中の単語の極性スコアの平均によって文全体の極性スコア(-1 から 1 までの値)を算出する。応答ツイートの極性スコアに応じて対話コーパスを以下の3つのサブコーパスに分割する。

積極的性格対話コーパス 応答の $Score_{po}(t)$ が 0 より大きい発話対からなるコーパス。肯定的な単語を含むため、発話に対する応答は積極的な性格の持ち主によるものであると仮定する。

消極的性格対話コーパス 応答の $Score_{po}(t)$ が 0 より小さい発話対からなるコーパス。否定的な単語を含むため、発話に対する応答は消極的な性格の持ち主によるものであると仮定する。

一般対話コーパス 応答の $Score_{po}(t)$ が 0 である発話対からなるコーパス。その応答は、積極的、消極的いずれの性格も感じられないと仮定する。

以下、積極的性格対話コーパスと消極的性格対話コーパスをまとめて「性格対話コーパス」と呼ぶ。

3.3 対話モデルの学習

前項で述べた対話コーパスを訓練データとし、発話を入力としてそれに対する応答を出力する系列変換モデルを学習する。学習モデルとして Transformer[8] を利用する。

積極的あるいは消極的な性格を持つ自由対話システムを構築するためには、積極的性格対話コーパスあるいは消極的性格対話コーパスを訓練データとして対話モデルを学習するのが直接的な方法である。しかし、4 節で後述するように、Twitter 対話コーパスにおいて応答が肯定または否定の極性を持つ発話対の数は少なく、対話モデルを構築するために十分な訓練データ量を確保することは難しい。そこで、以下の二段階で対話モデルを学習する。

Step 1. 一般対話コーパスを用いて対話モデルを学習する。ここでの狙いは、比較的量の多い一般対話コーパスを用いて強い個性を持たない一般的な自由対話モデルを学習することにある。

Step 2. 前述のモデルを初期パラメタとして、性格対話コーパスを用いて Transformer を再学習する。ここでは積極的または消極的な性格を持つ対話システムを学習することを狙う。

以下、Step 1 で学習される対話モデルを「汎用対話モデル」、Step 2 で学習される対話モデルを「積極的性格対話モデル」または「消極的性格対話モデル」(2つをまとめて「性格対話モデル」と呼ぶ。

上記に加え、「汎用対話モデル」を学習する前に、JPersonaChat[9] を用いて Transformer を事前学習する。JPersonaChat は、あらかじめ 5 文程度で定義されたペルソナ(プロフィール)を作成し、クラウドワーカーにそのペルソナに沿ってチャットをするように依頼し、その対話を収録したコーパスである。JPersonaChat は一般対話コーパスに比べて規模は小さいが、Twitter から収集した発話対は対話として成立していないことがあるのに対し、JPersonaChat では自然な対話が収録されている。対話は 3 つ以上の発話から構成されるが、本研究では 2 つの発話の組に分割して使用する。

4 予備実験

本節では、提案手法を評価するための準備として、学習した対話モデルの大きな評価や、提案手法の問題点を明らかにするための予備実験について述べる。

4.1 実験手順

3 節で述べた手法にしたがい、積極的性格対話モデルと消極的性格対話モデルを学習する。

学習に用いた対話コーパスの発話対の数を表 1 に示す。応答ツイートの極性判定の結果、全体の対話コーパスのうち、およそ 2.8% が肯定的と判定され、5.7% が否定的と判定された。大部分のツイート(91.5%) は中立と判定されたため、性格対話コーパスの量は一般対話コーパスと比べてかなり少ない。また、消極的な性格を表すツイートは積極的な性格を表すツイートよりも多く、その比率はおよそ 2 倍であった。

表 1 に示した JPersonaChat 以外のそれぞれのサブコーパスをランダムに分割し、80% を訓練データ、14% を開発データ、6% をテストデータとした。訓練データから汎用対話モデル、積極的性格対話モデル、消極的性格対話モデルの 3 つを学習した。開発データは、本実験では、学習した対話モデルが入力

表1 対話コーパス

	発話対の数
一般対話コーパス	588,162
積極的対話コーパス	18,135
消極的対話コーパス	36,425
JPersonaChat[9]	56,793

文に対してある程度自然な応答を生成できることを確認するために用いた。次に、学習した対話モデルを用いて、テストデータにおける発話対のうち最初の発話を入力として与え、それに対する応答を生成した。

4.2 生成された応答の例

テストデータにおける発話に対して生成された応答の例を表2に示す。同じ発話を入力しても、積極的対話モデルでは積極的な個性が、消極的対話モデルでは消極的な個性が感じられる応答が生成された例が確認できた。提案手法による対話モデルの性格付けがある程度有効であることがわかった。

一方、不自然あるいは不適切な応答が生成される事例も多く見られた。2つの性格対話モデルの違いについては、「～しよう」といった何かを提案する発話に対しては応答に違いがあったが、それ以外の発話については応答に明確な差が見られないことが多かった。表2の例のように積極的または消極的な性格付けに成功している応答は限定的である。また、同じ文字や単語を繰り返すなど、そもそも応答として不自然な文が生成される事例も多く見られた。

5 今後の課題

本研究の今後の課題について述べる。現状では訓練データから Transformer を学習しているが、そのデータ量は必ずしも十分ではない可能性がある。そこで事前学習済みモデルの利用を検討する。具体的には、事前学習済み対話モデル [9] や BART 日本語 Pretrained モデル [10] を初期モデルとし、一般対話コーパスや性格対話コーパスを用いて再学習することを検討している。また、Transformer 学習時の損失関数に応答の極性を反映させる。積極的(消極的)性格対話モデルを学習する際には応答の極性が高い(低い)ほど小さい損失を与えることで、個性を持つ対話システムを学習する。さらに、現在は積極的対話モデルと消極的対話モデルを別々に学習

表2 生成された応答の例

U:	そこのお兄ちゃんポケモンしようや
R-gold:	1時までならいいぞい
R-pos:	いいよ
R-neg:	やりたいけどできない
R-neu:	そんなんでできるんかい ww
U:	沢山お話ししようね!!
R-gold:	今回って事前予約みたいなやつあるのかな...?
R-pos:	したい!!
R-neg:	緊張してきたんだけど、
R-neu:	ぜひぜひお願いします
U:	バイトしよ!!一緒に!!
R-gold:	したみみみみ
R-pos:	今から楽しみよ
R-neg:	まあお金はないし
R-neu:	まあ、言うてまだ夏休みやけど
U:	りえこー!野球しようぜー!
R-gold:	いつまでおるん?
R-pos:	とても楽しみですですね!!!
R-neg:	わたしとっても苦手なんですよ...
R-neu:	なにその名前!?
U:	今夜はタコパーです!さあ、準備しようっ!
R-gold:	めんどくさいやつ頑張ってください笑
R-pos:	今日もよろしくねー
R-neg:	頑張る・・・
R-neu:	いつになっちゃうんですか!?

Uは元の発話、R-goldはテストデータにおけるUに対する応答。R-pos、R-neg、R-neuはそれぞれ積極的対話モデル、消極的対話モデル、汎用対話モデルで生成されたUに対する応答。

しているが、極性を表す潜在変数を導入し、これによって対話モデルの応答の極性を制御することも検討したい。

次に、被験者による提案手法の評価実験を行い、提案手法の有効性を確認する。人と対話モデルによる会話を記録し、第三者にその会話を提示して、システムの応答の品質をいくつかの観点から評価してもらう。評価の観点として、文法の正確さ、内容の妥当性、個人性(システム応答は積極的もしくは消極的といった個性を感じるか)などを検討している。特に個人性について汎用対話モデルと性格対話モデルを比較し、対話モデルの性格付けが実現できているかを評価する。

性格対話コーパスの品質評価も行う。現在、osetiを用いて極性判定を行い、肯定的な応答や否定的な応答を抽出して性格対話コーパスを作成しているが、狙い通りに積極的・消極的な性格を反映した応答が集められているかは確認していない。したがって、性格対話コーパスの文をサンプリングして人手による評価を行い、必要に応じて性格対話コーパスの構築方法を改良する。

参考文献

- [1] Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In **Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics**, pp. 994–1003, 2016.
- [2] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics**, pp. 2204–2213, 2018.
- [3] Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. Long time no see! open-domain conversation with long-term persona memory. In **Findings of the Association for Computational Linguistics: ACL 2022**, pp. 2639–2650, 2022.
- [4] 関田崇宏, 白井清昭. マイクロブログからの対話コーパスの自動構築. 情報処理学会第 82 回全国大会 第 2 分冊, pp. 57–58, 2020.
- [5] oseti, (2023-1 閲覧). <https://github.com/ikegami-yukino/oseti>.
- [6] 小林のぞみ, 乾健太郎, 松本裕治, 立石健二, 福島俊一. 意見抽出のための評価表現の収集. 自然言語処理, Vol. 12, No. 3, pp. 203–222, 2005.
- [7] 東山昌彦, 乾健太郎, 松本裕治. 述語の選択選好性に着目した名詞評価極性の獲得. 言語処理学会第 14 回年次大会発表論文集, pp. 584–587, 2008.
- [8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Advances in Neural Information Processing Systems**, Vol. 30, 2017.
- [9] Hiroaki Sugiyama, Masahiro Mizukami, Tsunehiro Arimoto, Hiromi Narimatsu, Yuya Chiba, Hideharu Nakajima, and Toyomi Meguro. Empirical analysis of training strategies of transformer-based Japanese chat systems, 2021.
- [10] 田中佑, 村脇有吾, 河原大輔, 黒橋禎夫. 日本語 Wikipedia の編集履歴に基づく入力誤りデータセットと訂正システムの改良. 自然言語処理, Vol. 28, No. 4, pp. 995–1033, 2021.