Prefix Tuning とキャラクタ属性の加減算を利用したキャラクタ風発話生成

藤原 寛隆 [1]　新納 浩幸 [2]
[1] 茨城大学工学部情報工学科　[2] 茨城大学大学院理工学研究科情報科学領域
20t4066x@vc.ibaraki.ac.jp　hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

概要

特定のキャラクタの特徴を反映した発話を生成する技術は小説やアニメ，ゲームなどの応用において大いに需要がある．しかしながら既存手法の多くは変換規則を人手で構築する必要があり高コストである．また，対象キャラクタの言語モデルを構築する場合にも収集できる訓練データに限りがある．そこで，本研究では少ない訓練データから良質な言語モデルを構築する手法を提案する．具体的には，キャラクタの持つ属性情報をベクトル (prefix) として明示的に与える．その際，類似属性を持つ他キャラクタの発話から属性ベクトルを prefix-tuning の形式で学習させることで良質なベクトル表現を獲得する．また prefix 同士の加減算を利用することで属性同士の加減算を試みる．実験の結果，提案手法ではキャラクタの特徴を捉えた発話を生成することが難しいことがわかった．一方で一部の実験では発話に改善が見られ，属性情報を明示的に与えることや属性の加減算が有効に機能する可能性が示唆された．

1　はじめに

近年，大規模言語モデル（Large Language Model, 以下 LLM と略す）やその周辺技術の目覚ましい発展により言語モデルによるある程度自然な対話が実現された．また，API 等の整備により質問応答システムやゲームのキャラクタとの対話に LLM を組み込むなどの応用もみられるようになった．このような背景から，対話システムにキャラクタ性を持たせた発話を生成する技術には大いに需要がある．

キャラクタ性を反映した発話文を生成する研究は LLM 台頭以前よりいくつか行われてきたが，その多くが規則ベースの手法であり，手作業による規則の構築が高コストであるという問題があった．また，キャラクタの発話分を収集しその発話者の言語モデルを構築する手法が考えられるが，小説やアニメ，ゲームなどの発話者から収集できる発話文の量には限りがあるという問題がある．

そこで本論文では少量の発話文からキャラクタの特徴を有した発話文を生成する言語モデルの構築を試みる．一般に言語モデルの fine-tuning には十分な量の訓練データが必要であり，少量の発話文のみからキャラクタの特徴を捉えることは困難である．そのため本論文ではキャラクタの属性に注目し，似た属性を持つキャラクタの発話文を活用することを提案する．具体的には，属性をベクトルとしてして扱い，類似属性を持つキャラクタの発話から prefix-tuning の形式で属性ベクトルの表現学習を行う．その後学習した属性ベクトルを生成の際に入力文に付加することで対象キャラクタの特徴を捉えた発話の生成を目指す．また，属性ベクトル同士の加減算についても調査を行う．例えば，”ツンデレ”という属性は”ツンツン”と”デレデレ”のようにより小さな属性に分けて考えることができる．そこで，それらを個別にベクトルとして表現しベクトル同士の加算の形で表現することを考える．このようにすることで属性ベクトルの再利用性の向上や効率的な学習が期待される．

実験では日本語オープンコンテンツデータセットプロジェクトの Rosebleu ゲームシナリオをコーパスとし，属性を性別と Big Five (外向性，協調性，情緒安定性など) の高低の 22 属性とした．ベースラインは少量の発話文のみから prefix-tuning したものとし，提案手法で学習させた属性ベクトルを prefix としたモデルと比較した．実験の結果，提案手法ではキャラクタの特徴を捉えた発話を生成することが難しいことがわかった．一方で一部の実験では発話に改善が見られ，属性情報の明示的な付与や属性の加減算が言語モデルを利用したキャラクタ風発話の生成に有効に働く可能性が示唆された．

2 関連研究

2.1 キャラクタ風発話生成

話者の属性を利用した発話の特徴付けに関する研究は盛んに行われている．Mairesse と Walker は初めて高度にパラメータ化された会話生成器として PERSONAGE (personality generator) を提案した [1]．PERSONAGE は話者の性格として Big Five を採用しており，外向性の軸に沿って多様な発話が生成できることが示された．宮崎らは話者のキャラクタ性を特徴づける言語表現の基礎的分析として，性別，年齢，会話相手との親密度についてどのような語彙の省略や書き換え，挿入があるのかを調査した [2]．また各文の機能部をキャラクタの属性に基づき確率的に選択した書き換え規則に従って変換することによってキャラクタ性を変換する研究もある [3]．

キャラクタ風の発話生成には GPT などの言語モデルを使用したものも提案されている．岸野らは T5 を使用して同一作品の別キャラクタの発話を対象キャラクタ風の発話に変換することで対象キャラクタの発話文を増補する手法を提案した [4]．また増補した発話文を用いて DAPT を行った後，対象キャラクタの発話文で TAPT を行うことで発話者の特徴を有した発話文を生成する言語モデルを構築することができることを示した．

2.2 Prefix Tuning

Li らは fine-tuning の代替手法として prefix-tuning を提案した [5]．これは言語モデル自体の重みを更新せず，タスクごとに学習可能な小さなベクトル (prefix) をすべての層の入力系列の前に付加することで学習させる．実験ではベースとなる言語モデルの 0.1%のパラメータで fine-tuning と同等の性能が得られることが示された．また彼らはそれぞれの層の prefix を個別に学習させるのではなく，より小さなベクトル v を $dim(v)$ から $dim(h_i)$ に写す MLP で変換した方が学習が安定することを示した．ただし，h_i は各層の prefix をすべて結合したベクトルである．

本研究ではキャラクタ属性のベクトル表現に prefix を使用する．

3 提案手法

本研究の提案手法を図 1，図 2 に示す．本研究では prefix-tuning を用いて属性ベクトルを学習する．学習は属性ベクトルを一般的な属性の特徴を表現する共通属性ベクトルと各キャラクタ固有の特徴を表現するキャラクタ属性ベクトルに分けて行う．これにより，一人称や作品固有の用語などのキャラクタや作品固有の表現が共通属性ベクトルを利用する際に悪影響を及ぼす可能性を低減することが期待される．

全体の学習は共通属性の学習の後対象キャラクタ属性ベクトルの学習を学習させるという流れをとる．まず，大量の複数キャラクタの発話から共通属性ベクトルを学習させ，それを固定した状態で対象キャラクタの少量の発話からキャラクタ属性ベクトルを学習させる．

3.1 共通属性ベクトルの学習

共通属性は加減算可能であることが求められる．そこで，word2vec の skip-gram [6] のように同時に出現する属性に対して内積が小さくなるように損失関数を修正する．具体的には，１発話について発話者の共通属性の中から１つランダムに選び出し，選ばれなかった他の属性との内積の総和を損失関数に加える．これをバッチ内のすべての発話について行う．従って，最小化する目的関数は言語モデル自体の損失と上記のベクトル同士の内積の総和の和となる．

共通属性ベクトルの学習では，共通属性ベクトル，属性空間から prefix 空間に写す MLP（Prefix Generator），prefix を各層に写す MLP と各キャラクタのキャラクタ属性ベクトルを学習させる[1)]．ここで学習されるキャラクタ属性ベクトルは対象キャラクタの発話生成には使用されないが，キャラクタや作品固有の情報が共通属性ベクトルに入り込まないように学習させる．

3.2 対象キャラクタベクトルの学習

対象キャラクタベクトルの学習では，対象キャラクタのキャラクタ属性ベクトルと prefix を各レイヤーに写す MLP のみを学習させる．ここで学習し

1) 実験では，各属性ベクトルのサイズを 1024，prefix を各層に写す MLP の hidden size は 512 とした．また，Prefix Generator は属性ベクトルを 1024 → 512 → 1024 と変換する 3 層の MLP とし，活性化関数を tanh とした．

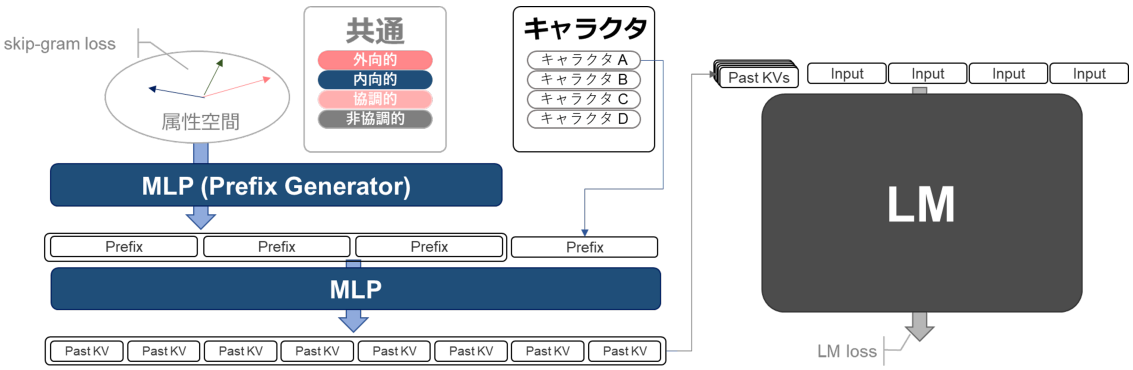

図 1　共通属性ベクトルの学習

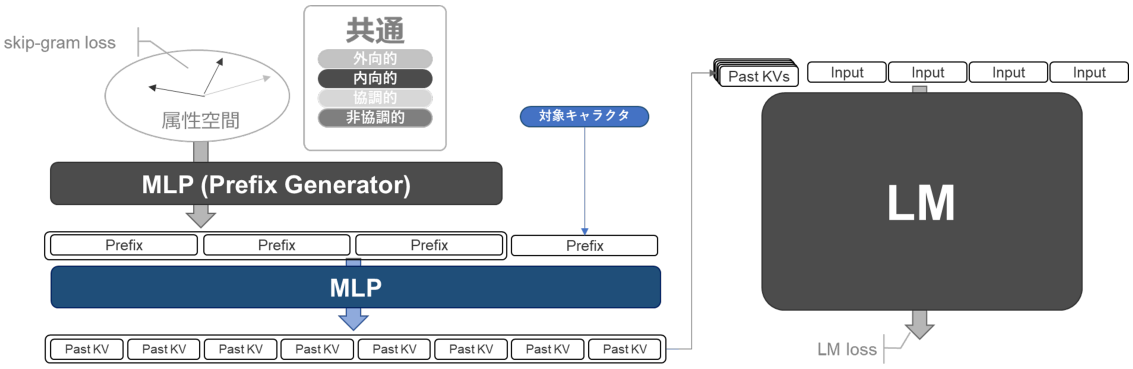

図 2　キャラクタ属性ベクトルの学習

た対象キャラクタベクトルを生成時に利用する．

4　実験

4.1　言語モデル

ベースの原語モデルには rinna 社が公開している japanese-gpt2-medium を使用した[2)]．これは Hugging Face 社の Transformers ライブラリから，モデル名 ”rinna/japanese-gpt2-medium” で利用することができる．

4.2　入力・出力

入力文は直前の対話 1，2 文について”発話者：発話文 \n”を結合し，さらに対象キャラクタについて”[SEP] 発話者：”を結合した文となる．入力は入力文と対象キャラクタの持つ共通属性 prefix とキャラクタ属性 prefix を結合したものとなる．出力は直前の対話に続く対象キャラクタの発話文である．

4.3　データ

日本語オープンコンテンツデータセットプロジェクトの Rosebleu ゲームシナリオ[3)]の一部から 40 キャラクタについて入力文を収集した．共通の属性については Big Five の視点から各項目について高低のラベルと男女のラベル計 22 種についてラベル付けを行った．ラベル付けは著者一名により行われた．

4.4　評価指標

キャラクタ性を反映した生成文であるかどうかは人手により評価を行った．評価者は著者一名であり，表 1 の評価基準に基づき生成文 100 件に対して評価を行った．

表 1　評価基準

評価	説明
◎	○ を満たし，参照応答に極めて近い
○	△ を満たし，キャラクタらしい発話である
△	対話として成立している（キャラクタらしさは問わない）
×	会話が破綻している

4.5　実験結果

prefix-tuning と提案手法でキャラクタ”空”の言語モデルを構築し，それぞれについて生成文 100 件に対して評価を行った．prefix-tuning ではプレフィ

2）https://huggingface.co/rinna/japanese-gpt2-medium
3）https://gitlab.com/open_contents_datasets/Rosebleu

表 2　生成文 100 件に対する人手評価

	t=50		t=100	
	Good(◎, ○)	Natural(◎, ○, △)	Good(◎, ○)	Natural(◎, ○, △)
PREFIX-TUNE	**23**	25	**23**	25
Ours(r = 0.25)	20	**28**	20	**29**
Ours(r = 0.5)	17	23	15	21

表 3　生成文 100 件に対する人手評価（加算，損失の修正の有効性検証）

		t=50		t=100	
		Good(◎, ○)	Natural(◎, ○, △)	Good(◎, ○)	Natural(◎, ○, △)
r=0.25	Ours	20	28	**20**	**29**
r=0.25	$\text{Ours}_{no-same}$	**33**	**40**	16	27
r=0.25	$\text{Ours}_{no-same-loss}$	16	25	**20**	28
r=0.5	Ours	17	23	15	21
r=0.5	$\text{Ours}_{no-same}$	**20**	**26**	**26**	**35**
r=0.5	$\text{Ours}_{no-same-loss}$	17	25	17	28

クス長 $t = 50, 100$ としてそれぞれ”空”の全発話のうち 100 件について 40epoch 学習させた．提案手法についてはプレフィックス長 $t = 50, 100$，prefix 全体に占めるキャラクタ属性の割合を $r = 0.25, 0.5$ とした．また，共通属性ベクトルの学習には”空”以外の 39 キャラクタのすべての対話について 20epoch，キャラクタベクトルの学習に”空”の全発話のうち 100 件について 20epoch，計 40epoch 学習させた．prefix-tuning を PREFIX-TUNE，提案手法を Ours として評価結果を表 2 に示す．

人手評価の結果から，少量の発話から prefix-tuning を行った方が提案手法に比べキャラクタらしい発話を生成できることがわかった．一方で人手評価の Natural のスコアが提案手法の方が高いことから，キャラクタ性を問わない文章の自然さという点では提案手法の方が優れているとわかった．

5　考察

5.1　属性の加算の有効性

実験では，同じ属性の組を持つ発話文が共通属性の訓練データに含まれており，真に属性の加算が有効に機能しているかどうかの検証が不十分である．そこで，共通属性の訓練データから対象キャラクタの”空”と同じ属性の組を持つキャラクタの発話文を除外し再度生成文 100 件に対して評価を行った．結果を $\text{Ours}_{no-same}$ として表 3 に示す．

評価の結果 $t = 100, r = 0.25$ を除くすべての場合で Good, Natural ともにスコアが悪化しなかった．このことから，属性の加算は有効に機能しているものと推察される．各スコア改善した原因については，類似属性を持つキャラクタの発話文を訓練データから抜いたことでデータ数のばらつきが緩和されたことが考えられる．$t = 100, r = 0.25$ で悪化した原因についてはより詳細な調査が必要である．

5.2　損失修正の有効性

さらに属性の加算において共通属性ベクトル同士の内積を損失として加算することが有効に機能しているかどうかを検証するため，損失を修正したものとそうでないものについて評価を行った．結果を $\text{Ours}_{no-same-loss}$ として表 3 に示す．

評価の結果 $t = 100, r = 0.25$ を除くすべての場合で Good, Natural ともに $\text{Ours}_{no-same}$ よりも低いスコアとなった．このことから，損失の修正が有効に機能していることがわかる．また，損失を修正することによってデータ数のばらつきによる悪影響が緩和された可能性も考えられる．

以上を踏まえ，訓練データにおけるデータ数のばらつきを含むより詳細な検証を今後の課題とする．

6　おわりに

本研究ではキャラクタの属性をベクトルで表現し明示的に与えることで少量の発話文から特徴を捉えた発話を生成できるかを調査した．実験の結果，提案手法ではキャラクタの特徴を捉えた発話を生成することが難しいことがわかった．

また属性同士の加算についても調査を行った．その結果一部設定でベースラインを上回る結果となり，キャラクタの属性情報の明示的な付与や加減算が有効に働く可能性が示唆された．

7 謝辞

本研究は国立国語研究所の共同研究プロジェクト「テキスト読み上げのための読みの曖昧性の分類と読み推定タスクのデータセットの構築」及び JSPS 科研費 23K11212 の助成を受けています.

参考文献

[1] François Mairesse and Marilyn Walker. PERSONAGE: Personality generation for dialogue. In Annie Zaenen and Antal van den Bosch, editors, **Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics**, pp. 496–503, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[2] 宮崎千明, 平野徹, 東中竜一郎, 牧野俊朗, 松尾義博, 佐藤理史. 話者のキャラクタ性に寄与する言語表現の基礎的分析. 言語処理学会 第 20 回年次大会 発表論文集, pp. 232–235, 2014.

[3] 宮崎千明, 平野徹, 東中竜一郎, 牧野俊朗, 松尾義博, 佐藤理史. 文節機能部の確率的書き換えによる言語表現のキャラクタ性変換. 人工知能学会論文誌, Vol. 13, No. 1, pp. DSF–515, 2016.

[4] 望叶岸野, 嘉那子古宮, 浩幸新納. T5 による特定キャラクター風発話への変換とその言語モデルの構築. Technical Report 13, 茨城大学大学院理工学研究科情報工学専攻, 東京農工大学大学院工学研究院先端情報科学部門, 茨城大学大学院理工学研究科情報科学領域, 2022.

[5] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation, 2021.

[6] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.