漫画話者認識における VLM の有効性

呂 博軒 [1,2]　能地 宏 [2]
[1] 東京科学大学　[2]Mantra 株式会社
lyu@lr.pi.titech.ac.jp　noji@mantra.co.jp

概要

本研究では，漫画におけるゼロショット話者認識という課題に取り組む．既存研究では，大規模言語モデルとコンピュータビジョンモデルを組み合わせた手法が提案されているが，複雑なパイプラインに依存し，必ずしも精度向上に貢献しないという課題がある．そこで本研究では，単一の Vision and Language Model を活用した新たなゼロショット漫画話者認識手法を提案する．実験の結果，提案手法が既存手法を上回る精度を達成することを示す．興味深いことに，既存手法と提案手法における視覚情報の寄与を詳細に分析した結果，視覚情報が認識精度に及ぼす影響が限定的であることを明らかにする．

1　はじめに

漫画における話者認識とは，漫画内の台詞がどのキャラクターによって発話されているかを特定するタスクである．キャラクターは漫画のストーリーを構成する中核的要素であるため，このタスクは漫画の機械翻訳をはじめとする漫画関連タスクの基盤となる [1]．既存の漫画機械翻訳研究において，話者情報が翻訳の質を向上させる重要な要素であることが実証されている [1]．漫画機械翻訳において話者認識の重要性を示す具体例として，図 1 における「返した！」という台詞の英語への翻訳を考える．この台詞を適切に翻訳するためには，話者の特定が不可欠である．このような事例は，画像の情報に基づき，台詞の話者を考慮しなければ解決できない問題である． 本研究では，ゼロショット漫画話者認識に取り組む．これは，実環境において漫画のアノテーション付きデータの入手が困難であることに起因する．例えば，新刊漫画に対して即時的な話者認識が要求される場合などが想定される．なお本研究では，Li ら [2] の設定に準拠し，画像内のテキストが事前に正確に抽出されている状況下での話者認識に焦点を当てる．この前提により，テキスト抽出時のエラーに左右されることなく，漫画話者認識の本質的な課題に注力することが可能となる．

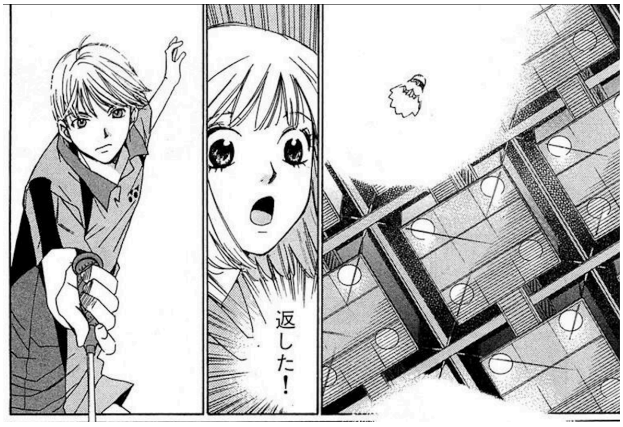

図 1　漫画機械翻訳における話者情報の有用性：台詞「返した！」の英訳が，話者が主人公である場合は *“I gave the ball back!”*，主人公の妹である場合は *“He gave the ball back!”* となる．“やまとの羽根”© 咲 香里

上述の定義のゼロショット漫画話者認識に関して，我々の調査範囲では，Li ら [2] の研究が唯一の公開された先行研究である．彼らはタスクの定義を確立し，大規模言語モデル (Large Language Model, LLM) とコンピュータビジョンモデルを組み合わせた認識手法を提案した．我々は彼らのタスク定義を踏襲しつつも，その解決手法には再考の余地があると考える．彼らの方法は繰り返しを含む複雑なパイプラインに依存しており，我々の調査によれば，このパイプラインの一部は話者認識の精度向上にほとんど寄与せず，場合によっては性能を低下させる要因となることが判明したためである．

本研究の発見は以下の通りである：

- 単一の Vision and Language Model (VLM) を活用した漫画話者認識手法を新たに提案し，既存法を上回る精度を達成した．
- 既存手法および提案法において，視覚情報が認識精度に及ぼす影響を詳細に分析した結果，予想に反して，視覚情報の寄与が限定的であることを明らかにした．

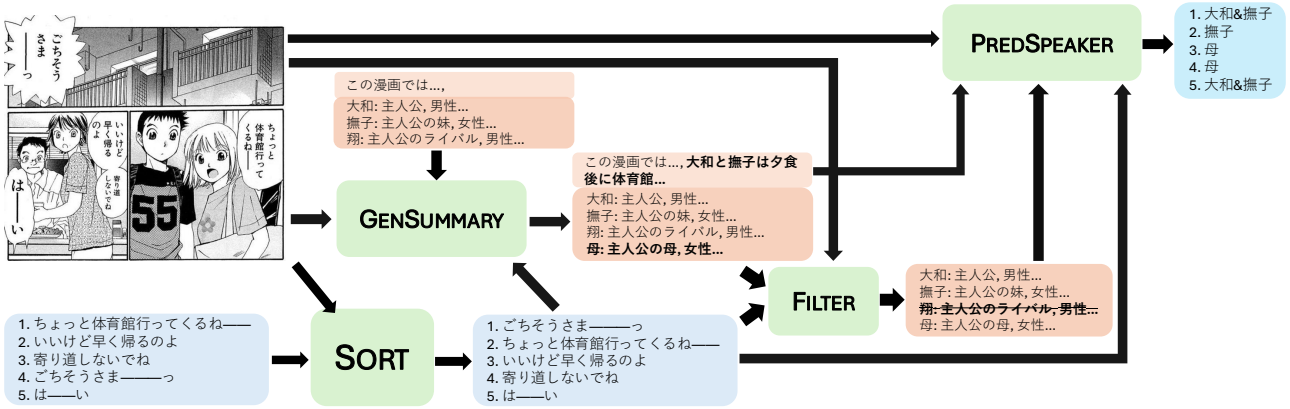

図 2　提案手法におけるチャンク内の処理の具体例．“やまとの羽根”© 咲 香里

2　関連研究

我々の知る限りでは，ゼロショット漫画話者認識タスク直接関連する先行研究は，Li ら [2] の研究のみである．彼らの手法は二段階のプロセスで構成されている．第一段階では，LLM を用いて台詞テキストのみに基づく話者認識を行う．第二段階では，コンピュータビジョンモデルにより，キャラクターの身体領域と吹き出しの位置関係という視覚的特徴を考慮して，第一段階での認識結果を修正している．

次に，非ゼロショット設定における関連研究について述べる．初期の研究では，キャラクターの身体領域と吹き出しの視覚的位置関係を用いるなど，ルールベースの手法が主流であった [3]．近年では深層学習を活用した手法も提案されており，例えば中川ら [4] は，キャラクターの身体領域と吹き出しの位置関係を考慮しつつ，回帰型ニューラルネットワーク [5] を用いてテキストの意味的特徴を取り入れる手法を提案している．また，Li ら [6] は，キャラクターの身体領域と吹き出しの対応関係をより正確に予測するため，Scene Graph Generation モデルの使用を提案している．

3　漫画話者認識

3.1　問題設定

一冊の漫画は N ページから構成される．n 番目のページの画像を I^n，台詞テキスト列を $T^n = [t_0^n, t_1^n, \ldots, t_{M-1}^n]$，テキスト列に対応する話者ラベルの列を $S^n = [s_0^n, s_1^n, \ldots, s_{M-1}^n]$ と表す．ここで，$I = [I^0, I^1, \ldots, I^{N-1}]$ は一冊全体の画像列を，$T = [T^0, T^1, \ldots, T^{N-1}]$ は台詞テキスト列を，$S = [S^0, S^1, \ldots, S^{N-1}]$ は話者ラベル列を表す．また，任意の $0 \le i \le M-1$ に対して，s_i^n は t_i^n に対応する話者ラベルを表すものとする．

ゼロショット漫画話者認識とは，S に含まれるいかなる要素にもアクセスすることなく，I と T のみに基づいて，各台詞に対応する話者ラベルを予測することを指す．この設定は，新刊漫画を処理する際に，視覚情報と台詞テキスト情報のみを用いて各台詞の話者を特定する必要がある現実の課題を模擬したものである．

3.2　提案法

提案法のパイプラインは，**台詞順序整理**，**概要とキャラクタープロフィール生成**，**話者絞り込み**，**話者認識**という 4 つのモジュールで構成される．これらのモジュールはすべて，同一の VLM へのプロンプトによって実現される．

本手法の特徴は，漫画を複数のチャンクに分割し，各チャンクを順次処理する点にある．形式的には，一冊の漫画を K 個のチャンクに分割し，各チャンク k $(0 \le k < K)$ は C ページから構成されるものとする：$I_k = [I^{kC}, I^{kC+1}, \ldots, I^{(k+1)C-1}]$，$T_k = [T^{kC}, T^{kC+1}, \ldots, T^{(k+1)C-1}]$，及び $S_k = [S^{kC}, S^{kC+1}, \ldots, S^{(k+1)C-1}]$ である．ただし，最後のチャンク $k = K-1$ については，残りのページ数が C より少ない場合がある．

提案法の流れを擬似コード 1 に示す．また，チャンク内の処理の具体例を図 2 に示す．

台詞順序整理　はじめに，漫画全体の台詞テキスト T について，ページごとに整理を行う．これは，使用したデータセットの台詞順序が実際の発話順序と一致していない可能性があるためである．第 n ページの画像 I^n と台詞テキスト T^n に対して，台詞順序整理器は正しい順序の台詞テキスト

Algorithm 1 提案法の疑似コード

Require:

一冊全体の画像列 I, 一冊全体の台詞テキスト列 T, チャンク数 K, ページ数 N

Ensure:

各チャンクの台詞話者ラベル S_k

$T' \leftarrow []$

1: **for** $n = 1$ to N **do**

$T'^n \leftarrow \textsc{Sort}(I^n, T^n)$ ▷ 台詞順序整理

$T' \leftarrow T' + [T'^n]$

2: **end for**

3: **for** $k = 1$ to K **do**

$(SM_k, CP_k) \leftarrow \textsc{GenSummary}(I_k, T'_k, SM_{k-1}, CP_{k-1})$ ▷ 概要とキャラクタープロフィール生成

$CP'_k \leftarrow \textsc{Filter}(I_k, T'_k, CP_k)$ ▷ 話者絞り込み

$S_k \leftarrow \textsc{PredSpeaker}(I_k, T'_k, SM_k, CP'_k)$ ▷ 話者認識

4: **end for**

T'^n $(0 \le n \le N-1)$ を予測することを目的とする：

$$\textsc{Sort}(I^n, T^n) \rightarrow T'^n. \tag{1}$$

この処理の後，チャンクごとに概要とキャラクタープロフィール生成，及び話者認識を行う．

概要とキャラクタープロフィール生成　このモジュールは，話者認識に必要な文脈情報を提供することを目的とする．形式的には，概要とキャラクタープロフィール生成器は現在のチャンクの画像 I_k，現在のチャンクの修正した順序の台詞 T'_k，現在の累積概要 SM_{k-1}，及び現在の累積キャラクタープロフィール CP_{k-1} を入力とし，現在のチャンクの画像とテキストを考慮した概要とキャラクタープロフィールを生成する：

$$\textsc{GenSummary}(I_k, T'_k, SM_{k-1}, CP_{k-1}) \rightarrow (SM_k, CP_k), \tag{2}$$

ここで，SM_k は現在のチャンクの内容を含む更新後のストーリー概要，CP_k は更新後のキャラクタープロフィールを表す．なお，$k = 0$ の場合，SM_{k-1} と CP_{k-1} は空文字列となる．

話者絞り込み　話者絞り込みモジュールの目的は，各チャンク内の登場キャラクターを特定し，後の話者認識の分類数を減らすためである．形式的には，第 k チャンクにおける話者絞り込み器は I_k，T'_k，及び CP_k を入力とし，チャンク内に登場したキャラクターのプロフィルー CP'_k を出力する：

$$\textsc{Filter}(I_k, T'_k, CP_k) \rightarrow CP'_k. \tag{3}$$

話者認識　各チャンクの概要とキャラクタープロフィール，及び各チャンク内の登場キャラクターを取得後，話者認識モジュールはこれらの情報に基づいて，現在のチャンク内の各台詞の話者を認識する．具体的には，第 k チャンク内の全ての台詞 T'_k に対して，話者認識器は I_k，T'_k，SM_k，及び CP'_k を入力し，第 k チャンク内の話者ラベル S_k を認識する：

$$\textsc{PredSpeaker}(I_k, T'_k, SM_k, CP'_k) \rightarrow S_k. \tag{4}$$

このようなチャンクごとの処理パイプラインを設計した主な理由は，VLM が限られた context window で局所的（現在のチャンクの画像と台詞）および大域的（累積概要，累積キャラクタープロフィール）な情報を同時に利用して話者を認識できるようにしつつ，長編漫画の処理を可能とするためである．

また，上記で紹介した提案法に加えて，全てのモジュールに画像を除外した手法 (以降，「**提案法 w/o images**」と呼ぶ) も研究対象とした．

4　実験

4.1　実験設定

評価データセットとして `Manga109`[6] を使用した．会話の話者を識別するために必要な文脈情報が，前巻の欠落により得られない問題を避けるため，同データセットから 10 作品の第 1 巻を選定した．評価指標には，台詞単位での正解率を採用した．

LLM には OpenAI の GPT-4o (`gpt4o-2024-08-06`)[7] を用い，NVIDIA RTX A6000 を 8 枚搭載した計算機環境にて Li ら [2] の手法を再実行し，比較対象とした．本実験において，Li ら [2] の手法については，彼らが論文中で示した原設定に従い，最大修正反復回数（$iter$）を 2 に設定した．

提案法における VLM には GPT-4o (`gpt4o-2024-08-06`) を適用した．チャンクサイズ (C) を $[2, 5, 8, 12]$ の範囲で変化させ，その影響を評価した．チャンクサイズの最大値を 12 とした理由は，これを超える値を設定すると，`gpt4o-2024-08-06` の最大系列長を超えるトークン数となることが事前に検証されたためである．

その他の詳細は付録 A に記述されている．

4.2　実験結果

十冊の漫画を用いて平均正解率を調査した．評価対象は「All」(全話者の台詞) と「$filtered$ (α)」(台

手法/評価対象	Li ら [2] (*iter* = 0)	Li ら [2] (*iter* = 1)	Li ら [2] (*iter* = 2)	提案法 (C = 12)	提案法 (C = 12) w/o images
All	0.5497	0.5678	0.4929	0.6366	**0.6640**
filtered (α = 50)	0.6423	0.6618	0.6736	0.6938	**0.7200**
filtered (α = 100)	0.6803	0.7063	0.7253	0.7373	**0.7474**
filtered (α = 150)	0.6980	0.7507	**0.7810**	0.7701	0.7784

表 1　十冊の漫画からなる評価データセットにおける正解率：「*All*」は全話者の台詞を評価対象としたものであり，「*filtered* (α)」は台詞数が α 回以上出現する話者の台詞を評価対象としたものである．正解率は台詞レベルで計算されている．Li ら [2] の手法における「*iter*」は，コンピュータビジョンモデルによって LLM の認識結果を修正する際の反復回数を表している．本研究で提案する手法における「C」は，1 つのチャンクに含まれるページ数を示している．

詞数が α 以上の話者の台詞）の 2 つのカテゴリに分類した．結果を表 1 に示す．また，タイトルごとの正解率を付録 B に示す．

「*All*」の欄の結果から，提案法 (C = 12) 及び提案法 (C = 12) w/o images の正解率は *iter* の値によらず，Li ら [2] の手法を上回った．したがって，提案法は Li ら [2] の手法よりも優れていることが示された．

さらに，Li ら [2] の手法では，*iter* を増やすことが必ずしも全体（*All*）の精度向上につながらないことが明らかになった．一方で，台詞数の少ないキャラクターの台詞を除外した場合（*filtered*），*iter* の増加が精度向上に寄与することが確認された．これらの結果は，Li ら [2] が用いたコンピュータビジョンモデルによる LLM の認識結果の修正というプロセスが，台詞数の多いキャラクターの台詞に対してのみ効果的である可能性を示唆している．

5　考察

5.1　チャンクサイズの影響

手法	提案法	提案法 w/o images
C = 2	0.5710	0.6155
C = 5	**0.6494**	0.6477
C = 8	0.6491	0.6469
C = 12	0.6366	**0.6640**

表 2　チャンクサイズが提案法の正解率に与える影響：全キャラクターのセリフ（*All*）を評価対象とする．数値は 10 冊の漫画の平均正解率である．

C を大きくすると，VLM は一度により多くの非圧縮の文脈情報（I_k と T'_k）を参照できるが，VLM は長い文脈の処理が不得意なため [8]，トークン数が多すぎると話者認識に重要な情報の理解が困難になる可能性がある．そこで，チャンクサイズと話者認識の精度の相関性を実験的に明らかにする．

表 2 に示すように，提案法では C = 5 という中程度のチャンクサイズで最高の正解率が得られた．これは，C がより大きい場合，入力のトークン数が過多となり，性能が低下したためと考えられる．一方，画像を使用しない提案法では，C = 12 で最高の正解率を達成した．これは，画像入力が多くのトークンを占めるため，それらを省くことで VLM が処理能力の範囲内でより多くのページを理解できたためと考えられる．

5.2　画像情報の影響

表 2 の結果から，同じチャンクサイズでも画像を使用しない提案法が画像使用時より高い正解率を示す場合があり，特に C = 12 の「提案法 w/o images」が最高正解率を達成したことが分かる．これは，VLM が画像情報にアクセスでき，それを話者の特定に活用するよう指示されているにもかかわらず，実際には画像情報を十分に活用できていないことを示唆している．「提案法」の出力を手動で確認したところ，エラーの大部分は画像参照により容易に解決可能なものであった．例えば，そのページに登場していないキャラクターの台詞として誤って認識するケースなどである．この発見は，ゼロショット漫画話者認識において VLM に画像情報を効果的に活用させることが課題であることを示している．

6　おわりに

本稿では，単一の VLM を活用した新たなゼロショット漫画話者認識手法を提案した．実験結果から，提案手法は既存手法と比較してより高い認識精度を達成することを確認した．また，興味深いことに，視覚情報が認識精度に及ぼす影響は限定的であることが明らかになった．

今後の課題として，視覚情報をより効果的に活用できるような手法の検討が挙げられる．例えば，吹き出しとキャラクター身体領域との関連性をより深く考慮する機構を導入することなどが考えられる．

参考文献

[1] Ryota Hinami, Shonosuke Ishiwatari, Kazuhiko Yasuda, and Yusuke Matsui. Towards fully automated manga translation. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 14, pp. 12998–13008, May 2021.

[2] Yingxuan Li, Ryota Hinami, Kiyoharu Aizawa, and Yusuke Matsui. Zero-shot character identification and speaker prediction in comics via iterative multimodal fusion. In Proceedings of the 32nd ACM International Conference on Multimedia, MM '24, p. 7366–7374, New York, NY, USA, 2024. Association for Computing Machinery.

[3] Christophe Rigaud, Nam Le Thanh, J.-C. Burie, J.-M. Ogier, Motoi Iwata, Eiki Imazu, and Koichi Kise. Speech balloon and speaker association for comics and manga understanding. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 351–355, 2015.

[4] 諒中川, 猛梅澤, 範高大澤. 漫画におけるセリフ位置と意味の時系列を考慮した話者キャラクターの推定. マルチメディア, 分散協調とモバイルシンポジウム 2019 論文集, Vol. 2019, pp. 1291–1297, 06 2019.

[5] S Hochreiter. Long short-term memory. Neural Computation MIT-Press, 1997.

[6] Yingxuan Li, Kiyoharu Aizawa, and Yusuke Matsui. Manga109dialog: A large-scale dialogue dataset for comics speaker detection. In 2024 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6, 2024.

[7] OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, et al. Gpt-4o system card, 2024.

[8] Junqi Ge, Ziyi Chen, Jintao Lin, Jinguo Zhu, Xihui Liu, Jifeng Dai, and Xizhou Zhu. V2pe: Improving multimodal long-context capability of vision-language models with variable visual position encoding, 2024.

A　実験設定詳細

提案手法では，話者認識の完了後，VLM を用いて予測された話者ラベルの表層形を正解ラベルの表層形に対応付ける．例えば，予測された「大和」を正解ラベルの「鳥羽大和」へと対応付ける処理を行う．この対応付けの際，VLM への入力として正解ラベルの表層形，予測ラベルの表層形，および T を与える．なお，このようなラベル表層形の利用は Li ら [2] の問題設定において認められており，彼らの手法でも正解ラベルの表層形が入力として最初から与えられている．

GPT-4o を用いる全ての実験では，温度パラメータを 0 に設定した．

B　タイトルごとの正解率

タイトルごとの正解率を表 3 と 4 に示す．なお，一部の作品タイトルについては略称を使用している：「さまよえる少年」は「さまよえる少年に純愛を」，「ライジングガール」は「ライジングガール！〜人見絹枝物語〜」，「うるとら」は「うるとら☆イレブン」，「しまっていこうぜ」は「しまっていこうぜ！第一巻」，「タップ君」は「タップ君の探偵室」である．

漫画タイトル	$C=2$	$C=5$	$C=8$	$C=12$
最速!!	0.5553	0.6194	0.6789	0.6112
大和の羽根	0.7335	0.7724	0.7638	0.7627
さまよえる少年	0.4467	0.7173	0.6720	0.6597
ライジングガール	0.6648	0.6028	0.6366	0.5950
うるとら	0.3844	0.5317	0.4868	0.4612
しまっていこうぜ	0.5320	0.5913	0.6378	0.5296
やさしい悪魔	0.5498	0.7765	0.8329	0.8257
その気でＡＢＣ	0.6822	0.7092	0.6858	0.7101
太陽にスマッシュ！	0.7325	0.7199	0.7630	0.7668
タップ君	0.4283	0.4536	0.3333	0.4442
平均	0.5710	0.6494	0.6491	0.6366

表 3　**提案法**のタイトルごとの正解率：評価対象は全話者の台詞である．本研究で提案する手法における「C」は，1 つのチャンクに含まれるページ数を示している．

漫画タイトル	$C=2$	$C=5$	$C=8$	$C=12$
最速!!	0.5663	0.5892	0.6240	0.6661
大和の羽根	0.6957	0.7584	0.7702	0.7626
さまよえる少年	0.6217	0.6867	0.6891	0.6585
ライジングガール	0.6467	0.6162	0.6248	0.6484
うるとら	0.3519	0.4357	0.4543	0.4744
しまっていこうぜ	0.6064	0.6276	0.6463	0.6650
やさしい悪魔	0.8125	0.8381	0.8191	0.8374
その気でＡＢＣ	0.6741	0.6993	0.6885	0.7280
太陽にスマッシュ！	0.7135	0.7237	0.7123	0.7630
タップ君	0.4655	0.5020	0.4402	0.4362
平均	0.6155	0.6477	0.6469	0.6640

表 4　**提案法 w/o images** のタイトルごとの正解率．本研究で提案する手法における「C」は，1 つのチャンクに含まれるページ数を示している．