

対話単位の判定と後続文脈の付与を用いた 大規模言語モデルによるチャット対話分離の提案

高田尚輝 Yin Minn Pa Pa 吉岡克成 森辰則
横浜国立大学

takada-naoki-xs@ynu.jp, {yinminn-papa-jp, yoshioka, tmori}@ynu.ac.jp

概要

多人数参加型オンラインチャットにおける対話の交錯は、文脈理解を著しく阻害する。この問題に対し、発話系列を応答関係のある対話クラスタへ分割する対話分離が提案された。本タスクの方法論は長らく教師あり学習が支配的であり、唯一行われた大規模言語モデル (LLM) ベースの手法の予備研究では、LLM が従来手法に劣ると報告された。本稿では、LLM ベースの手法を再評価するため、対話単位の割り当て (DLA) および後続文脈の利用 (SC) を提案する。ベンチマークを用いた評価実験の結果、DLA と SC を統合した手法は全評価指標において最高精度を達成した。アプリケーション研究では、モデルによる依存性は認められるものの、両手法が性能向上に寄与する要素であることが実証された。本研究は、対話分離において、教師あり学習による手法から LLM ベースの手法への転換を示すものである。

1 はじめに

多数の参加者によって構成される Slack や Discord 等のオンラインチャットにおいて、複数の対話が同時に進行し交錯する現象が生じる。この交錯はユーザがチャット中の文脈を追跡する難易度を上げる。システムによる解析においても、文脈にノイズが混入するため処理が困難となる。この課題に対し、対話分離 (Dialogue Disentanglement) という手法が提案された [1, 2]。対話分離は図 1 に示すように、対話が交錯している発話系列を応答関係で結びついた一貫性のある対話クラスタへ分割するタスクである。本技術は、対話状態追跡 [3, 4] や応答生成 [5, 6, 7, 8] といった下流タスクの重要な基盤となる。

従来、このタスクは教師あり機械学習による発話間ペアワイズ分類として扱われてきた [2, 9, 10]。一方で、大規模言語モデル (LLM) は高度な文脈理解



図 1 対話分離後の対話を色ごとに示す。

能力と推論力を有しており、対話分離への応用が期待される。しかし、LLM を本タスクへ適用した唯一の試みでは、従来の非 LLM 手法と比較して精度が大幅に劣ることが報告された [11]。この性能差は、LLM に対して従来のペアワイズ分類と同様の「発話単位の割り当て」を強いたことに起因すると考えられる。

そこで本研究では、高精度な対話分離を LLM ベースで行うため、新たに 2 つの手法を導入する。第 1 は、個別の親発話ではなく、既知の対話クラスタへの帰属を判定させる対話単位の割り当て (DLA: Dialogue-Level Assignment) である。第 2 は、判定対象の発話に続く後続文脈を補助的根拠として参照させる手法 (SC: Subsequent Context) である。本研究の貢献は以下の通りである。第一に、DLA および SC を提案し、これらを用いた LLM ベースの対話分離が既存の最高精度を凌駕することを示す。第二に、アプリケーション研究を通じ、DLA と SC が精度を向上させる要素であることを実証する。同時に、

その有効性がモデルやプロンプト構成に大きく依存することも明らかにする。

2 提案手法

2.1 対話分離のタスク定義

チャットデータ中の全発話系列を会話 C と呼び、 C から分離された対話の集合を D とする。ここで、各 $d_j \in D$ は、特定の話題や目的を共有し、応答関係で結ばれた一連の発話からなる対話クラスタである。これに基づき、タスクは次のように定式化される。入力は時系列順に並んだ n 個の発話からなる会話 $C = (u_1, u_2, \dots, u_n)$ である。各発話 $u_i \in C$ は、タプル $u_i = (t_i, s_i, m_i)$ で表される。ここで t_i は発話時刻、 s_i は話者、 m_i はテキストメッセージである。目的は、 C を互いに素な対話集合 $D = \{d_1, d_2, \dots, d_p\}$ に分割することである。この分割は、次の2つの条件を満たす、 C の厳密な分割でなければならない。第1は網羅性であり、 $C = \bigcup_{d_j \in D} d_j$ が成立することである。第2は排他性であり、 $\forall j \neq k, d_j \cap d_k = \emptyset$ が成立することである。

2.2 LLM による対話分離手法:DLA, SC

本手法の全体像を図2に示す。会話の先頭から末尾へ向けて順番に、判定の対象となる「対象発話」を1件選択し、応答関係の有無を判定する。本稿では、対話単位の割り当て (DLA: Dialogue-Level Assignment) と後続文脈の利用 (SC: Subsequent Context) という2つの手法を提案する。まず、DLAについて述べる。DiHRL[11]の予備実験では、従来の発話間ペアワイズ分類をもとにLLMによる対話分離の手法が考案された。この手法は応答関係を2つの発話間のみ関係として扱うものである。LLMは先行発話群と各発話のID番号を受け取り、対象発話がどの「発話」と同じ対話であるか、そのIDを出力するように指示される。新しい対話の始まりである場合には対象発話自身の発話IDを出力する。これに対しDLAでは、先行発話群を過去の判定によって構築された対話クラスタ集合として構造化し、LLMへ提示する。各対話クラスタには、話者リストや最終発話時刻といった、対話に関する情報も付与している。LLMは、対象発話が既存の対話クラスタのいずれに帰属するか、あるいは新規対話の開始であるかを判定する。LLMの出力はJSON形式で3つのキーを指定しており、(1) 対象発話が新規

対話の始まりであるかどうかの2値フラグ、(2) 既存の対話の続きである場合にはその対話のID、(3) 判断理由、をJSON形式で出力させる。

次に、後続文脈の利用 (SC: Subsequent Context) について述べる。会話の文脈を追跡するうえで、発話の意図や指示語の参照先は、その後の発話の反応によって初めて明確化される場合がある。本手法では、対象発話に続く一定数の発話群を補助的な判断材料としてプロンプトに含める。LLMに後続発話を参照させることで、対象発話に対する理解度を向上させることが目的である。

本手法は、プロンプトに含める先行発話、後続発話の件数をハイパーパラメータとして持ち、指定された範囲内にある発話のみがLLMに与えられる。プロンプトへ組み込まれるチャットデータは、先行発話群から作られた対話系列、対象発話、および後続する発話系列である。LLMはこのチャットデータと対話分離のタスク指示を受け、JSON形式で3つのキーを出力することにより応答関係を判定する。判定に基づき対話構造は更新され、次ステップの入力として利用される。この反復的なプロセスにより、会話の対話構造を判定する。詳細なタスク定義は付録Aに、プロンプトの概略を付録Bに示す。

本研究では、DLAとSCの有効性を検証するため4つの設定を用意した。第1に、DLAとSCの双方を利用しない **Baseline** である。これは従来の機械学習における発話間ペアワイズ分類をLLMで再現したものであるが、出力形式のみ本手法と整合させている。すなわち、DiHRL[11]とタスク定式化は同一であるが、出力形式が異なる。第2に、SCを除外しDLAのみを適用する **DLA** である。第3に、DLAを除外しSCのみを適用する **SC** である。第4に、両手法を統合した **DLA+SC** である。

3 実験設定

3.1 データセット

本研究では、対話分離の標準的なベンチマークであるUbuntu IRCデータセット[12]を使用した。本チャットデータは、Ubuntu OSに関する技術的な課題解決を目的とした実際のチャットログから構成されており、複数の参加者による交錯した対話が含まれている。データセットはtrain, dev, testの3つに分割されている。このうちdevおよびtestセットは、複数のアノテータによる検証を経た高品質な正解

LLMへの指示：対象発話が応答関係を持つ対話のIDを示してください。もし新しい対話の始まりであった場合、“new”を出力してください。

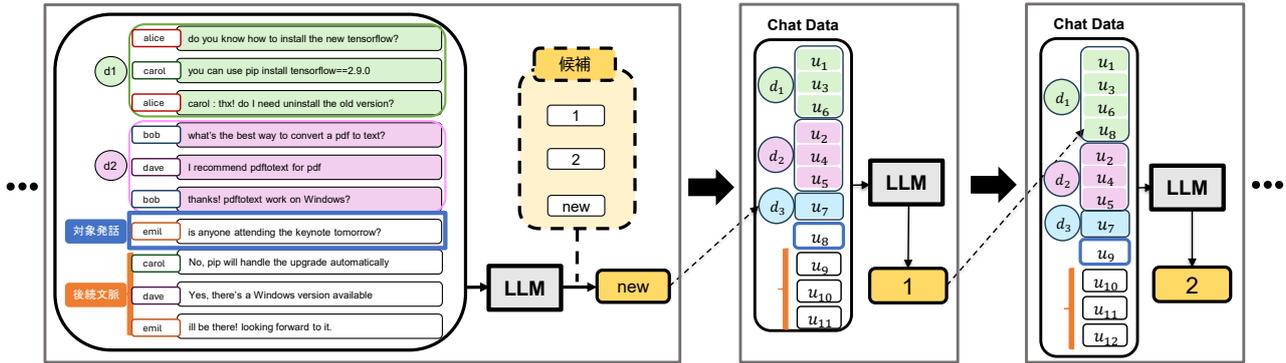


図2 DLA+SCにおけるLLMへの指示内容と判定フローの全体像

データを含み、devは2500件、testは5000件の発話を含む。先行研究との比較を行う実験ではtestセットを使用し、モデルの挙動分析やアブレーション研究にはdevセットを用いた。

3.2 評価指標

評価の公平性を期すため、最新の先行研究であるDiHRL[11]に準拠した指標を採用した。評価の観点は2つである。第1に、クラスタリングの全体的な整合性を評価するため、Variation of Information (VI) [13]およびOne-to-One accuracy (1-1) [2]を用いた。第2に、対話クラスタの完全一致した数を評価する、適合率 (P)、再現率 (R)、およびF1スコアを用いた[12]。クラスタ単位の完全一致は難易度が高く、最も厳格な基準である。アブレーション研究ではF1スコアのみ分析対象として用いた。

3.3 実験設定

最先端LLMによる評価と広範なアブレーション研究を行うため、クローズドおよびオープン、両者のLLMを用いた。クローズドモデルへの入力チャットデータの範囲は、先行発話を70件、後続発話を40件に固定した。一方、オープンソースモデルについては、プロンプト内の発話件数が精度に与える影響を調査するため、発話件数を可変パラメータとして扱った。なお、全ての実験において一貫した出力を促すため、temperatureを0.0に設定した。

4 実験結果

4.1 予備実験

LLMを用いた対話分離において、提案手法DLA+SCの優位性を検証する。比較対象として、

表1 LLM対話分離におけるDLA+SCの有効性

	Method	VI	1-1	P	R	F1
GPT4.1-mini	DiHRL[11]	57.39	26.40	0.00	0.00	0.00
	DLA+SC	<u>86.02</u>	<u>72.60</u>	<u>8.11</u>	<u>8.82</u>	<u>8.45</u>
GPT4.1	DiHRL[11]	58.40	30.60	0.00	0.00	0.00
	DLA+SC	<u>92.48</u>	<u>84.40</u>	<u>30.56</u>	<u>32.35</u>	<u>31.43</u>
Gemini2.5-flash	DiHRL[11]	73.51	50.80	11.11	11.76	11.43
	DLA+SC	94.59	89.80	<u>44.12</u>	<u>44.12</u>	<u>44.12</u>
Gemini2.5-pro	DiHRL[11]	85.28	72.00	33.33	29.41	31.25
	DLA+SC	<u>93.82</u>	<u>84.80</u>	56.25	52.94	54.55

LLMを対話分離に適用した唯一の先行研究であるDiHRL[11]で考案されたLLMベースの手法を採用し、実験データにはIRC devセット500発話を用いた。結果を表1に示し、下線はモデルごとの最高値、太字は全手法中の最高値を表す。全てのLLMにおいて、提案手法はDiHRL[11]を凌駕した。

4.2 ベンチマーク評価

表2 提案手法と既存手法の比較

	Method	VI	1-1	P	R	F1
GPT4.1	Elsner [1]	82.10	51.40	12.10	21.50	15.50
	BERT [14]	90.80	75.00	32.50	29.30	36.60
	BERT+MF [15]	92.00	77.00	—	—	40.90
	PtrNet [16]	94.20	80.10	44.90	44.20	44.50
	DiaBERT [9]	93.20	79.70	42.10	47.90	44.80
	Struct [10]	<u>94.60</u>	<u>84.20</u>	<u>51.80</u>	<u>51.80</u>	<u>51.70</u>
	DiHRL [11]	94.23	<u>84.20</u>	47.97	49.86	48.90
	DLA+SC	95.39	86.34	46.38	48.73	47.53
	DLA+SC	96.88	90.88	60.39	60.56	60.48
Gemini2.5-pro	DLA+SC	97.16	90.78	58.81	63.94	61.27

提案手法と非LLMベースの既存手法との比較結果を表2に示す。実験データにはIRC testセット5000発話を使用した。表中の下線は既存手法および

び提案手法それぞれの最高値、太字は全手法中の最高値を表す。Gemini2.5 使用時の提案手法は全ての評価指標において既存手法を凌駕した。本結果は、長らく非 LLM の教師あり機械学習が支配的であった対話分離において、LLM ベースの手法が従来手法を凌駕することを実証するものである。

4.3 アブレーション研究

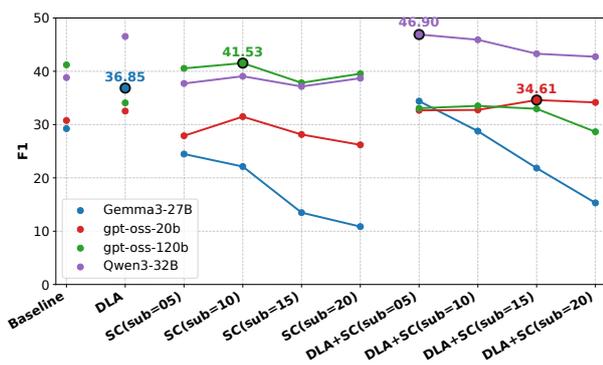


図3 モデル、手法ごとのアブレーション

DLA と SC の個別の寄与を定量的に検証する。IRC dev セット 2500 発話における F1 スコアを計測し、図 3 にその結果を示す。図中の *sub* は後続文脈として与えた発話数を表し、各モデルの最高値が強調表示されている。最高性能を達成する手法の組み合わせはモデルごとに異なることが判明した。しかし、いずれのモデルにおいても、DLA または SC の導入が性能向上をもたらしている。よって、両手法は高精度な対話分離を実現するための要素であると考えられる。

続いて、プロンプト内の発話件数と精度との関係を分析した。Qwen3-30B-A3B-thinking-2507 を用い、IRC dev セット 250 発話を対象とした。先行発話を 40 件から 100 件まで 20 件ずつ、後続発話を 0 件から 40 件まで 10 件ずつ変動させ、精度の推移を分析した。分析の結果、発話件数と精度の間に一貫した傾向は見出せなかった。なお、本実験における最高精度は、先行発話 100 件後続発話 40 件の条件下、SC のみを適用した際に達成された。

5 誤判定の分析

オープンモデルの出力に対し、定量的な分析を実施した。プロンプトに含まれる発話数、トークン数、および割り当て候補数は、精度との間に統計的に有意な相関を示さなかった。ただし、この知見は本実験設定の範囲内に限定されるものであり、LLM

の一般的特性として結論付けるものではない。

さらに、Qwen3-32B の出力に対して定性的な分析を行った。誤判定には主に 2 つの傾向が確認された。第 1 に、短い相槌や挨拶などの発話（例：“hi”、“WTF!”）における誤りである。これらは文脈上、近接する同一話者の発話へ割り当てべきである。しかし、これらを誤って無関係な対話クラスターへ割り当ててしまう傾向が見られた。この事象は、LLM が話者の連続性よりも、発話単体の意味的内容を重視して判定している可能性を示唆する。あるいは、推論過程において話者の考慮が不十分であることに起因すると推察される。第 2 に、特定の専門用語を含む発話（例：Ubuntu のバージョンを指す“hoary”）における誤りである。この誤判定は、LLM ベースの対話分離の判定において、表層的な意味の単語からの推測ではなく、発話内容に対する深い意味理解が重要である可能性を示唆している。同時に、判定の際に専門知識を適切に外挿することが、精度向上に寄与する可能性があるかと推察される。

6 おわりに

LLM を用いた対話分離が既存の最先端手法を凌駕することを実証した。提案手法である DLA および SC の有効性はモデルに依存する一方、精度向上の要因であることが示された。本研究が対話分離タスクを従来の非 LLM 手法から LLM 中心のアプローチへと転換させることを期待する。

7 限界と課題

本手法は、参照可能な文脈範囲に限界がある。入力可能なトークン数には上限があり、範囲外にある長距離の応答関係は捕捉できない。範囲の拡大は、モデルの許容するプロンプト容量を超過するリスクを伴う。加えて、運用コストも課題である。発話 1 件ごとの反復的な処理と長大なプロンプトが必要であり、時間的および金銭的なコストが高い。

本手法の汎用性には議論の余地がある。使用したデータセットはプログラミングの技術的な話題に特化しており、他ドメインでの会話内容について有効性は未検証である。また、本チャットデータは公開されており、LLM の事前学習データに含まれている可能性も否定できない。さらに、プロンプト設計にも改善の余地が残されている。本研究では、単純な Zero-shot を用いたが、Few-shot や Chain-of-Thought 等の導入により、精度向上が見込まれる。

謝辞

本研究の一部は、N E D O（国立研究開発法人新エネルギー・産業技術総合開発機構）の委託事業「経済安全保障重要技術育成プログラム／先進的サイバー防御機能・分析能力強化」（JPNP24003）によるものである。また、本研究の一部はJSPS 科研費 JP24K15084, JP23H00491 の助成を受けたものである。

参考文献

- [1] Micha Elsner and Eugene Charniak. You talking to me? a corpus and algorithm for conversation disentanglement. In **Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)**, pp. 834–842. Association for Computational Linguistics, 2008.
- [2] Micha Elsner and Eugene Charniak. Disentangling chat. In **Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics**, pp. 117–126. Association for Computational Linguistics, 2010.
- [3] Yawen Ouyang, Moxin Chen, Xinyu Dai, Yinggong Zhao, Shujian Huang, and Jiajun Chen. Dialogue state tracking with explicit slot connection modeling. In **Proceedings of the 58th annual meeting of the association for computational linguistics**, pp. 34–40, 2020.
- [4] Zheng Zhang, Lizi Liao, Minlie Huang, Xiaoyan Zhu, and Tat-Seng Chua. Neural multimodal belief tracker with adaptive attention for dialogue systems. In **The world wide web conference**, pp. 2401–2412, 2019.
- [5] Xiaoyu Cai, Yao Fu, Hong Zhao, Weihao Jiang, and Shiliang Pu. Memory graph with message rehearsal for multi-turn dialogue generation. In **Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM)**, pp. 108–117, Atlanta, GA, USA, 2022. Association for Computing Machinery. CC BY-SA 4.0.
- [6] Jia-Chen Gu, Chao-Hong Tan, Chongyang Tao, Zhen-Hua Ling, Huang Hu, Xiubo Geng, and Daxin Jiang. HeterMPC: A heterogeneous graph neural network for response generation in multi-party conversations. In **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume1: Long Papers)**, pp. 5086–5097, Dublin, Ireland, May 2022. Association for Computational Linguistics.
- [7] Yunlong Liang, Fandong Meng, Ying Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. Infusing multi-source knowledge with heterogeneous graph neural network for emotional conversation generation. In **Proceedings of the AAIL conference on artificial intelligence**, Vol. 35, pp. 13343–13352, 2021.
- [8] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Evangelos Kanoulas, Christof Monz, and Maarten De Rijke. Conversations with search engines: Serp-based conversational response generation. **ACM Transactions on Information Systems (TOIS)**, Vol. 39, No. 4, pp. 1–29, 2021.
- [9] Tianda Li, Jia-Chen Gu, Xiaodan Zhu, Quan Liu, Zhen-Hua Ling, Zhiming Su, and Si Wei. Dialbert: A hierarchical pre-trained model for conversation disentanglement. **arXiv preprint arXiv:2004.03760**, 2020.
- [10] Xinbei Ma, Zhuosheng Zhang, and Hai Zhao. Structural characterization for dialogue disentanglement. **arXiv preprint arXiv:2110.08018**, 2021.
- [11] Bobo Li, Hao Fei, Fei Li, Shengqiong Wu, Lizi Liao, Yinwei Wei, Tat-Seng Chua, and Donghong Ji. Revisiting conversation discourse for dialogue disentanglement. **ACM Transactions on Information Systems**, Vol. 43, No. 1, pp. 1–34, 2025.
- [12] Jonathan K Kummerfeld, Sai R Gouravajhala, Joseph J Peper, Vignesh Athreya, Chulaka Gunasekara, Jatin Ganhotra, Siva Sankalp Patel, Lazaros C Polymenakos, and Walter Lasecki. A large-scale corpus for conversation disentanglement. In **Proceedings of the 57th annual meeting of the association for computational linguistics**, pp. 3846–3856, 2019.
- [13] Marina Meilă. Comparing clusterings by the variation of information. In **Learning Theory and Kernel Machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24–27, 2003. Proceedings**, pp. 173–187. Springer, 2003.
- [14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. **arXiv preprint arXiv:1907.11692**, 2019.
- [15] Rongxin Zhu, Jey Han Lau, and Jianzhong Qi. Findings on conversation disentanglement. In Afshin Rahimi, William Lane, and Guido Zuccon, editors, **Proceedings of the 19th Annual Workshop of the Australasian Language Technology Association**, pp. 1–11, Online, December 2021. Australasian Language Technology Association.
- [16] Tao Yu and Shafiq Joty. Online conversation disentanglement with pointer networks. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 6321–6330, Online, November 2020. Association for Computational Linguistics.

A DLA+SC の定義

Algorithm 1 DLA+SC (Dialogue-Level Assignment + Subsequent Context)

```

1: Input:  $C = (u_1, u_2, \dots, u_n)$ , where  $u_i = (t_i, s_i, m_i)$  with  $t_i$  being the timestamp,  $s_i$  the speaker ID, and  $m_i$  the message content; window sizes  $W_{prev}$  and  $W_{subs}$ ; max retries  $N \leftarrow 5$ 
2: Output: A partition of  $C$  into dialogues  $D = \{d_1, d_2, \dots, d_p\}$ 
3: function GREEDYDISENTANGLE( $C, W_{prev}, W_{subs}$ )
4:    $D \leftarrow \{\{u_1\}\}$ 
5:   for  $i \leftarrow 2$  to  $n$  do
6:      $u_{target} \leftarrow u_i; C_{prev} \leftarrow \{u_k \mid \max(1, i - W_{prev}) \leq k < i\}; C_{subs} \leftarrow \{u_k \mid i < k \leq \min(n, i + W_{subs})\}$ 
7:      $D_{candidate} \leftarrow \{d \in D \mid d \cap C_{prev} \neq \emptyset\}; Prompt \leftarrow \text{FORMATPROMPT}(D_{candidate}, u_{target}, C_{subs})$ 
8:      $assignment \leftarrow \text{"new"}; Prompt^* \leftarrow Prompt$ 
9:     for  $retry \leftarrow 1$  to  $N$  do
10:       $Output \leftarrow \text{LLM}(Prompt^*)$  ▷ The LLM returns a reference to  $d_j \in D_{candidate}$  or "new"
11:       $is\_valid \leftarrow \text{ISJSONVALID}(Output)$  ▷ Check if JSON is valid
12:      if  $is\_valid$  then
13:         $assignment \leftarrow \text{PARSEJSON}(Output)$ 
14:        break ▷ Exit retry on valid json
15:      else
16:         $Prompt^* \leftarrow Prompt + \text{warning\_message}$  ▷ Append warning for next attempt
17:      if  $assignment$  indicates "new" then  $D \leftarrow D \cup \{\{u_{target}\}\}$  else  $d_j \leftarrow d_j \cup \{u_{target}\}$ 
18:    return  $D$ 
19: function FORMATPROMPT( $D_{candidate}, u_{target}, C_{subs}$ )
20:    $D_{candidate}^* \leftarrow \{\}$ 
21:   for each  $d_k \in D_{candidate}$  do
22:      $S_k \leftarrow \{s \mid \exists u \in d_k, u = (t, s, m)\}$  ▷ Unique list of speakers participating in  $d_k$ 
23:      $t_{last} \leftarrow \max\{t \mid \exists u \in d_k, u = (t, s, m)\}$  ▷ Timestamp of last utterance in  $d_k$ 
24:      $dt_k \leftarrow t_{target} - t_{last}$  ▷ Time gap from target to last utterance in  $d_k$ 
25:      $D_{candidate}^*[k] \leftarrow \{S_k, dt_k, d_k\}$ 
26:    $Prompt \leftarrow \text{instruct}_{disentangle} + D_{candidate}^* + u_{target} + C_{subs} + u_{target} + \text{instruct}_{output}$ 
27:   return  $Prompt$ 

```

B プロンプト概略

```

# Instruction
You are given a multi-user chat with each line labeled with an index number, timestamp, speaker's name, and text message for an utterance. Your task is to identify to which previous dialogue the target utterance is responding. Assign the target utterance to exactly one existing dialogue from candidate dialogue IDs, or determine it starts a new dialogue. Note that the utterance is more likely to be responding to a nearby one. Subsequent utterances are provided only as reference information - never select the id from Subsequent utterances.
# Chat data
## Candidate dialogue IDs 136, 169, 173, 175, 184
## Previous dialogues
[ { "dialogue_id": "136",
  "participants": [ "Elementalist", ... "worktoner" ],
  "time_difference_from_target_seconds": 480.0,
  "utterances": [
    { "utterance_id": 136,
      "timestamp": "2016-12-19T21:18:00.000000",
      "speaker": "MOUD",
      "message": "Hey all" }, ...
  ]
}, ...
]
## Target utterance
...
## Subsequent utterances
...
# Output (JSON ONLY)
- "is_new_dialogue": boolean(true if the target is determined to be the start of a new dialogue, false if it is a continuation of an existing dialogue).
- "dialogue_id": If is_new_dialogue is true, set to null. If is_new_dialogue is false, set the ID of the selected candidate dialogue.
- "reason": Detailed explanation based as to why that decision (start of a new dialogue/continuation of an existing dialogue) was made.
Output example:
{ "is_new_dialogue": false,
  "dialogue_id": "17",
  "reason": "..."}

```