Findings on Conversation Disentanglement

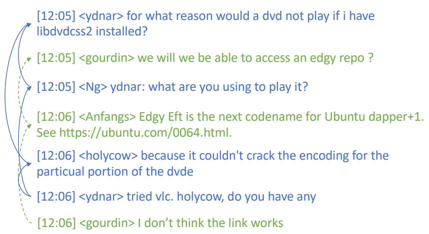
Rongxin Zhu, Jey Han Lau, Jianzhong Qi

School of Computing and Information System, The University of Melbourne

Introduction

1.1 Problem Definition

Conversation disentanglement aims at identifying separate threads in multi-party conversations, acting as an important preprocessing step for high-level tasks of multi-party conversations such as conversation summarization and response generation. The figure below shows two threads in different colors, with reply-to links between utterances.



1.2 Limitation of previous methods

- transformer-based models are not systematically compared with respect to performance, memory consumption and speed
- previous methods don't leverage **dialogue history** when measuring the similarity between UOI and a candidate, or introduce **noise** in context expansion
- greedy decoding algorithm recovers threads by finding the parent utterance for each utterance of interest (UOI) **independently**

Methodology

2.1 Pairwise Model

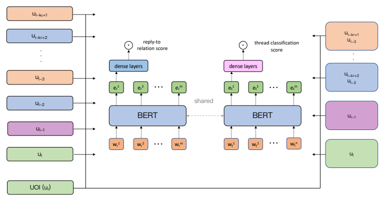
We conduct empirical study to compare rule-based, feature-based and transformer models that measures the similarity between UOI and each candidate (we regard k=50 past utterances, including UOI, as the candidate pool). Experiments show that BERT [1] +MF (manual features) is still a strong baseline.

Model	Link Prediction Precision	Recall	F1	Ranking R@1	R@5	R@10	Clustering 1-1	VI	F
Last Mention	37.1	35.7	36.4	-	-	-	21.4	60.5	4.0
GLOVE+MF	71.5	68.9	70.1	70.2	95.8	98.6	76.1	91.5	34.0
MF	71.1	68.5	69.8	70.2	94.0	97.3	75.0	91.3	31.5
POLY-BATCH	39.3	37.9	38.6	40.8	69.8	80.8	52.3	80.8	9.8
POLY-INLINE	42.2	40.7	41.4	42.8	70.8	81.3	62.0	84.4	13.6
ALBERT	46.1	44.4	45.3	46.8	77.3	88.4	68.6	87.9	22.4
BERT	48.2	46.4	47.3	48.8	75.4	84.7	74.3	89.3	26.3
BERT+TD	67.9	65.4	66.6	66.9	90.6	95.3	76.0	91.1	34.9
BERT+MF	**73.9**	**71.3**	**72.6**	**73.9**	**95.8**	**98.6**	**77.0**	**92.0**	**40.9**

Model	GPU Mem (GB)	Speed (ins/s)
BERT	18.7	9.4
ALBERT	14.6	9.4
POLY-INLINE	9.9	16.8
POLY-BATCH	5.1	36.4

We compare the GPU memory consumption and speed of four transformer models, finding that Poly-encoder [2] is the fastest and most memory efficient one, with a sacrifice of performance.

2.2 Context Expansion using Multi-task Learning



$$L = -\Big(\sum_{i=1}^{N} \sum_{j=1}^{k_c} \mathbb{1}(y_r = j) \log s_{ij}^{r} + \alpha \sum_{i=1}^{N} \sum_{l=1}^{k_t} \mathbb{1}(y_t = l) \log s_{il}^{t} \Big)$$

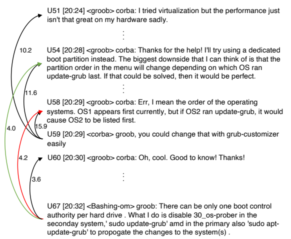
Multi-task Learning: we conduct utterance-to-utterance classification (reply-to relation identification) and utterance-to-thread classification (assign each UOI to an existing thread or create a new thread) at the same time, using a shared BERT model with separate dense layers for two tasks. In training, we use ground truth reply-to relations and thread labels. During inference, we only do reply-to relation identification, which does not introduce noise if the predicted threads are not completely correct (e.g., a predicted thread may contain utterances from other threads). The loss function is a weighted sum of two loss terms for reply-to relation identification and thread classification, respectively.

Model	Link Prediction Precision	Recall	F1	Ranking R@1	R@5	R@10	Clustering 1-1	VI	F
BERT	48.2	46.4	47.3	48.8	75.4	84.7	74.3	89.3	26.3
BERT+MF	73.9	71.3	72.6	73.9	95.8	98.6	77.0	92.0	40.9
MULTI ($\alpha = 1$)	65.6	63.2	64.4	66.7	91.8	95.6	64.6	87.7	24.3
MULTI ($\alpha = 5$)	66.9	64.5	65.7	65.4	91.8	95.6	68.7	88.8	27.4
MULTI ($\alpha = 10$)	65.2	62.9	64.0	64.4	91.4	95.6	70.3	89.5	28.1
MULTI ($\alpha = 20$)	64.7	62.4	63.5	63.9	91.0	95.0	68.3	88.8	26.7
MULTI+MF ($\alpha = 1$)	72.8	70.2	71.5	71.9	94.0	96.4	76.3	91.8	36.1
MULTI+MF ($\alpha = 5$)	73.3	70.7	72.0	72.4	94.0	96.5	72.8	90.8	33.1
MULTI+MF ($\alpha = 10$)	72.2	69.6	70.8	70.4	93.4	96.4	71.8	90.2	29.9
MULTI+MF ($\alpha = 20$)	70.8	68.2	69.5	69.4	93.4	97.3	73.2	90.6	28.6

Our multi-task learning framework is helpful when manual features are not available. It doesn't outperform BERT+MF pairwise model.

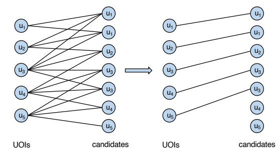
2.3 Bipartite Graph Matching for Conversation Disentanglement

Motivation: pairwise models achieve a high recall@5, which means the correct parents are ranked high but not at the top sometimes. Bipartite matching-based algorithm recovers threads by identifying the parent utterance of a set of UOIs jointly. When conflict occurs (multiple UOIs choose the same parent), some UOI may fall back to choose its second-best candidate as parent.



U_{67} chooses U_{54} as parent in global decoding algorithm but chooses U_{58} in greedy algorithm.

We first build a bipartite graph with two sets of nodes, containing all UOIs and all candidates, respectively. Edges are created between each UOI, and its top-5 ranked candidates from pairwise models. Then we frame conversation disentanglement as a **maximum-weight bipartite matching [3]** problem. The aim is to find a subset of edges, representing the predicted reply-to relations.



	Precision	Recall	F1
Oracle	88.4	85.2	86.8
Rule-Based	73.7	70.9	72.3
FFN	73.8	71.0	72.3
BERT+FFN	72.9	70.3	71.5

The bipartite graph matching-based algorithm has the potential to outperform greedy approaches. Oracle means we use ground truth node frequencies for all candidate nodes. We tried to predict node frequencies, but the results are not ideal. More effective methods to predict node frequencies are needed.

Conclusion

- BERT combined with manual features is still a strong baseline for conversation disentanglement
- The multi-task learning framework that conducts utterance-to-utterance and utterance-to-thread classification at the same time outperforms pairwise models when manual features are not available
- Bipartite graph matching-based conversation disentanglement shows potential to outperform greedy approaches.

References

1. Jacob Devlin, Ming-Wei Chang, Kenton Lee, andKristina Toutanova. 2019. BERT: Pre-training ofdeep bidirectional transformers for language under-standing. InProceedings of the 2019 Conferenceof the North American Chapter of the Associationfor Computational Linguistics: Human LanguageTechnologies, Volume 1 (Long and Short Papers),pages 4171–4186, Minneapolis, Minnesota. Associ-ation for Computational Linguistics.
2. Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux,and Jason Weston. 2019. Poly-encoders: Architec-tures and pre-training strategies for fast and accuratemulti-sentence scoring. InInternational Conferenceon Learning Representations.Y. Zhang, C. Bajaj, B.-S. Sohn. Adaptive and Quality 3D Meshing from Imaging Data, ACM Symposium on Solid Modeling and Applications. pp. 286-291, Seattle, June 2003.
3. AMH Gerards. 1995. Matching.Handbooks in opera-tions research and management science, 7:135–224.

The 19th Annual Workshop of the Australasian Language Technology Association. 8-10 December 2021

* Please contact rongxinz1@student.unimelb.edu.au for further information.