Towards Modeling Role-Aware Centrality for Dialogue Summarization

Xinnian Liang\textsuperscript{1,}\textsuperscript{1} Chao Bian\textsuperscript{2,}\textsuperscript{2} Shuangzhi Wu\textsuperscript{2}, and Zhoujun Li\textsuperscript{1}

\textsuperscript{1}State Key Lab of Software Development Environment, Beihang University, Beijing, China
\textsuperscript{2}ByteDance Lark AI, Beijing, China

\{xnliang, lizj\}@buaa.edu.cn; {wufurui, zhangchaoyue.0}@bytedance.com

Abstract

Role-oriented dialogue summarization generates summaries for different roles in dialogue (e.g. doctor and patient). Existing methods consider roles separately where interactions among different roles are not fully explored. In this paper, we propose a novel Role-Aware Centrality (RAC) model to capture role interactions, which can be easily applied to any seq2seq models. The RAC assigns each role a specific sentence-level centrality score by involving role prompts to control what kind of summary to generate. The RAC measures both the importance of utterances and the relevance between roles and utterances. Then we use RAC to re-weight context representations, which are used by the decoder to generate role summaries. We verify RAC on two public benchmark datasets, CSDS and MC. Experimental results show that the proposed method achieves new state-of-the-art results on the two datasets. Extensive analyses have demonstrated that the role-aware centrality helps generate summaries more precisely.

1 Introduction

The last few years have seen a land rush in research of generating summaries for dialogue such as meeting text and daily chatting due to the ever growing dialogue corpus from online conversation tools (Zhu et al., 2020; Feng et al., 2021a; Zhong et al., 2021; Chen and Yang, 2021; Liu and Chen, 2021). Typically, Dialogue summarization aims at compressing the main content of a long conversation into a short text (Qi et al., 2021; Zou et al., 2021; Feng et al., 2021b; Zhang et al., 2022). Different from traditional summarization tasks on document text, the main challenge of dialogue summarization is to summarize from utterances of multiple roles, who may have different opinions and interact with some of the other roles (Lin et al., 2021, 2022).

\textsuperscript{1}The authors contribute equally
\textsuperscript{2}Contribution during internship at ByteDance Inc.
\textsuperscript{3}Corresponding Author

Figure 1: A dialogue summarization example.

Recently, Lin et al. (2021) pointed out that it is equally important to summarize the main content of each role in addition to the whole dialogue. Thus, they proposed a more practical task: The role-oriented dialogue summarization, which aims at generating summaries for specified roles, e.g. user summary and agent summary. Figure 1 shows an example of customer service and user dialogue about changing order delivery address. The role-oriented dialogue summarization generates summary for both user (e.g. User Summary) and agent (e.g. Agent Summary). The two summaries are different in content and opinion. Additionally, there is also an overall summary to summarize the whole dialogue.

There are several methods focused on the role-oriented summarization task. Lin et al. (2021) trains different models for different role-oriented summaries by splitting their utterances, however, they ignore interactions between roles. Lin et al. (2022) proposed a role-interaction attention model. They modeled role-wise interactions through cross-attention and self-attention in the decoder. How-
ever, their method has to assign each role a specific decoder. In addition, the role-interaction has to be conducted between every two roles. That means both the model parameter and complexity increase with the number of roles.

In this paper, we propose a novel Role-Aware Centrality (RAC) model for the role-oriented dialogue summarization task. Centrality is widely used to measure the salience of sentences in a given document (Zheng and Lapata, 2019; Liang et al., 2021, 2022). The RAC assigns each role a specific Centrality. Specifically, we first propose a role prompt that is attached to the start of the dialogue. The role prompt is used to guide what kind of summary to generate (i.e. user summary or agent summary). Then we compute the centrality scores of each utterance. The final Role-Aware Centrality is calculated by an interaction of role prompt and centrality scores. During decoding, we use the RAC to reweight the dialogue context, which is used by the decoder to generate the summaries. We propose role prompts for each role together with the overall summary. In this way, different summaries can be modeled in a unified seq2seq framework. In addition, the RAC can be easily applied to any sequence-to-sequence model with any number of roles. To evaluate the effectiveness of the RAC, we apply the RAC to three types of seq2seq structure: PGN, BERTAbs, and BART, and verify the models on two public Chinese dialogue summarization datasets: CSDS and MC. Experimental results show that our RAC can improve all of their performance while accelerating the convergence of training. Additionally, the RAC based BART achieves new state-of-the-art results.

We summarize our contributions as follows:

- We propose a novel Role-Aware Centrality (RAC) model for the role-oriented dialogue summarization task to model both role-aware salient context and role interactions.

- The RAC models different kinds of summaries in a unified seq2seq framework without computational complexity increasing as roles increase.

- Our model can be applied to different seq2seq models, where the RAC-based BART achieves new state-of-the-art results.

2 Methodology

In this section, we will introduce our proposed Role-Aware Centrality (RAC) model and the combination with the seq2seq structure. The main framework is shown in Figure 2. It consists of three components: bidirectional encoder, role-aware centrality model, and auto-regression decoder.

2.1 Task Formalization

Given a dialogue $\mathcal{D}$ with $n$ utterances $\{u_1, \ldots, u_n\}$ and $m$ roles $\{r_1, \ldots, r_m\}$. Each utterance $u_i$ contains a role $r_k \in R$ and text content $s_i$. We simply concatenate them by “:” and get utterance $u_i = r_k : s_i$. For different roles $r_k$, the data have different summary $y^{r_k}$. In this paper, we employ $y^{user}$ and $y^{agent}$ to represent summaries of two roles and $y^{final}$ to represent the summary of the whole dialogue. Our method can also be applied for datasets with multi-roles.

2.2 Role Prompts

Previous models always trained different models for different role-oriented summary generation. Lin et al. (2022) pointed out that it hurts the performance of the model. We employ role prompts to
control the generation of different summaries and this ensures we only train a single model. Specifically, we attach “[User Summary]”, “[Agent Summary]”, and “[Final Summary]” to the start of each dialogue for summaries generation. The input context is re-formalized as “[Prompt] Dialogue Contexts” and then tokenized as $T$ tokens $\{t_i\}_{i=1}^T$. We show the main results in Table 1 and Table 2. All reported results of [model]+RAC are the average of three checkpoints. The bold number represents the best result for each block, and the underlined represents the best global result. BERT model in the table means BERTAbs. We can see that BART+RAC outperforms all comparison models and achieve state-of-the-art results on CSDS and MC datasets. In addition, different types of seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained models, achieve the best results. This proves that the knowledge learned in the pre-training phase of

$$P(\hat{y}) = \text{Decoder}(\{h_i\}_{i=1}^T) \quad (5)$$

In the training stage, the model learns the optimal parameters $\theta$ by minimizing the negative log-likelihood.

3 Experiments and Analysis

3.1 Basic Settings

We evaluate our method on two public datasets: CSDS (Lin et al., 2021) and MC (Song et al., 2020). The comparison baselines are PGN (See et al., 2017), BERTAbs (Liu and Lapata, 2019), PGN/BERTAbs-both (Lin et al., 2022) and our implemented BART-both. The comparison metrics are ROUGE-2 / L (Lin, 2004)², BLEU (Papineni et al., 2002)³, BERTScore (Zhang* et al., 2020), and MoverScore (Zhao et al., 2019)⁵. For MoverScore, we use Chinese-bert-wwm-ext⁶ to provide the embeddings of summaries. The results of ROUGE-1 and more details of experiments are shown in the appendix.

3.2 Main Results

We show the main results in Table 1 and Table 2. All reported results of [model]+RAC are the average of three checkpoints. The bold number represents the best result for each block, and the underlined represents the best global result. BERT model in the table means BERTAbs. We can see that BART+RAC outperforms all comparison models and achieve state-of-the-art results on CSDS and MC datasets. In addition, different types of seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained models, achieve the best results. This proves that the knowledge learned in the pre-training phase of

$$\lambda$$ is a hyperparameter to control the influence of RAC. The auto-regression decoder generates the final summary based on the re-weighted context representations $\{h_i\}_{i=1}^T$. $P(\hat{y}) = \text{Decoder}(\{h_i\}_{i=1}^T) \quad (5)$

In the training stage, the model learns the optimal parameters $\theta$ by minimizing the negative log-likelihood.

3 Experiments and Analysis

3.1 Basic Settings

We evaluate our method on two public datasets: CSDS (Lin et al., 2021) and MC (Song et al., 2020). The comparison baselines are PGN (See et al., 2017), BERTAbs (Liu and Lapata, 2019), PGN/BERTAbs-both (Lin et al., 2022) and our implemented BART-both. The comparison metrics are ROUGE-2 / L (Lin, 2004)², BLEU (Papineni et al., 2002)³, BERTScore (Zhang* et al., 2020), and MoverScore (Zhao et al., 2019)⁵. For MoverScore, we use Chinese-bert-wwm-ext⁶ to provide the embeddings of summaries. The results of ROUGE-1 and more details of experiments are shown in the appendix.

3.2 Main Results

We show the main results in Table 1 and Table 2. All reported results of [model]+RAC are the average of three checkpoints. The bold number represents the best result for each block, and the underlined represents the best global result. BERT model in the table means BERTAbs. We can see that BART+RAC outperforms all comparison models and achieve state-of-the-art results on CSDS and MC datasets. In addition, different types of seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained models, achieve the best results. This proves that the knowledge learned in the pre-training phase of

$$\lambda$$ is a hyperparameter to control the influence of RAC. The auto-regression decoder generates the final summary based on the re-weighted context representations $\{h_i\}_{i=1}^T$. $P(\hat{y}) = \text{Decoder}(\{h_i\}_{i=1}^T) \quad (5)$

In the training stage, the model learns the optimal parameters $\theta$ by minimizing the negative log-likelihood.

3 Experiments and Analysis

3.1 Basic Settings

We evaluate our method on two public datasets: CSDS (Lin et al., 2021) and MC (Song et al., 2020)¹. The comparison baselines are PGN (See et al., 2017), BERTAbs (Liu and Lapata, 2019), PGN/BERTAbs-both (Lin et al., 2022) and our implemented BART-both. The comparison metrics are ROUGE-2 / L (Lin, 2004)², BLEU (Papineni et al., 2002)³, BERTScore (Zhang* et al., 2020), and MoverScore (Zhao et al., 2019)⁵. For MoverScore, we use Chinese-bert-wwm-ext⁶ to provide the embeddings of summaries. The results of ROUGE-1 and more details of experiments are shown in the appendix.

3.2 Main Results

We show the main results in Table 1 and Table 2. All reported results of [model]+RAC are the average of three checkpoints. The bold number represents the best result for each block, and the underlined represents the best global result. BERT model in the table means BERTAbs. We can see that BART+RAC outperforms all comparison models and achieve state-of-the-art results on CSDS and MC datasets. In addition, different types of seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained models, achieve the best results. This proves that the knowledge learned in the pre-training phase of

$$\lambda$$ is a hyperparameter to control the influence of RAC. The auto-regression decoder generates the final summary based on the re-weighted context representations $\{h_i\}_{i=1}^T$. $P(\hat{y}) = \text{Decoder}(\{h_i\}_{i=1}^T) \quad (5)$

In the training stage, the model learns the optimal parameters $\theta$ by minimizing the negative log-likelihood.

3 Experiments and Analysis

3.1 Basic Settings

We evaluate our method on two public datasets: CSDS (Lin et al., 2021) and MC (Song et al., 2020)¹. The comparison baselines are PGN (See et al., 2017), BERTAbs (Liu and Lapata, 2019), PGN/BERTAbs-both (Lin et al., 2022) and our implemented BART-both. The comparison metrics are ROUGE-2 / L (Lin, 2004)², BLEU (Papineni et al., 2002)³, BERTScore (Zhang* et al., 2020), and MoverScore (Zhao et al., 2019)⁵. For MoverScore, we use Chinese-bert-wwm-ext⁶ to provide the embeddings of summaries. The results of ROUGE-1 and more details of experiments are shown in the appendix.

3.2 Main Results

We show the main results in Table 1 and Table 2. All reported results of [model]+RAC are the average of three checkpoints. The bold number represents the best result for each block, and the underlined represents the best global result. BERT model in the table means BERTAbs. We can see that BART+RAC outperforms all comparison models and achieve state-of-the-art results on CSDS and MC datasets. In addition, different types of seq2seq models can all have an appreciable improvement with our RAC and the gain of the BART model is extremely obvious. It is worth mentioning that the performance of the PGN-based models is better than BERTAbs-based models, while the BART-based models, which are also pre-trained models, achieve the best results. This proves that the knowledge learned in the pre-training phase of

$$\lambda$$ is a hyperparameter to control the influence of RAC. The auto-regression decoder generates the final summary based on the re-weighted context representations $\{h_i\}_{i=1}^T$. $P(\hat{y}) = \text{Decoder}(\{h_i\}_{i=1}^T) \quad (5)$

In the training stage, the model learns the optimal parameters $\theta$ by minimizing the negative log-likelihood.
| Model | ROUGE-2 | ROUGE-L | BLEU | BERTScore | MoverScore |
|-------|---------|---------|------|-----------|------------|
| PGN   | 39.19/37.06/35.12 | 53.46/51.05/47.59 | 30.03/29.64/28.25 | 77.96/76.68/76.13 | 59.00/58.68/58.23 |
| PGN-both | 40.37/39.10/36.50 | 55.14/53.85/49.12 | 32.58/33.54/29.78 | 78.69/79.52/76.74 | 59.48/59.32/58.64 |
| PGN+RAC | 40.86/40.74/36.92 | 55.98/54.56/50.04 | 32.94/33.86/30.46 | 78.87/79.90/77.03 | 59.64/59.72/58.61 |
| BERT  | 37.59/36.39/33.82 | 52.40/50.44/46.83 | 29.90/30.17/26.99 | 78.52/79.23/76.39 | 58.23/58.10/57.79 |
| BERT-both | 40.12/40.70/36.37 | 54.87/55.17/49.52 | 32.13/32.04/29.23 | 79.85/80.70/77.23 | 59.52/59.55/58.46 |
| BERT+RAC | 40.34/41.05/36.75 | 55.12/55.53/49.89 | 32.24/32.19/29.91 | 79.89/80.69/77.27 | 59.86/59.58/58.66 |

Table 1: Results on the CSDS dataset test set.

| Model | ROUGE-2 | ROUGE-L | BLEU | BERTScore | MoverScore |
|-------|---------|---------|------|-----------|------------|
| PGN   | 81.25/94.32/77.91 | 84.34/94.77/81.47 | 71.50/87.66/68.10 | 92.90/97.60/91.74 | 80.90/93.84/79.69 |
| PGN-both | 81.93/94.59/78.78 | 84.94/95.06/82.20 | 72.77/87.82/69.63 | 93.23/97.71/92.15 | 81.67/94.04/80.52 |
| PGN+RAC | 82.45/94.72/79.11 | 85.33/96.41/82.76 | 72.98/88.00/69.99 | 93.45/97.92/92.32 | 81.88/94.35/80.83 |
| BERT  | 79.90/94.48/76.78 | 83.04/95.06/80.30 | 68.19/87.20/64.09 | 92.68/97.86/91.71 | 81.28/93.90/80.48 |
| BERT-both | 80.76/94.62/77.54 | 83.68/95.14/80.84 | 69.33/87.40/65.40 | 93.02/97.90/91.91 | 82.69/94.20/81.02 |
| BERT+RAC | 81.30/94.80/77.91 | 84.07/95.22/81.36 | 69.73/87.80/65.91 | 93.11/97.89/92.29 | 82.56/94.41/81.42 |
| BART  | 84.75/94.99/82.33 | 87.38/95.37/85.30 | 73.68/90.29/68.93 | 93.65/97.94/92.63 | 82.35/94.17/81.27 |
| BART-both | 85.22/95.42/82.89 | 87.75/95.11/85.78 | 73.87/90.70/69.31 | 93.69/97.88/92.69 | 82.32/94.02/81.40 |
| BART+RAC | 86.29/95.86/84.58 | 88.47/96.12/86.56 | 74.18/91.22/70.08 | 94.01/98.13/92.84 | 82.88/95.10/81.95 |

Table 2: Results on the MC dataset test set.

| Model | ROUGE-2 | ROUGE-L | BLEU | BERTScore | MoverScore |
|-------|---------|---------|------|-----------|------------|
| BART  | 59.07/58.78/53.89 |
| BART+Prompt | 59.42/58.96/54.03 |
| BART+CW | 59.61/59.13/54.11 |
| BART+RW | 59.64/59.22/54.26 |
| BART+RAC | 59.77/59.54/54.41 |

Table 3: Ablation study on the CSDS dataset.

BERTAbs has a limited gain on generative tasks. Overall, our proposed RAC is effective for role-oriented dialogue summarization tasks.

### 3.3 Ablation Study

We do an ablation study to evaluate the contribution from different components of our proposed RAC mechanism. The improvement of each component for the BART model is shown in Table 3. Prompt represents prompt-based joint training. CW represents utterance centrality weights. RW represents the role-aware relevance weight. From the results, we can see that RW contributes the most performance and all components are vital for the final results of BART+RAC. This result demonstrates the effectiveness of our proposed RAC components.

### 3.4 Convergence Analysis

Our RAC can be seen as prior knowledge to guide the training of the summarization model. To investigate the impact of our RAC, we compare the convergence speed of three models and show it in Figure 3. We can see that BART+RAC can converge to a better result with fewer epochs, proving that RAC provides useful information for the model to summarize the dialogue. Compared with our RAC, BART-both (Lin et al., 2022) makes limited improvement for the BART model.
4 Conclusion

In this paper, we bring the degree centrality into dialogue summarization and proposed a role-aware centrality (RAC) model to capture role-interaction information. Experiments on two datasets demonstrated that our proposed RAC model is effective and achieved new state-of-the-art results. Furthermore, our RAC can models different kinds of summaries in a unified seq2seq framework without computational complexity increasing as roles increase.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 62276017, U1636211, 61672081), the 2022 Tencent Big Travel Rhino-Bird Special Research Program, and the Fund of the State Key Laboratory of Software Development Environment (Grant No. SKLSDE-2021ZX-18).

References

Jiaao Chen and Diyi Yang. 2021. Simple conversational data augmentation for semi-supervised abstractive dialogue summarization. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6605–6616, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiachong Feng, Xiaocheng Feng, Bing Qin, and Xinwei Geng. 2021a. Dialogue discourse-aware graph model and data augmentation for meeting summarization. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pages 3808–3814. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Xiachong Feng, Xiaocheng Feng, Libo Qin, Bing Qin, and Ting Liu. 2021b. Language model as an annotator: Exploring DialoGPT for dialogue summarization. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1479–1491, Online. Association for Computational Linguistics.

Xinnian Liang, Jing Li, Shuangzhi Wu, Mu Li, and Zhoujun Li. 2022. Improving unsupervised extractive summarization by jointly modeling facet and redundancy. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 30:1546–1557.

Xinnian Liang, Shuangzhi Wu, Mu Li, and Zhoujun Li. 2021. Improving unsupervised extractive summarization with facet-aware modeling. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1685–1697, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haitao Lin, Liqun Ma, Junnan Zhu, Lu Xiang, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2021. CSDS: A fine-grained Chinese dataset for customer service dialogue summarization. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 4436–4451, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haitao Lin, Junnan Zhu, Lu Xiang, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2022. Other roles matter! Enhancing role-oriented dialogue summarization via role interactions. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2545–2558, Dublin, Ireland. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Zhengyuan Liu and Nancy Chen. 2021. Controllable neural dialogue summarization with personal named entity planning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 92–106, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

MengNan Qi, Hao Liu, YuZhuo Fu, and Ting Liu. 2021. Improving abstractive dialogue summarization with hierarchical pretraining and topic segment. In Findings of the Association for Computational Linguistics: EMNLP 2021, pages 1121–1130, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.
Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. 2020. Summarizing medical conversations via identifying important utterances. In Proceedings of the 28th International Conference on Computational Linguistics, pages 717–729, Barcelona, Spain (Online).

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In International Conference on Learning Representations.

Yusen Zhang, Ansong Ni, Ziming Mao, Chen Henry Wu, Chenguang Zhu, Budhadiya Deb, Ahmed Awadallah, Dragomir Radev, and Rui Zhang. 2022. Summ: A multi-stage summarization framework for long input dialogues and documents. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1592–1604, Dublin, Ireland. Association for Computational Linguistics.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China. Association for Computational Linguistics.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 6236–6247, Florence, Italy. Association for Computational Linguistics.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 5905–5921, Online. Association for Computational Linguistics.

Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang. 2020. A hierarchical network for abstractive meeting summarization with cross-domain pretraining. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 194–203, Online. Association for Computational Linguistics.

Yicheng Zou, Lujun Zhao, Yangyang Kang, Jun Lin, Minlong Peng, Zhuoren Jiang, Changlong Sun, Qi Zhang, Xuanjing Huang, and Xiaozhong Liu. 2021. Topic-oriented spoken dialogue summarization for customer service with saliency-aware topic modeling. Proceedings of the AAAI Conference on Artificial Intelligence, 35(16):14665–14673.

|       | CSDS | MC  |
|-------|------|-----|
| Train Size | 9,101 | 29,324 |
| Val. Size  | 800  | 3,258 |
| Test Size  | 800  | 8,146 |
| Input Length | 321.92 | 292.21 |
| User Sum. Length | 37.28 | 22.37 |
| Agent Sum. Length | 48.08 | 95.32 |
| Final Sum. Length | 83.21 | 114.54 |

Table 4: Statistical information of two datasets.

|       | CSDS | MC  |
|-------|------|-----|
| PGN   | 55.58/53.55/50.20 | 85.32/94.82/82.56 |
| PGN-both | 57.20/56.08/51.62 | 85.98/95.10/83.37 |
| PGN+RAC | 57.62/56/32/52.01 | 86.38/95.26/83.80 |
| BERT  | 53.87/52.72/49.57 | 84.07/95.10/81.53 |
| BERT-both | 57.24/54.36/51.92 | 84.69/95.18/82.02 |
| BERT+RAC | 57.35/54.75/52.23 | 85.12/95.50/82.62 |
| BART  | 59.07/58.78/53.89 | 88.37/95.42/86.33 |
| BART-both | 59.21/58.93/54.01 | 88.52/95.63/87.06 |
| BART+RAC | 59.77/59.54/54.41 | 89.43/96.78/88.21 |

Table 5: ROUGE-1 score in two datasets.

A Datasets

We evaluated our model on two public Chinese dialogue summarization datasets: CSDS and MC. CSDS is a customer service dialogue dataset and MC is a medical inquiry summarization dataset. Each dialogue also includes a summary of the patient’s description and an analysis of the doctor’s suggestions. We also note them as a summary for users and agents. We use the official crawling script to acquire the dataset and follow the data split from (Lin et al., 2022). The statistical information of these two datasets are shown in Table 4.

B Implementation Details

We employ chinese-bart\(^7\) model to initialize our transformer-based seq2seq model. We also combine our proposed role-aware centrality mechanism into PGN and BERTAbs model. The training setting of them follows (Lin et al., 2022). BART, BART-both, and BART+RAC were all trained on four V100 32G devices and the maximum input length is 512, the learning rate is 1e-4, the total batch size is 64 and the epoch is 5.

C ROUGE-1 Score on Two Datasets

Limited by the page width, we put the results of ROUGE-1 in the appendix. From the results,\(^7\)https://huggingface.co/uer/bart-base-chinese-cluecorpussmall
our model still achieves the expected good results, which are consistent with the results in the main table.

D Case Study

We sample an example from the data set to show the final summary of the dialogue generated in the CSDS. We can see that BART tends to copy a large amount of tokens from the input contexts. Our BART+RAC can condense the input text and generate high quality summary.
Figure 4: An example from the CSDS dataset.