Abstract

Pre-trained models have brought remarkable success on the text summarization task. For dialogue summarization, the subdomain of text summarization, utterances are concatenated to flat text before being processed. As a result, existing summarization systems based on pre-trained models are unable to recognize the unique format of the speaker-utterance pair well in the dialogue. To investigate this issue, we conduct probing tests and manual analysis, and find that the powerful pre-trained model can not identify different speakers well in the conversation, which leads to various factual errors. Moreover, we propose three speaker-aware supervised contrastive learning (SCL) tasks: Token-level SCL, Turn-level SCL, and Global-level SCL. Comprehensive experiments demonstrate that our methods achieve significant performance improvement on two mainstream dialogue summarization datasets. According to detailed human evaluations, pre-trained models equipped with SCL tasks effectively generate summaries with better factual consistency.

1 Introduction

Dialogue summarization aims to condense the essential information in the dialogue into a brief text. Compared with text summarization, the conversations are semi-structured data and contain multiple participants who shall be distinguished (Gurevych and Strube, 2004; Feng et al., 2021a). Furthermore, dialogues combine features such as informal language, coreference, and repetition (Chen and Yang, 2020). All of these bring new challenges to the existing text summarization methods.

Although pre-trained models have achieved great success in text summarization (Liu and Lapata, 2019; Lewis et al., 2020; Zhang et al., 2020), how to properly utilize them in dialogues with a special speaker-utterance structure is still an obstacle. A line of previous work utilizes pre-trained models and deals with dialogue summarization as flat text. Chen and Yang (2020) segment dialogues into blocks from multiple semantic views and process them using BART. Feng et al. (2021b) use DialoGPT (Zhang et al., 2019) as an unsupervised annotator to help models understanding conversations. However, due to the gap with the pre-training object, the pre-trained models are hard to capture speaker information. To investigate these, we conduct a manual analysis on a popular dataset SAMSum (Gliwa et al., 2019) and discover that, even for the state-of-the-art model BART, 50% of the generated summaries contain factual errors for dialogues with multiple speakers. Among them, up to 68.4% are caused directly by speaker confusion and speaker missing (see Section 3.2). As shown in Table 1, the model’s inability to identify speakers results in serious factual inconsistencies.

Another tributary of previous work (Zhao et al., 2019; Liu and Chen, 2019; Zhu et al., 2020; Lei
et al., 2021) utilizes the hierarchical network instead of pre-trained models to leverage the dialogue’s structural information. However, how to explicitly model the information of speakers in pre-trained sequence-to-sequence (seq2seq) models remains unsolved. Zhu et al. (2020) introduces speaker embedding to distinguish speakers for meetings with fixed participants. However, in most cases, the number and identity of the participants in the conversations are unknown. Thus the trained embedding is not a general solution.

Intuitively, if the representation derived from the encoder has sufficient information to identify speakers, the decoder will produce superior summaries, especially for summaries that follow a pattern of *someone does something* as shown in Table 1. In this paper, we first conduct a probing experiment to show that the representation of the dialogue obtained from BART can not distinguish speakers well. To address this issue, we use contrastive learning to improve the alignment of the representation derived from the encoder, i.e., to make the encoder output diverse hidden states based on corresponding speakers. We propose three speaker-aware supervised contrastive learning tasks: *Token-level SCL*, *Turn-level SCL*, and *Global-level SCL*. By jointly training these tasks in the fine-tuning stage, we can substantially improve the model’s ability to identify different speakers and further understand the content of the whole dialogue. Comprehensive experiments and human evaluations on SAMSum and AMI (McCowan et al., 2005) reveal that our models generate summaries with higher ROUGE scores and better factual consistency. Our main contributions include (a) this is the first work to give a detailed investigation of the speaker identification problem in dialogue summarization, (b) proposing speaker-aware SCL tasks to address the problem, and evaluating our methods with the experimental and manual examination.

2 Method

2.1 Probing Test

To investigate how well pre-trained seq2seq models can distinguish speakers, we conduct a simple probing experiment on SAMSum, a widely-used dialogue summarization corpus. Concretely, we first encode the dialogue text with the BART (Lewis et al., 2020) encoder and randomly sample $K$ tokens to obtain their hidden states. Then, in pairs, we aggregate and feed these hidden states into an MLP to determine whether they are from the same speaker. We train the MLP layers on the SAMSum training set (in other words, we freeze the parameters of the BART encoder during the training stage) and then assess the classification accuracy on the test set. For vanilla BART, the accuracy is 58.1%. After fine-tuning BART with the summarization task on SAMSum before the probing test (parameters are not frozen), the accuracy is still only 60.2%1. Given the task’s simplicity, this result indicates that pre-trained seq2seq models can not identify speakers well from flat dialogue text.

2.2 Supervised Contrastive Learning Tasks

To address the above problem, inspired by the research about contrastive learning (Mikolov et al., 2013; Saunshi et al., 2019; He et al., 2020; Velickovic et al., 2019), we introduce SCL tasks during the fine-tuning stage to minimize the distance between representations of utterances from the same speaker and vice versa.

Formally, a dialogue $D = (t_1, t_2, \ldots, t_n)$ consists of $n$ turns, and each turn $t_i$ contains the utterance $u_i$ and the corresponding speaker $s_i$, that is, $t_i = (s_i, u_i)$. Firstly, we use Transformer-Encoder (Vaswani et al., 2017) to model the dialogue-level contextual representation of each token.

$$H = \text{Transformer-Encoder}(D),$$ (1)
where the input sequence is the concatenation of all turns. Then, we can generate the summary $\hat{y}$ with Transformer-Decoder. The generation loss $L_{gen}$ is cross-entropy loss between $\hat{y}$ and gold summary $y$.

**Incorporating Contrastive Loss** To enable the utterance representation to contain more speaker information, we incorporate three levels of contrastive losses.

Generally, let $(o_i, s_i)$ denote a sampled token or a sampled utterance and its associative speaker. The contrastive loss $L_{ctr}$ for the SCL task is calculated as follows:

$$L_+ = \sum_{i,j} s_i = s_j \log(\sigma(o_i \cdot o_j)),$$ (2)

$$L_- = \sum_{i,j} s_i \neq s_j \log(1 - \sigma(o_i \cdot o_j)),$$ (3)

$$L_{ctr} = L_+ + L_-,$$ (4)

where $\sigma$ is logistic function that measures the similarity between two representations and $o_i$ is the

1By jointly training the Global-level SCL task in fine-tuning stage, the accuracy reaches 77.9%.
Figure 1: Overview of our speak-aware SCL tasks. Token-level SCL and Turn-level SCL mean the model needs to discriminate whether two tokens/turns are from the same speaker. Global-level SCL let the model choose what the speaker might say in a particular turn when given all the utterances of this speaker. The representations are obtained by inputting the whole dialogue into the encoder.

2.2.1 Token-level SCL
The first task is the Token-level SCL which means the model distinguishes whether two tokens are from the same speaker. As illustrated in Figure 1(a), we randomly sample \( m \) token-speaker pairs \( T = \{(o_1, s_1), (o_2, s_2), \ldots, (o_m, s_m)\} \) from \( D \), where \( o_i \) is a token and \( s_i \) is the corresponding speaker. The hidden state of \( o_i \) obtained through the encoder is used to represent the \( i \)-th sample.

2.2.2 Turn-level SCL
Compared with Token-level SCL, we increase the granularity of the input to fuse the semantic information of the context. As shown in Figure 1(b), we randomly sample two turns from \( D \) and mask the speaker names in text, denoted as \((o_i, s_i)\) and \((o_j, s_j)\). Then we derive \( o_i \) by taking the mean pooling of the hidden states of all tokens in \( o_i \).

2.2.3 Global-level SCL
To maximize the mutual information between utterances of the same speaker (Linsker, 1988; Kong et al., 2019), we extend the Turn-level SCL task to Global-level SCL by introducing global information. Intuitively, we can understand the speaking style of a specific person from all the words he or she has said. Therefore, we provide the model with all the utterances of a certain speaker and let it choose what this speaker might say in a particular turn (described in Figure 1(c)). Concretely, we first mask all the speaker names and randomly sample a speaker whose utterances set \( \tilde{S}_i \) which has at least two elements. Among \( \tilde{S}_i \), we randomly choose a utterance \((o_i, s_i)\) as the positive sample, and randomly choose another utterance \((o_j, s_j)\) from \( D - \tilde{S}_i \) as the negative sample.

The final loss \( L = \lambda L_{ctr} + L_{gen} \) and \( \lambda \) is the weight coefficient to adjust the ratio of \( L_{ctr} \) and \( L_{gen} \) in the final loss \( L \). The model is supposed to maximize similarity among samples of the same speaker and vice versa while optimizing for the summary generation.

3 Experiment
In this section, we conduct experiments and human evaluations on the popular datasets SAMSUM and AMI. More descriptions of the datasets and the implementation details can be found in the Appendix.

3.1 Experimental Result and Analysis
We provide several latest strong seq2seq models as baselines, including PGNet (See et al., 2017), UniLM (Dong et al., 2019) and BART+DialoGPT (Feng et al., 2021b) in the first part of Table 2. Following previous settings (Gliwa et al., 2019; Feng et al., 2021a), we use pyrouge\(^2\) package for evaluation on SAMSUM and use pyrouge\(^3\) on AMI. Experimentally, our models obtain clear improvement on both two datasets compared to the BART baseline, and achieve the state-of-the-art result on SAMSUM.

\(^2\)https://pypi.org/project/py-rouge/
\(^3\)https://github.com/bheinzerling/pyrouge
Table 2: Results on the test sets of SAMSum and AMI, and "R" is short for "ROUGE".

<table>
<thead>
<tr>
<th>Model</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGNet (See et al., 2017)</td>
<td>40.08</td>
<td>15.28</td>
<td>36.63</td>
<td>42.60</td>
<td>14.01</td>
<td>22.62</td>
</tr>
<tr>
<td>UniLM (Dong et al., 2019; Zhu et al., 2021)</td>
<td>50.00</td>
<td>26.03</td>
<td>42.34</td>
<td>50.61</td>
<td>19.33</td>
<td>25.06</td>
</tr>
<tr>
<td>Multi-view BART (Chen and Yang, 2020)</td>
<td>53.42</td>
<td>27.98</td>
<td>49.97</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>BART+DialoGPT (Feng et al., 2021b)</td>
<td>53.70</td>
<td>28.79</td>
<td>50.81</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>PGN+DialoGPT (Feng et al., 2021b)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>50.91</td>
<td>17.75</td>
<td>24.59</td>
</tr>
<tr>
<td>BART</td>
<td>53.01</td>
<td>28.05</td>
<td>49.89</td>
<td>50.67</td>
<td>17.18</td>
<td>24.96</td>
</tr>
<tr>
<td>BART + Token-level SCL task</td>
<td>53.85</td>
<td>29.21</td>
<td>50.94</td>
<td>51.03</td>
<td>17.23</td>
<td>25.21</td>
</tr>
<tr>
<td>BART + Turn-level SCL task</td>
<td>54.12</td>
<td>29.53</td>
<td>51.10</td>
<td>51.15</td>
<td>17.85</td>
<td>25.45</td>
</tr>
<tr>
<td>BART + Global-level SCL task</td>
<td>54.22</td>
<td>29.87</td>
<td>51.35</td>
<td>51.40</td>
<td>17.81</td>
<td>25.30</td>
</tr>
</tbody>
</table>

3.2 Human Evaluation

We also conduct human evaluations to investigate if our method leads to fewer factual errors. Automatic metrics like FACTCC (Kryściński et al., 2019) are not used since the neural-model-based metrics perform poorly in dialogue data due to the significant domain gap. And most of the factual errors in the dialogue summarization are caused by misidentification of the speaker, which can not be reflected by automatic metrics. Here we use BART and BART with the Global-level SCL task for comparison.

Error Types: Firstly, we divide the factual errors into three categories manually: (a) Speaker Confusion: Model confuses speakers participating in a specific event; (b) Speaker Missing: A speaker is mentioned in the gold summary, while the model hits the event but misses this speaker. (c) Semantic Error: Errors caused by a misunderstanding of semantics, and they are not directly related to any speakers. More cases about the error types can be found in Appendix.

Table 3: The number of factual errors for the baseline model and our model on the SAMSum dataset.

<table>
<thead>
<tr>
<th>Model</th>
<th>BART</th>
<th>BART + Global SCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speaker Confusion</td>
<td>8.6</td>
<td>4.1</td>
</tr>
<tr>
<td>Speaker Missing</td>
<td>26.0</td>
<td>20.2</td>
</tr>
<tr>
<td>Semantic Errors</td>
<td>18.7</td>
<td>16.8</td>
</tr>
</tbody>
</table>

4 Conclusion

In this paper, we focus on the speaker identification problem in the dialogue summarization task. Through the probing test and manual analysis, we find that the existing pre-trained model cannot identify different speakers well in the conversation, leading to factual errors. Therefore, we propose three speaker-aware SCL tasks to address this problem. Experimental results and human evaluations illustrate the effectiveness of our methods.

4 Please note that there may be multiple types of errors in a single sample.
5 Ethical Considerations

For human evaluation in section 3.2, we recruited two annotators to see if there are any factual inconsistencies in generated summaries. The generators of all summaries are hidden from the annotators to avoid any subjective bias in our proposed methods. For the SAMsum dataset, we give more priority to dialogues with more speakers and adopt a random strategy when the numbers of speakers are same. Furthermore, both annotators were compensated fairly.

References


We apply our methods on the large version of BART and evaluate our model on SAMSum and AMI datasets using ROUGE score (Lin and Och, 2004). SAMSum consists of 16,369 samples with an average of 2.4 participants and 83.9 words. AMI consists of 137 meeting records of four fixed speakers, which have 4,757 words on average. Due to the limitation of our computing resources, all our inputs are truncated to 1,024 tokens. We use the same split as Glvéa et al. (2019) and Zhu et al. (2020) for SAMSum and AMI, respectively.

### B Implementation Details

<table>
<thead>
<tr>
<th>Hyperparameters</th>
<th>SAMSum</th>
<th>AMI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Size</td>
<td>8</td>
<td>24</td>
</tr>
<tr>
<td>Total Steps</td>
<td>10,000</td>
<td>600</td>
</tr>
<tr>
<td>Eval Steps</td>
<td>1,000</td>
<td>20</td>
</tr>
<tr>
<td>Learning Rate</td>
<td>[2e-5,3e-5]</td>
<td>[2e-5,3e-5]</td>
</tr>
<tr>
<td>Label Smoothing Factor</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>Warm-up Type</td>
<td>linear</td>
<td>linear</td>
</tr>
<tr>
<td>Warm-up Steps</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>Max Target Length</td>
<td>128</td>
<td>300</td>
</tr>
</tbody>
</table>

Table 4: Hyperparameters we used for fine-tuning BART on SAMSum and AMI.

Some of our hyperparameters are listed in Table 4. Other hyperparameters are the same as the default of facebook/bart-large of transformers. The weight coefficient factor \( \lambda \) is searched from \{0.01, 0.001\}. It takes up to 2 hours for one run on SAMSum or AMI using one GeForce RTX 3090.

We use the validation set to select the best checkpoint, and evaluate the checkpoint on the test set.

### C Human Evaluation on AMI

<table>
<thead>
<tr>
<th>Model</th>
<th>BART</th>
<th>BART + Global SCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speaker Confusion</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Speaker Missing</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Semantic Errors</td>
<td>18</td>
<td>14</td>
</tr>
</tbody>
</table>

Table 5: The number of factual errors for the baseline model and our model on the AMI dataset.

For the AMI dataset, we evaluate all 20 samples of the test set, and the result is shown in Figure 5. All speakers in AMI are fixed, so the two models rarely confuse them. However, as a result of the truncation, some utterances are only left a small part and become easily overlooked. With our method, BART can better identify the corresponding speakers. Due to the loss of input information, both models have a large number of semantic errors. Compared to the baseline model, our model decreases the number by 22.2%.

---

5https://github.com/huggingface/transformers
D Case Study

In order to better illustrate the three types of errors mentioned in Section 3.2, we provide more cases here. An example of confusing speakers is shown in Table 1 of the main paper. Examples of missing speakers and semantic errors are shown in Table 6.

---

Dialogue Text 1

Ann: Congratulations!! Ann: You did great, both of you! Sue: Thanks, Ann Julie: I’m glad it’s over! Julie: That’s so cute of you, girl! Ann: Let’s have a little celebration tonight! Sue: I’m in Julie: me too!! aww

Gold Summary 1

Ann, Sue and Julie did a great job and they will have a little celebration tonight.

Baseline Summary 1 by BART

Sue and Julie are going to celebrate their success tonight.

Our Summary 1

Ann, Sue and Julie are celebrating their wins.

Dialogue Text 2

Sarah: omg Laura! sorry you didn’t get any replies!!! Did you manage? Laura: hahaha! Awkssss... no worries, I solved it Sarah: awkward silence <crickets> Laura: hahaha no it’s all good really!! Raf: Laura, I’m so sorry!!! been so swamped, totally forgot to text you back! where are you?? Sarah: Exotic little island called Linate :D Laura: Sarah which hotel are you at?? I’m here too!!

Gold Summary 2

Neither Raf nor Sarah remembered to reply to Laura but she managed anyway. Both Sarah and Laura are in Linate.

Baseline Summary 2 by BART

Laura didn’t get any replies to Sarah’s messages.Laura is on an island called Linate. Laura and Sarah are staying at the same hotel.

Our Summary 2

Laura didn’t get any replies from Sarah and Raf. Sarah and Laura are on an exotic little island called Linate.

---

Table 6: Sample 1 is an example about the speaker missing error. The summary generated by BART misses Ann. The dialogue sample is from the SAMSum dataset. Sample 2 is an example about the semantic error and the speaker missing error. The summary generated by BART misses Raf (Speaker Missing Error), and makes it out of thin air that Sarah and Laura are staying at the same hotel (Semantic Error). All samples are from the SAMsum dataset.