A Multi-Task Learning Approach for Summarization of Dialogues

Saprativa Bhattacharjee
Department of Information Technology
Government Polytechnic Daman
India
saprativa.bhatt@gov.in

Kartik Shinde
Department of Civil Engineering
Indian Institute of Technology Patna
India
kartik_1901ce16@iitp.ac.in

Tirthankar Ghosal
Charles University
MFF, ÚFAL
Czech Republic
ghosal@ufal.mff.cuni.cz

Asif Ekbal
Department of Computer Science and Engineering
Indian Institute of Technology Patna
India
asif@iitp.ac.in

Abstract

We describe our multi-task learning based approach for summarization of real-life dialogues as part of the DialogSum Challenge shared task at INLG 2022. Our approach intends to improve the main task of abstractive summarization of dialogues through the auxiliary tasks of extractive summarization, novelty detection and language modeling. We conduct extensive experimentation with different combinations of tasks and compare the results. In addition, we also incorporate the topic information provided with the dataset to perform topic-aware summarization. We report the results of automatic evaluation of the generated summaries in terms of ROUGE and BERTScore.

1 Introduction

Much of the early works on summarization devoted attention to monologues such as news articles (Nallapati et al., 2016; Narayan et al., 2018), patents (Sharma et al., 2019), Wikipedia articles (Liu et al., 2018; Cohen et al., 2021), scientific research papers (Cohan et al., 2018), Government reports (Huang et al., 2021) and even court judgements (Gao et al., 2019). But more recently, the focus of the summarization community has started shifting from monologues to dialogues largely owing to the rising popularity of chatbots, personal assistants, instant messaging platforms and online meetings. While monologues are characterised by the fact that they are authored by a single person, a dialogue involves the utterances of more than one participant (which alone can make them inherently more difficult to summarize). However, the available dialogue summarization datasets (Gliwa et al., 2019; Zhu et al., 2021; Feigenblat et al., 2021) are fewer in number, limited in scale, domain-specific and sometimes even extremely noisy and semi-structured (Carletta et al., 2005; Janin et al., 2003) as compared to the datasets available for monologue texts.

To mitigate these issues a high-quality large-scale dialogue summarization dataset named DialogSum was released by Chen et al. (2021a). The dataset consists of a wide variety of task-oriented dialogues from daily-life conversations. One sample dialogue and its corresponding summary from DialogSum’s training set is presented in Figure 1, which is a conversation between a doctor and his patient on the topic of getting a check-up. To further encourage research in dialogue summarization, the authors proposed a shared task named DialogSum Challenge (Chen et al., 2021b) as part of INLG 2022, and in this article, we describe our submission to the shared task as Team IITP-CUNI.

Specifically, we attempt to tackle the problem of abstractive dialogue summarization through the use of a multi-task learning model (Ruder, 2017; Crawshaw, 2020; Vandenhende et al., 2020) based on Transformers (Vaswani et al., 2017). We intend to improve the main task of abstractive summarization of the dialogues through the auxiliary tasks...
of extractive summarization, novelty detection and language modeling. Additionally, we also explore the usefulness of topic-aware summarization, as in the DialogSum dataset, topics are provided along with the summaries (see Figures 1 and 2).

The rest of the paper is organised as follows. Related work is presented in Section 2. The DialogSum Challenge is described in details in Section 3. Section 4 presents our system. Results and discussion are in Section 5. Finally, the conclusion is drawn in Section 6.

2 Related Work

In this section, we discuss some of the most recent works on dialogue summarization and multi-task learning strategies for abstractive summarization. For long dialogue summarization, Zhong et al. (2021) proposed a window-based pre-training strategy using five different types of dialogue-related noise – speaker mask, turn splitting, turn merging, text infilling and turn permutation. At first, the window is corrupted with noise, and then the model is tasked with de-noising and reconstructing the window. On the other hand, Zhang et al. (2022) utilize a multi-stage approach for dealing with long dialogues. In the preliminary stages, they segment the input and produce coarse summaries, while in the final stage, the coarse summaries are used to generate the final fine-grained summary. Zhang et al. (2021) studied the effectiveness of different strategies to deal with long dialogues and concluded that a retrieve-then-summarize pipeline model works better in comparison to Longformer (Beltagy et al., 2020) or HMNet (Zhu et al., 2020). However, in the case of DialogSum, as the input data is well within the limit of the popular pre-trained Transformer models such as BART (Lewis et al., 2020), we are not faced with any such issues. Moreover, Chen et al. (2021a) have shown that the larger version of BART performs better than others on DialogSum. We start our investigation with this strong baseline.

Another direction of work has been the incorporation of topic information to further improve the abstractive dialogue summarization. In this direction, Zou et al. (2021) proposed a novel topic-augmented two-stage dialogue summarizer (TDS) along with a saliency-aware neural topic model (SATM) to perform topic-aware summarization of customer service dialogues. Qi et al. (2021) fused the topic segmentation embedding along with positional embedding in the utterance-level encoder input of a hierarchical Transformer architecture. To capture the topic information of dialogues Liu et al. (2021) came up with two contrastive learning strategies, namely coherence detection and sub-summary generation. And all of them reported performance benefits of taking topic information into account while performing abstractive summarization. We too explore the topic-aware summarization as the DialogSum dataset provides topic information along with the summaries.

A slightly different but closely related task that deserves mention is that of automatic minuting of meeting transcripts. The first shared task on Automatic Minuting (AutoMin) (Ghosal et al., 2021a) at Interspeech 2021 and the SIGDial 2021 Special Session on Summarization of Dialogues and Multi-Party Meetings (SummDial) (Ghosal et al., 2021b) brought out a plethora of interesting works targeting the task such as the attempt to use BART for generation of readable minutes (Shinde et al.,
Singh et al. (2021) present an empirical analysis of the state-of-the-art summarization models for the task of generating meeting minutes and arrive at the conclusion that they are far from being satisfactory. A novel dataset of meetings in English and Czech (Nedoluzhko et al., 2022) is also being released to further encourage the research community to take up the challenging task.

Lee et al. (2021) claim to be the first ones to have applied multi-task learning to dialogue summarization task. Leveraging Part-of-Speech (PoS) information, they constructed a syntax-aware dialogue summarization model on SAMSum corpus (Gliwa et al., 2019). The main intuition behind their approach is that different speaker roles are characterised by different syntactic structures (voiceprints), which could be captured via POS information. More recently, for low-resource datasets Magooda et al. (2021) experimented with several combinations of auxiliary tasks for abstractive summarization in a multi-task setting. They concluded that a certain combination of tasks indeed improved the abstractive summarization results across different datasets and models. Prior to these, in the multi-task setting, the primary task of abstractive summarization has been combined and experimented with several other auxiliary tasks such as entailment generation (Pasunuru et al., 2017); question generation and entailment generation (Guo et al., 2018); extractive summarization (Chen et al., 2019); text categorization and syntax labeling (Lu et al., 2019); dialogue act classification and extractive summarization (Manakul et al., 2020); keyword extraction and key-sentence extraction (Xu et al., 2020).

Very recently, Chen et al. (2022) formulated the five different tasks of dialogue understanding (DU) as a unified generation task. These tasks include dialogue summarization, dialogue completion, dialogue state tracking, slot filling and intent detection. Then they experimented with eight different multi-task training strategies and concluded that their proposed method achieves superior performance on both few-shot as well as zero-shot settings. These encouraging results of the multi-task learning strategies on abstractive summarization motivated us to apply the same to the DialogSum Challenge.

### 3 DialogSum Challenge

In this section, we give a brief overview of the DialogSum Challenge by first describing the dataset and then going through the task description.

#### 3.1 Dataset Description

The DialogSum dataset consists of a total of 13,460 dialogue-summary pairs, out of which 12,460 (92.6%) are in the training set, 500 (3.7%) in the development set and 500 (3.7%) more in the test set, as depicted in Figure 3. The dialogue data has been collected from multiple sources, namely 58.22% from DailyDialogue dataset (Li et al., 2017), 16.94% from DREAM dataset (Sun et al., 2017).
Table 1: DialogSum dataset split statistics. ‘#Dialogues’ contains absolute values while rest of the columns report average values. ‘Len.’ stands for Length. ‘hidden’ is the hidden test set for which only the dialogues and topics have been released publicly and hence the Summary Length and %-Compression details are not available.

<table>
<thead>
<tr>
<th>Split</th>
<th>#Dialogues</th>
<th>#Turns</th>
<th>Turn Len.</th>
<th>Dialogue Len.</th>
<th>Summary Len.</th>
<th>%-Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>train</td>
<td>12460</td>
<td>9.49</td>
<td>20.10</td>
<td>191.37</td>
<td>29.36</td>
<td>83.72</td>
</tr>
<tr>
<td>dev</td>
<td>500</td>
<td>9.38</td>
<td>20.17</td>
<td>188.89</td>
<td>27.21</td>
<td>84.74</td>
</tr>
<tr>
<td>test</td>
<td>500</td>
<td>9.71</td>
<td>20.04</td>
<td>196.12</td>
<td>23.76</td>
<td>86.70</td>
</tr>
<tr>
<td>hidden</td>
<td>100</td>
<td>10.88</td>
<td>19.03</td>
<td>209.42</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Some statistics of interest for each split of the dataset are presented in Table 1. Although the training, development and test sets are quite similar in terms of the average number of turns and the average turn length, the test set average dialogue length is larger while the average summary length is smaller than the other two sets. This also gets reflected in the test set’s marginally higher compression ratio. Moreover, the average dialogue length of the hidden test set is higher than all other sets, but this may be attributed to the smaller size of the hidden set. In training and development sets, for each dialogue, one human written summary is provided. Figure 1 shows an example dialogue-summary pair from the training set. In addition to the summary, the human annotators also provide the topic information. On the other hand, for each dialogue in the test set, three human written reference summaries are provided. Figure 2 shows an example dialogue from the test set and its three reference summaries. For each reference summary, its corresponding topic is also provided.

In addition to the above, the organizers have also released a hidden test set consisting of 100 dialogues. Only the dialogues and topic information are provided for this hidden set, while the summaries have not been made public. The organizers will use this set for evaluation of the submitted models.

3.2 Task Description

The shared task participants need to design a model which will take as input the dialogue text and produce the corresponding abstractive summary. For automatic evaluation, each system-generated summary will be evaluated against the three human written reference summaries and the average ROUGE scores (Lin, 2004) and BERTScore (Zhang et al., 2020) will be used to determine the position on the DialogSum Challenge’s leaderboard. Out of these two metrics, ROUGE (R1, R2 and RL) will be used as the primary metric, while BERTScore will be used as a supplementary metric. Additionally, the generated summaries will also be evaluated against the human-written summaries of the hidden test set. The lowest, highest and averaged scores will be reported for both the multi-reference test sets.

For human evaluation, the submitted summaries will be judged on the following parameters: (i) fluency, consistency, relevance and coherence; (ii) co-reference information; (iii) intent identification; (iv) discourse relation; and (v) objective description. For more details about these parameters, we would like to refer the readers to the shared task paper (Chen et al., 2021b).

4 Our System

We employ a multi-task learning approach for the DialogSum Challenge. In multi-task learning, a machine learning model is trained simultaneously on more than one related task (Crawshaw, 2020).
Usually, there is a main task and one or more auxiliary tasks. In our case, the main task is abstractive summarization and the auxiliary tasks are extractive summarization, novelty detection and language modeling. There are many variants of multi-task learning. In this work, we employ a hard parameter sharing (Ruder, 2017) Transformers-based architecture in which all tasks share the same encoder layers but have task-specific decoder and/or LM head(s). The multi-task model architecture is depicted in Figure 4. It consists of a single BART encoder which is shared amongst all the tasks. The BART decoder is used for the main task of abstractive summarization, while task-specific heads are used for each of the respective auxiliary tasks. We now describe each of the tasks of our model one-by-one:

Abstractive Summarization (AS): For the main task of abstractive summarization, the transcripts are given as input to the BART encoder and the abstractive summaries are obtained as output from the BART decoder. This is a sequence-to-sequence task accomplished with the encoder-decoder architecture. In cases where we want to run only the single task for establishing the baseline, only this task is undertaken while keeping all other auxiliary tasks inactive through the training parameters.

Extractive Summarization (ES): The task of extractive summarization is formulated as a classification task where the goal is to classify a given sentence as either belonging to or not belonging to the extractive summary. The inputs are given in the format [CLS] SW1, SW2, ..., SWn [SEP] CW1, CW2, ..., CWm. Here, [CLS] is the start token, [SEP] is the separator token, SW1...SWn is the sentence to be classified as belonging to the extractive summary or not and CW1...CWm is the context around the sentence SW1...SWn. The sentence and the context around it are chosen in such a way that the maximum combined length does not exceed 1024 tokens.

Novelty Detection (ND): Novelty detection in NLP refers to the identification of novel text, i.e., text containing new information (Ghosal et al., 2022). This task is also formulated as a classification task. For this task, we use data from three different sources: (i) Quora Question Pair (QQP) dataset1 consisting of more than 400 thousand question pairs. Each such pair is annotated with a binary value which indicates whether or not the questions in the pair are duplicates of each other. (ii) Microsoft Research Paraphrase Corpus (MRPC) (Dolan and Brockett, 2005) is a corpus consisting of 5,801 sentence pairs from news articles where each pair is annotated by humans as being either a paraphrase or not and (iii) data created from the three reference summaries given in the public test set of DialogSum. We assume that the three reference summaries are paraphrases (non-novel) of each other. Since there are 500 dialogues, each with three reference summaries, we obtain 1,500 non-novel samples. We also extract a similar number of novel samples by taking summaries from two different dialogues, as shown in Table 2. The input is given in the form [CLS] source text [SEP] target text, and the task of the model is to classify the pair as either novel or non-novel (duplicates).

Language Modeling (LM): We perform masked language modeling on the gold summaries from the training set as per the training strategy adopted by Devlin et al. (2019). For this, 15% of the input tokens are masked and out of this, 80% are replaced by special tokens, 10% with random words and the remaining 10% are left unchanged.

1https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ref. Summary 1</td>
<td>Ref. Summary 2</td>
<td>0</td>
</tr>
<tr>
<td>Ref. Summary 2</td>
<td>Ref. Summary 3</td>
<td>0</td>
</tr>
<tr>
<td>Ref. Summary 1</td>
<td>Ref. Summary 3</td>
<td>0</td>
</tr>
<tr>
<td>Ref. Summary (Dn)</td>
<td>Ref. Summary (Dm)</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2: Novelty dataset created from the three reference summaries provided in the public test set of DialogSum. Ref. Summary (Dn) & Ref. Summary (Dm) denotes reference summaries from different dialogues.
5 Results and Discussion

In this section, we first describe the experimental setup used and then present the results. Finally, we analyse the summaries generated by our best-performing model.

5.1 Experimental Setup

We run all the experiments on two NVIDIA A100-PCIE-40GB GPUs using a batch size of 4 for both training and evaluation and mostly use the default values for hyperparameters. The BART model is initialized with facebook/bart-large and then finetuned using task-specific datasets. Mixed-precision training using fp16 is utilized for faster training and lesser memory footprint. We make use of the summarization script released by Hugging Face and the multi-task learning ideas introduced by Magooda et al. (2021). The ROUGE evaluations are done using py-rouge and BERTScore evaluations using bert_score as suggested by the organizers of DialogSum Challenge.

5.2 Results

We provide all the results from our experiments in Table 3. The reported performance is the average of the scores of system-generated summaries with respect to the three reference summaries provided in the public test set. We consider the single-task setting where only abstractive summarization (AS) is done without any auxiliary tasks as the baseline. For the topic-aware abstractive summarization (AS[T]), we supply the topic information by prepending it to the input dialogue to the BART encoder as [CLS] TOPIC [SEP] Dialogue. We observe a marginal improvement in the scores using this strategy.

In the multi-task setting, we experiment with different combinations of tasks as well as data. The best ROUGE scores are obtained when abstractive summarization is done along with extractive summarization (ES), while the best BERTScore is obtained when abstractive summarization is combined with novelty detection (ND). Since extractive summaries were not provided with the Dialgosum dataset, we used bert-extractive-summarizer to obtain the same. Alongside the newly created extractive data from DialogSum, we also experiment with the extractive summary data from AMI (Carletta et al., 2005). Results show that the model trained with auxiliary task of extractive summarization (from AMI) outperforms all others. To explain such a performance, we analyze the outputs and test other configurations with both extractive datasets. However, in our observation, there are no apparent reasons for the model to perform in such a manner on AMI data. Finally, we account this to the fact that AMI is a dataset of meeting transcript and summaries, in which the information is widely dispersed throughout the discourse of the transcript, which have a lot of redundancies. While, dialogues from the DialogSum dataset are relatively shorter, with lesser redundant texts. Moreover, most of the lines from these dialogues (even those that are coherent with parts of summary), have a generic fashion of day-to-day speech. Hence, the BART model learns better from the extractive data from AMI.

5.3 Analysis

We take our best performing model and manually analyse the summaries generated by it. Figure 5 and Figure 6 present the worst three and best summaries according to the scores.
three summaries generated by the model in terms of ROUGE-1, respectively. It is to be kept in mind that the ROUGE scores reported are the average of the generated summary with respect to the three reference summaries. Let us first consider the case of the three worst summaries shown in Figure 5. In the case of the first system-generated summary, we can see that it is longer than each one of the three reference summaries and the content is quite different. In the second case, our model is unable to figure out that Person1 “thinks” she met/knows Person2. Rather the model generates the phrase “finds out”. Moreover, the last line, “Person2 has to go” is totally unnecessary for the summary. In the case of the third summary, although the system-generated summary conveys the same message as the reference summaries, yet the same is not reflected in terms of ROUGE-1 mainly because of the different set of unigrams used.

Let us now consider the best three summaries generated by our model as shown in Figure 6. In all three cases, it can be seen that the generated summary matches almost exactly to one of the three reference summaries. The second system-generated summary matches word-to-word with its first reference summary, while the first and third system-generated summaries differ with their respective best matches on only a single word. The

<table>
<thead>
<tr>
<th>R1</th>
<th>Model Generated Summary</th>
<th>Reference Summaries</th>
</tr>
</thead>
</table>
| 0.19 | Person1 warns Person2 Person2 will be arrested if Person2 calls Person1 again. | Person1 is angry about the crank calls.  
Person1 gets a crank call and is angry about it.  
Person1 receives a phone call but no one speaks. |
| 0.21 | Person1 and Person2 meet each other for the first time. Person1 finds out they have met before. Person2 has to go. | Person1 thinks that she knows Person2 somewhere, but Person2 denies it.  
Person1 thinks she has met Person2 somewhere, but Person2 thinks it’s a mistake.  
Person1 keeps asking where Person2’s from because she thinks she knows Person2 but Person2 denies it. |
| 0.21 | Person1 tells Tony that everything has been going wrong lately in the toy department of the shopping center. Person1 thinks Christmas does not mean much now except more work and more headaches. | Person1 complains to Tony that Christmas has made Person1 busier.  
Person1 works as a toy salesperson and feels so tired recently because Christmas is coming, and everyone’s shopping for presents.  
Person1 thinks selling gifts for kids is such an unpleasant job before Christmas. |

<table>
<thead>
<tr>
<th>R1</th>
<th>Model Generated Summary</th>
<th>Reference Summaries</th>
</tr>
</thead>
</table>
| 0.89 | Person1 congratulates Mr. Stuart on his winning the city marathon. | Person1 congratulates Mr. Stuart on winning a marathon.  
Person1 congratulates Mr. Stuart on winning the city marathon.  
Person1 congratulates Mr. Stuart on winning the city marathon. |
| 0.83 | Mr. Lee gives Mrs. Word a lift home. | Mr. Lee gives Mrs. Word a lift home.  
Mr. Lee gives Mrs. Word a lift home on a rainy night.  
Mr. Lee offers to give Mrs. Word a lift home on a terrible night. |
| 0.81 | Person2 shows Person1 the way to the central department stall and the national bank. | Person2 shows Person1 the way to the central department stall and the national bank.  
Person2 shows Person1 the way to the central department stall and the national bank.  
Person1 asks Person2 the way to the central department stall and the national bank. |
higher score of the first summary can be attributed to the fact that two out of the three reference summaries in this case turn out to be exactly the same, which takes the average score up.

6 Conclusion

In this paper, we describe our submission to the shared task on dialogue summarization named DialogSum Challenge at INLG 2022. DialogSum consists of 13,460 real-life scenario dialogues. We employ a multi-task learning approach for the task and achieve considerable improvement over the single-task baseline. Our best performing model is the multi-task combination of abstractive summarization as the main task and extractive summarization as the auxiliary task. We also incorporate the topic information supplied alongside the summaries to gain marginal improvement in performance over the baseline. In future work, we would like to experiment with other tasks to find the optimal combination. We would also like to explore methods other than multi-task learning for improving the abstractive summarization of dialogues.

Acknowledgements

Tirthankar Ghosal would like to acknowledge the support from the grant 19-26934X (NEUREM3) of the Czech Science Foundation and the European Union’s Horizon 2020 Research and Innovation Programme under Grant Agreement No 825460 for the project European Live Translator (ELITR).

References


