Calvados at MEDIQA-Chat 2023: Improving Clinical Note Generation with Multi-Task Instruction Finetuning

Kirill Milintsevich[1,2] and **Navneet Agarwal**[1]
[1]GREYC, ENSICAEN, Université de Caen Normandie, France
[2]Institute of Computer Science, University of Tartu, Estonia
{first_name}.{last_name}@unicaen.fr

Abstract

This paper presents our system for the MEDIQA-Chat 2023 shared task on medical conversation summarization. Our approach involves finetuning a LongT5 model on multiple tasks simultaneously, which we demonstrate improves the model's overall performance while reducing the number of factual errors and hallucinations in the generated summary. Furthermore, we investigated the effect of augmenting the data with in-text annotations from a clinical named entity recognition model, finding that this approach decreased summarization quality. Lastly, we explore using different text generation strategies for medical note generation based on the length of the note. Our findings suggest that the application of our proposed approach can be beneficial for improving the accuracy and effectiveness of medical conversation summarization.

1 Introduction

Medical conversations between doctors and patients play a crucial role in healthcare. The conversations help the doctors understand the patients' conditions, diagnose, and provide appropriate treatments. However, these conversations can be lengthy and complicated, leading to difficulties in summarizing the essential information for medical records. Automatic summarization of medical conversations can help reduce the workload of medical practitioners and improve the quality of patient care. Therefore, there is a growing interest in developing natural language processing (NLP) techniques for summarizing medical conversations.

Large Language Models (LLMs) such as BART (Lewis et al., 2020), GPT-3 (Brown et al., 2020) and T5 (Raffel et al., 2020) have shown to be powerful in various language generation tasks, including summarization. However, they are known to suffer from hallucinating, i.e. including the facts that are false in the output or corrupting the facts in the input (Maynez et al., 2020).

This paper proposes a method for summarizing medical conversations using T5-based models. We finetune two T5-based models on two datasets from the MEDIQA-Chat 2023 shared task. The first dataset consists of short transcriptions of doctor-patient conversations, while the second dataset contains full patient-doctor encounter transcriptions. Our method uses text-to-text modelling, representing the input as a dialogue and the output as a conversation summary. To tackle the hallucination problem, we modify the data using a clinical named entity recognition model to tag the entities in the input and output sequences. We suppose this enables the models to learn better to copy the relevant entities from the conversation to the generated summary. Additionally, we finetuned a single model on multiple tasks to improve its robustness.

Our results showed that finetuning a single model on multiple tasks improved the summary generation quality and reduced hallucination. On the other hand, introducing extra tags to the inputs worsened the summarization quality.

The remainder of this paper is organized as follows. In Section 2, we provide a brief overview of related work in the field of text and dialogue summarization and highlight the limitations of existing approaches. Section 3 briefly describes the data used for the MEDIQA-Chat 2023 shared task. In Section 4, we describe our proposed method in detail. In Sections 5 and 6, we present the experimental setup and results, followed by a thorough analysis of the effectiveness of our approach. Finally, we conclude the paper and outline future directions in Section 7.

2 Related Works

Several generative language models, such as BART (Lewis et al., 2020), GPT-3 (Brown et al., 2020), PEGASUS (Zhang et al., 2019a), and T5 (Raffel et al., 2020) are used for abstractive text summarization. All these models are

Proceedings of the 5th Clinical Natural Language Processing Workshop, pages 529–535

based on the Transformer encoder-decoder architecture (Vaswani et al., 2017).

Significant progress has been made in training the generative language models using multi-task setting by giving the natural language instructions (Brown et al., 2020; Ouyang et al., 2022; Chung et al., 2022). While these models are already powerful for zero-shot and few-shot settings, finetuning them on specific data can significantly improve the performance for different tasks.

In terms of dialogue summarization, common-domain datasets such as DialogSUM (Chen et al., 2021b), MediaSum (Zhu et al., 2021), SAMSum (Gliwa et al., 2019) have been used for training the generative language models. Medical conversations summarization has been generally understudied with the recent efforts by Kazi and Kahanda (2019), Yim and Yetisgen (2021), and Michalopoulos et al. (2022).

Since full conversations are generally lengthy and extend beyond a common input length limit of most of the pre-trained models, several efforts have been made to modify the Transformer self-attention mechanism to encode long texts (Beltagy et al., 2020; Guo et al., 2022) efficiently.

Finally, the problem of the faithfulness of the automatically generated text is especially crucial for medical domain (Maynez et al., 2020; Chen et al., 2021a)

3 Data

We work on two datasets that are a part of MEDIQA-Chat 2023 shared task (Ben Abacha et al., 2023a). The first dataset used for Task A consists of short transcriptions of doctor-patient conversations followed by one of 20 possible classification labels and a short note from a doctor summarizing the conversation (Ben Abacha et al., 2023b). The second dataset used for Task B contains full patient-doctor encounter transcriptions accompanied by a full clinical note based on the encounter (Yim et al., 2023). Table 1 shows a short summary of the datasets.

Task	#samples			Average length	
	Train	Dev	Test	Dialogue	Note
A	1201	100	200	150	59
B	67	20	40	1904	666

Table 1: Summary of the datasets. The average length is reported in tokens.

4 Method

We finetune two T5-based models: FLAN-T5 Base model (Chung et al., 2022) on Task A data and LongT5 Base model (Guo et al., 2022) on both Task A and Task B data. Since both inputs and outputs for the Task B data are much longer than Flan-T5 maximum context window (512 tokens), we only finetune it on the Task A data.

4.1 T5 Model Architecture

The original T5 model mostly follows the encoder-decoder Transformer architecture (Vaswani et al., 2017) with the following modifications: the authors use a simplified version of layer normalization with no additive bias, which is placed outside the residual path as well as a different version of the relative positional embeddings (Raffel et al., 2020). Raffel et al. (2020) train the T5 model on various downstream tasks, including text summarization, classification, question answering and machine translation. The data is annotated in such a way that each task is treated as a text-to-text problem with the input prefixed by a verbal task description.

Later, Chung et al. (2022) present the FLAN-T5 model, which is architecturally identical to the original T5 model but is finetuned for more tasks, such as chain-of-thought task, and uses different instruction templates to prefix the input data. In another work, Guo et al. (2022) proposes the LongT5 model, which uses Transient Global (TGlobal) Attention to encode long sequences efficiently. TGlobal attention is a combination of a sparse sliding-window local attention and global attention which adds additional dynamically constructed global, or transient, tokens to the final attention matrix.

4.2 Our Approach

Similar to the T5 finetuning approach, we represented the data for Tasks A and B as a text-to-text problem. For Task A, we prefixed the input dialogue with *"summarize short: "* and represented the output as a concatenation of the string representation of the section header prefixed with *"Section Header: "* and the section note prefixed with *"Section Text: "*. For Task B, the output note was split into four divisions: objective exam, subjective, objective results, assessment and plan. The input

Model	Task A	Task B	Tagged
TASKA-ONLY	✓	✗	✗
TASKB-ONLY	✗	✓	✗
TASKAB	✓	✓	✗
TASKAB-TAG	✓	✓	✓

Table 2: Description of the models used in the experiments. ✓in **Task A** and **Task B** columns mean that the data from Task A or B was used during finetuning. **Tagged** column corresponds to the usage of the data tagging technique.

dialogue was prefixed with *"summarize {division}: "* and the output note was prefixed with *"division note: "*, where *{division}* is a placeholder for the corresponding division name. We split the Task B output notes into smaller parts to equalize the length with the Task A notes.

To modify the data, we use Stanza's (Qi et al., 2020) clinical MIMIC-i2b2 named entity recognition (NER) model (Zhang et al., 2021) to tag inputs and outputs of both Task A and Task B data. This model has `PROBLEM`, `TEST`, and `TREATMENT` tags, all of which are commonly present in clinical data. To modify the data, we simply put `<extra_id_0>` token around the tagged sequence, irrespective of the NER tag. The idea behind this is that most of these entities are repeated both in the conversation and the summary. By tagging them, the models can learn better to copy them from the conversation to the generated summary.

For a more detailed example of the model's input and output for both Tasks A and B, refer to Appendix A.

5 Experimental Setup

To test the importance of each component of our solution, we finetuned the LongT5-Base model[1] with the configurations from the Table 2.

The models are finetuned for 20 epochs using the AdamW optimizer (Loshchilov and Hutter, 2017) and a learning rate of $5 \cdot 10^{-5}$. We trained all our models on a single A100 80GB GPU (University of Tartu, 2018) with a batch size of 8.

To generate the outputs, we used beam search with early stopping and beam width of 4, length penalty of 2.0 (Wu et al., 2016), and the Top-K sampling (Fan et al., 2018) with $k = 50$. Additionally, we limit the maximum generation length to 200 tokens for Task A and 512 tokens for Task B.

Additionally, we preprocessed the Task B data. First, we changed the role markers from "[doctor]" and "[patient]" to "Doctor:" and "Patient:". As a second step, we fixed the punctuation that had an extra space before it. For this, we first split the text by space token and reassembled it with the Treebank detokenizer from NLTK. This was done to ensure consistency between the Task A and B data. Finally, we applied a postprocessing step to TASKAB-TAG model to remove the generated `<extra_id_0>` tokens.

All the reported results were measured on the validation set using the evaluation script provided by the shared task organizers. The following metrics were used: ROUGE score (ROUGE_1, ROUGE_2, ROUGE_L) (Lin, 2004), BERTScore (R_{BERT}, P_{BERT}, F_{BERT}) (Zhang et al., 2019b), and BLEURT (Sellam et al., 2020).

For the final submission, we used TASKAB-TAG model as RUN1, the same model but with the Contrastive Search generation strategy (Su et al., 2022) as RUN2, and a FLAN-T5 base model[2] finetuned identically to TASKA-ONLY but on the tagged data as RUN3. We used all three models for Task A, with the inputs exceeding 512 token length truncated for the RUN3 model, and only RUN1 and RUN2 for Task B.

6 Results and Discussion

Tables 3 and 4 show the results on the validation dataset for Task A and B correspondingly. For both Tasks, the models show a similar pattern: TASKAB model generally performs better for most of the metrics, only falling behind the TASKA-ONLY model in P_{BERT} for Task A and in ROUGE_1 and ROUGE_2 for Task B. TASKAB-TAG model underperforms TASKA-ONLY model in all the metrics for Task A, however, shows better BERTScore performance than TASKB-ONLY model for Task B.

Upon closer inspection of the outputs, we noticed that due to the post-processing error, TASKAB-TAG sometimes produced the output with the space before the punctuation. During the tokenization for calculating the BERTScore and BLEURT, the punctuation with and without space before it results in a different output. Since both metrics use contextual token representations, these

[1] https://huggingface.co/google/long-t5-tglobal-base

[2] https://huggingface.co/google/flan-t5-base

Model	ROUGE_1	ROUGE_2	ROUGE_L	P_{BERT}	R_{BERT}	F_{BERT}	BLEURT
TASKA-ONLY	0.412	0.174	0.344	**0.750**	0.682	0.710	0.523
TASKAB	**0.426**	**0.191**	**0.354**	0.743	**0.705**	**0.718**	**0.542**
TASKAB-TAG	0.384	0.164	0.313	0.726	0.674	0.694	0.471

Table 3: Validation set results for the Task A data. The highest score for each metric is in bold.

Model	ROUGE_1	ROUGE_2	ROUGE_L	P_{BERT}	R_{BERT}	F_{BERT}	BLEURT
TASKB-ONLY	**0.424**	**0.211**	0.241	0.629	0.585	0.606	0.369
TASKAB	0.404	0.205	**0.254**	**0.651**	**0.601**	**0.624**	**0.384**
TASKAB-TAG	0.396	0.202	0.250	0.645	0.590	0.615	0.337

Table 4: Validation set results for the Task B data. The highest score for each metric is in bold.

Model	Acc	ROUGE_1	ROUGE_2	ROUGE_L	P_{BERT}	R_{BERT}	F_{BERT}	BLEURT	Aggr
RUN1	0.680	**0.395**	**0.186**	**0.332**	**0.728**	**0.682**	**0.700**	**0.472**	**0.522**
RUN2	**0.685**	0.360	0.161	0.306	0.703	0.665	0.678	0.445	0.494
RUN3	0.640	0.357	0.160	0.290	0.676	0.680	0.672	0.470	0.500

Table 5: Official test set results for the Task A data. **Acc** column corresponds to the section header classification accuracy and **Aggr** column corresponds to the aggregated score. The highest score for each metric is in bold.

Model	ROUGE_1	ROUGE_2	ROUGE_L	ROUGE_{LSum}
RUN1	0.4137	0.1967	**0.2432**	0.3692
RUN2	**0.4307**	**0.2017**	0.2394	**0.3861**

Table 6: Official test set results for the Task B data. The highest score for each metric is in bold.

extra spaces can negatively impact the final score. To test this, we removed the extra spaces before the punctuation and recalculated the metrics. This resulted in the increased P_{BERT} (+0.003), R_{BERT} (+0.001), F_{BERT} (+0.002) for both Task A and B, as well as BLEURT (+0.033 for Task A and +0.040 for Task B).

To further test the model's factual accurateness, we manually measured on the Task A validation data how well the model captured the age of the patient or other relevant people, the gender of the patient, and the dosage of the prescribed medicine. TASKAB model captured all three categories with the 100% accuracy; TASKAB-TAG model correctly captured the age, gender, and dosage 75%, 100%, and 86% of the times; TASKA-ONLY model showed the accuracy of 81% for age, 100% for gender, and 71% for dosage. Additionally, we tested if the models generated the patient's age and gender in the summary when it was not mentioned in the dialogue: TASKAB model generated the unmentioned patient's age and gender once, TASKAB-TAG twice, TASKA-ONLY trice.

Tables 5 and 6 show the official results on the test set for Task A and B correspondingly. For Task A, the models were ranked by the aggregated score which is calculated as the mean of ROUGE_1, F_{BERT}, and BLEURT. For Task B, the ranking was done by ROUGE_1 score. Overall, for Task A, our best system submission RUN1 was ranked 14th out of 31 total submissions; for Task B, RUN2 was ranked 19th out of 23 total submissions. From these results, RUN2 model that used the contrastive search generation strategy shows better results for longer text generation, however, RUN1 model with beam search generation strategy is better suited for shorter note generation.

Validation set results show that augmenting the data with the clinical named entity recognition tags worsens the model's performance. The NER tags might have introduced additional noise to the data that the model was not able to accommodate during training. Moreover, even though medication and disease names are generally shared between the conversation and the summary note they are not always formulated with the same words. Additionally, the automatic NER tagger may introduce annotation errors that may propagate into the final model. On the other hand, combining the data from both tasks and finetuning using the instruction prompting improved the generation quality.

7 Conclusion

In this paper, we presented our system for the MEDIQA-Chat 2023 shared task on clinical conversation summarization. We showed that finetuning a LongT5 model on several tasks simultaneously improved the model's overall performance and reduced the number of factual errors and hallucinations in the generated summary. On the other hand, augmenting the data with the in-text annotations from the clinical named entity recognition model decreased the summarization quality. Finally, we showed that different text generation strategies can be applied to medical note generation depending on the length of the note.

Code Availability

The code to reproduce the official submission results is available in the following GitHub repository: https://github.com/501Good/MEDIQA-Chat-2023-Calvados.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Asma Ben Abacha, Wen-wai Yim, Griffin Adams, Neal Snider, and Meliha Yetisgen. 2023a. Overview of the MEDIQA-Chat 2023 shared tasks on the summarization and generation of doctor-patient conversations. In *ACL-ClinicalNLP 2023*.

Asma Ben Abacha, Wen-wai Yim, Yadan Fan, and Thomas Lin. 2023b. An empirical study of clinical note generation from doctor-patient encounters. In *EACL 2023*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. 2021a. Improving faithfulness in abstractive summarization with contrast candidate generation and selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5935–5941, Online. Association for Computational Linguistics.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021b. Dialogsum: A real-life scenario dialogue summarization dataset. *arXiv preprint arXiv:2105.06762*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. *arXiv preprint arXiv:1911.12237*.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Nazmul Kazi and Indika Kahanda. 2019. Automatically generating psychiatric case notes from digital transcripts of doctor-patient conversations. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 140–148, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 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.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

George Michalopoulos, Kyle Williams, Gagandeep Singh, and Thomas Lin. 2022. MedicalSum: A guided clinical abstractive summarization model for generating medical reports from patient-doctor conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4741–4749, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. *arXiv preprint arXiv:2202.06417*.

University of Tartu. 2018. UT rocket.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. The aci demo corpus: An open dataset for benchmarking the state-of-the-art for automatic note generation from doctor-patient conversations. *Submitted to Nature Scientific Data*.

Wen-wai Yim and Meliha Yetisgen. 2021. Towards automating medical scribing : Clinic visit Dialogue2Note sentence alignment and snippet summarization. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 10–20, Online. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019b. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yuhao Zhang, Yuhui Zhang, Peng Qi, Christopher D Manning, and Curtis P Langlotz. 2021. Biomedical and clinical english model packages for the stanza python nlp library. *Journal of the American Medical Informatics Association*, 28(9):1892–1899.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. Mediasum: A large-scale media interview dataset for dialogue summarization. *arXiv preprint arXiv:2103.06410*.

A Data Formatting

Following is an example of the data formatting for **Task A** using the tagging from the clinical named entity recognition model.

Input

```
summarize short: Doctor:
Any know <extra_id_0> drug
allergies <extra_id_0>?
Patient: No.
```

Output

```
Section Header: ALLERGY
Section Text: No
<extra_id_0> known drug
allergies <extra_id_0>.
```

Following is an example of the data formatting for **Task B** using the tagging from the clinical named entity recognition model. For the sake of brevity, the input is abridged.

Input

```
summarize objective_exam:
Doctor: hi, martha. how
are you?
Patient: i'm doing okay.
how are you?
Doctor: i'm doing okay.
so, i know the nurse told
you about dax. i'd like
to tell dax a little bit
about you, okay?
Patient: okay.
Doctor: martha is a
50-year-old female
with a past medical
history significant for
<extra_id_0> congestive
heart failure <extra_id_0>,
<extra_id_0> depression
<extra_id_0> and
<extra_id_0> hypertension
<extra_id_0> who presents
for <extra_id_0> her
<extra_id_0> annual exam
<extra_id_0>. so, martha,
it's been a year since
i've seen you. how are
you doing?
...
Doctor: all right. that
sounds good. all right,
well, it's good to see
you.
Patient: good seeing you
too.
Doctor: hey, dragon,
finalize the note.
```

Output

```
objective_exam note:
<extra_id_0> PHYSICAL
EXAMINATION <extra_id_0>
Cardiovascular:
<extra_id_0> Grade 3/6
systolic ejection murmur
<extra_id_0>.
1+ <extra_id_0> pitting
edema of the bilateral
lower extremities
<extra_id_0>.
VITALS REVIEWED
<extra_id_0> Blood
Pressure <extra_id_0>:
<extra_id_0> Elevated
<extra_id_0>.
```