Teddysum at MEDIQA-Chat 2023: an analysis of fine-tuning strategy for long dialog summarization

Yongbin Jeong[1] **Ju-Hyuck Han**[2] **Kyung Min Chae**[2] **Yousang Cho**[2]
Hyunbin Seo[1] **KyungTae Lim**[3] **Key-Sun Choi**[4] **YoungGyun Hahm**[1]
[1]*Teddysum* [2]*Dept. of Medical Engineering, Konyang University*
[3]*Dep. Applied AI, Seoul National University of Science and Technology*
[4]*Dept. of Medical Artificial Intelligence, Konyang University*
{ybjeong, errol.seo, hahmyg}@teddysum.ai,
{21856503, 23856503, 22806506, kschoi}@konyang.ac.kr, ktlim@seoultech.ac.kr

Abstract

In this paper, we introduce the design and various attempts for Task B of MEDIQA-Chat 2023. The goal of Task B in MEDIQA-Chat 2023 is to generate full clinical note from doctor-patient consultation dialogues. This task has several challenging issues, such as lack of training data, handling long dialogue inputs, and generating semi-structured clinical note which have section heads. To address these issues, we conducted various experiments and analyzed their results. We utilized the DialogLED model pre-trained on long dialogue data to handle long inputs, and we pre-trained on other dialogue datasets to address the lack of training data. We also attempted methods such as using prompts and contrastive learning for handling sections. This paper provides insights into clinical note generation through analyzing experimental methods and results, and it suggests future research directions.

1 Introduction

Multi-turn dialogue summarization in the medical field is an important research area. Medical professionals need to make crucial decisions while consulting with various patients, so creating clinical notes from doctor-patient consultations, which record consultation details and diagnoses, is an essential task for both doctors and patients. However, having doctors write entire clinical notes is time-consuming and reduces consultation efficiency. herefore, it is important to develop technologies that can automatically generate clinical notes from conversation content, allowing doctors to simply review and modify the results, which can shorten consultation times. MEDIQA-Chat 2023's shared task(Ben Abacha et al., 2023) is a benchmark task for summarizing, classifying, and generating clinical dialogue data, and Task B(Yim et al., 2023) is a problem of generating a full clinical note from doctor-patient conversations.

This paper describes how we designed and addressed Task B in MEDIQA-Chat 2023 shared task. Task B is a problem that takes clinical consultation dialogues between doctors and patients as input and generates a summarized full clinical note, as shown in Table 1. This Task has three main challenging features that differentiate it from previous tasks:

1. **Long sequences**: This task takes long conversations as input, with an average of 1,246 words per conversation based on the training data, and generates long outputs with an average of 390 words.
2. **Structured output**: The output is semi-structured data, divided into sections. Some sections are composed of typical paragraph forms, while others are briefly represented using symbols like bullet points.
3. **Low-resourced**: The number of training data pairs is only 64, making it a relatively small dataset.

To effectively address these three challenging issues, we propose the following three methods:

1. **Long sequences**: Utilizing the DialogLED(Zhong et al., 2021) model, which is suitable for processing long conversation inputs.
2. **Structured output**: Implementing a robust model for specific section information by adding a prompt feature.
3. **Low-resourced**: Additional pre-training with outside knowledge using AMI(Carletta et al., 2005) and ICSI(Janin et al., 2003) datasets.

In this study, we first attempted to train the DialogLED model, which is pre-trained on long conversations, with the entire input and output for the

Proceedings of the 5th Clinical Natural Language Processing Workshop, pages 394–402
July 14, 2023 ©2023 Association for Computational Linguistics

Input	Output
[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 congestive heart failure , depression and hypertension who presents for her annual exam . so , martha , it's been a year since i've seen you . how are you doing ? [patient] i'm doing well . i've been traveling a lot recently since things have , have gotten a bit lighter . and i got my , my vaccine , so i feel safer about traveling . i've been doing a lot of hiking . uh , went to washington last weekend to hike in northern cascades, like around the mount baker area . [doctor] nice . that's great . i'm glad to hear that you're staying active , you know . i , i just love this weather . i'm so happy the summer is over . i'm definitely more of a fall person . [patient] yes , fall foliage is the best	CHIEF COMPLAINT Annual exam. HISTORY OF PRESENT ILLNESS Martha Collins is a 50-year-old female with a past medical history significant for congestive heart failure, depression, and hypertension who presents for her annual exam. It has been a year since I last saw the patient. . . . REVIEW OF SYSTEMS • Ears, Nose, Mouth and Throat: Endorses nasal congestion from allergies. • Cardiovascular: Denies chest pain or dyspnea on exertion. . . . PHYSICAL EXAMINATION • Cardiovascular: Grade 3/6 systolic ejection murmur. 1+ pitting edema of the bilateral lower extremities.

Table 1: Task B Input and Output example

long sequence problem. And, we pre-trained the model with the AMI and ICSI datasets for solving the low-resource problem before training with the MEDIQA-chat Task B(Yim et al., 2023) dataset. As a result, the ROUGE-1 score for the validation data was relatively high as 0.575 compared to the previous performance. However, when analyzing the results, there were some error cases where the necessary sections did not appear well, or the section name was incorrect, causing the subsequent context to flow in the wrong direction. Taking inspiration from how people create clinical notes while considering the overall section structure, we devised a way to give the model hints about the section that needs to be created. Like determining the title of an article in advance, we added prompts to the input to help the model understand which section to create and generated summary sentences for only that section. We then combined the summary sentences for each section to create a complete summary note. However, this approach increased the recall for sections but had the problem of many sections appearing that did not need to appear. The biggest issue was that the content discussed in other sections was repeated. In about 70% of cases, the content was repeated, resulting in a lower overall note score of 0.449 based on ROUGE-1. In addition, we tried various methods such as using only the section name as a prompt, changing the prompt sentence, or wrapping the section name in tokens, but these did not make a significant difference in performance.

To address these issues, we incorporated contrastive learning(Chen et al., 2020). We set the summary of the section corresponding to the prompt as the positive sample for contrastive learning and applied the cross-entropy loss, as usual, to generate summaries close to it. To avoid generating repetitive summaries similar to other sections, we used the summaries of other sections as negative samples and set the loss so that cosine similarity would decrease. As a result of this contrastive learning, the occurrence of repetitive content decreased by nearly 60%, indicating that contrastive learning had some influence.

This paper introduces various attempts for Task B, such as using the DialogLED model, creating partial summaries for each section using prompts, changing prompts, and employing contrastive learning to improve Task B's performance. While we have not yet found a perfect solution, we aim to provide valuable insights for considering approaches to Task B through quantitative and qualitative analysis of data and experimental results. Source code and all the trained models are available on our GitHub repository[1].

2 Related Work

Abstractive summarization of extended conversations is typically approached using generative models such as BART(Lewis et al., 2019). When the input exceeds the model's input length constraint, rudimentary techniques such as truncating the end or middle portions of the input are employed. Methods like Presumm(Liu and Lapata, 2019) address long inputs by extracting key sentences and subsequently performing abstractive summarization on them. Alternatively, models like DialogBERT(Gu et al., 2020) process inputs at the utterance level, encoding and merging them to accommodate the entire conversation. Recently, a trend towards models like Longformer(Beltagy et al., 2020), DialogLM, and DialogLED(Zhong et al., 2021) has emerged, as these models can handle longer inputs owing to their expanded input length capabilities. Prominent datasets for summarizing long conversations include AMI(Carletta et al., 2005) and ICSI(Janin et al., 2003) datasets. These datasets, like Task B, involve summarizing lengthy dialogues. However, unlike Task B, where the domain is clinical, the domain for these datasets is meetings, and the summaries are composed of unstructured paragraphs. And, models such as clinicalBERT(Huang et al., 2019), which are trained in the clinical domain, have emerged, but they have limitations in terms of input length compared to recent models.

3 Dataset and task design

The Task B of MEDIQA-Chat 2023, as presented in Table 1, involves generating a full clinical note based on clinical dialogues between docker and patient. The input comprises multi-turn conversations, with specialized terminology from the clinical domain such as disease names and medication names appearing frequently. The output is organized into distinct sections, including 'CHIEF COMPLAINT' and 'HISTORY OF PRESENT ILLNESS.' Notably, not all sections are required to appear, and there is no predefined structure specifying which sections should be included in the note. Among the sections, 'CHIEF COMPLAINT' is the most common, appearing in 63 out of 64 training samples, whereas 'PAST MEDICAL HISTORY' is the least common, occurring only once. Furthermore, the same section can be represented using different names, such as 'CC:' (four instances) and 'CHIEF COMPLAINT' (59 instances).

	Train	Valid
# of data	67	20
# of avg input Char	6443	6124
# of avg output Char	2649	2716
# of avg input Word	1246	1169
# of avg output Word	390	400
# of avg input Token	1480	1401
# of avg output Token	573	589
# of max input Token	3437	2020
# of max output Token	1192	1054
# of input over 512 tokens	67	20
# of output over 512 tokens	40	11

Table 2: Data Statistic

Table 2 illustrates the overall statistics of the dataset, highlighting the limited number of training samples (67). Furthermore, the input length is relatively long, averaging 1,246 words per training sample, which translates to an average of 1,480 tokens when tokenized using the BERT(Devlin et al., 2019) tokenizer. The longest document consists of 3,437 tokens, indicating that conventional models such as BART(Lewis et al., 2019), with a maximum input length of approximately 1,024 tokens, may not be suitable for directly processing inputs for this task.

Considering these aspects, the challenges to be addressed in this task, distinct from other summarization tasks, can be summarized as handling long dialogues, working with a small dataset, generating semi-structured notes, and adapting to the intricacies of the clinical domain.

4 Method

To address these challenges, we employed two primary approaches. The first approach involves using the entire dialogue as input and generating the full clinical note as output, thereby adopting an end-to-end learning method. The second approach consists of creating notes for each section individually, and

[1] https://github.com/teddysum/MEDIQA-Chat-2023-Teddysum

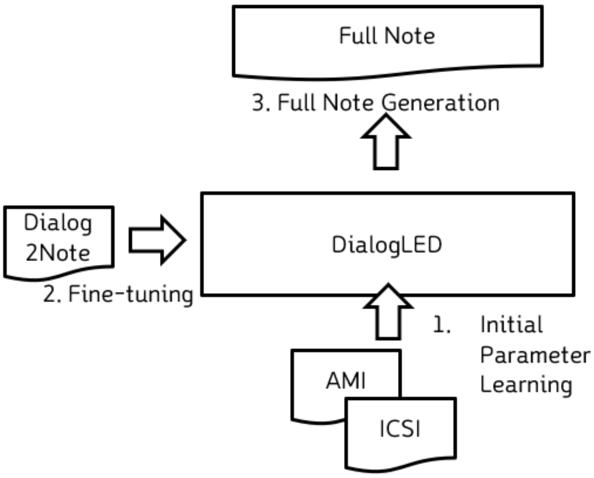

Figure 1: Fine-tuning to full clinical note

subsequently merging the section-specific notes to form a comprehensive full clinical note.

4.1 Full fine-tuning to clinical note

In the first approach, we designed a model based on the DialogLED model, which has been pre-trained on long conversational data, to handle the lengthy dialogue inputs. Although the DialogLED model is trained on dialogue data, it has not been trained specifically for summarization tasks. To overcome the limitations posed by the small dataset, we conducted pre-training using the AMI and ICSI datasets, which consist of summaries of long conversations. Subsequently, we fine-tuned the model using the MEDIQA dataset as illustrated in Figure 1. During this process, we did not explicitly handle the section structure, allowing the model to learn in an end-to-end manner. While this approach demonstrated the advantage of generating accurate content for the full clinical note, it fell short in creating section headers and properly distinguishing content between sections.

4.2 Prompt-based partial fine-tuning

The second approach involves handling the section structure through a prompt-based method. As the first approach struggled to generate section headers effectively, this method aims to produce content for each section, create section headers through post-processing, and combine the resulting partial notes to form a full note. We added a special token, <CMD>, to the prompt as in "`$conversation`<CMD>Based on this conversation, make a summary of the `$SectionName`," which is appended after the dialogue. Various forms of prompts were experimented with, as demonstrated in Table 3.

Style	**Prompt**
Style 1: simple prompt	`$conversation`<CMD>Based on this conversation, make a summary of the `$SectionName`
Style 2: special tokens (before/after)	`$conversation`<CMD>Based on this conversation, make a summary of the <SEC>`$SectionName`</SEC>
Style 3: section name only	`$conversation`<CMD> `$SectionName`

Table 3: Prompt styles

4.3 Prompt-based contrastive learning

Prompt-based partial fine-tuning method often led to the inclusion of content that should have appeared in other sections, particularly when the model's understanding of the prompt was inadequate. To address this issue, we also conducted experiments using a contrastive learning approach for training. In the traditional learning approach, the loss function computes the cross-entropy value for the correct summary. In contrast, the contrastive learning approach utilizes content from other sections as negative samples to prevent the generation of content from different sections. When the model's predicted value is y, the correct summary for the target section is p, and the summary for a randomly chosen section, excluding the target section, is n, the loss can be calculated as follows.

$$CEloss = CrossEntropy(y, p)$$

$$CSloss = CosineSimilarity(y, n)$$

$$loss = a * CEloss + (1 - a) * CSloss$$

5 Experiments

5.1 Implementation Details

In this study, we utilized the pre-trained DialogLED model[2]. The batch size was set to 4, with a maximum input length of 5120 and a maximum output length of 1024. We monitored the training process for up to 80 epochs. The AdamW optimizer was employed, with a learning rate of 2e-5, an epsilon value of 1e-8, 50 warm-up steps, and a

[2] https://huggingface.co/MingZhong/DialogLED-large-5120

learning rate decay of 0. For the training, 4 A100 GPUs(80GB) were used with 12GB dedicated to model uploading and 47GB used for uploading one batch of training and evaluation data. The first model, which fine-tuned the full note in one go, took approximately 3 minutes per epoch, while the second model, which generated summaries for individual sections, took around 13 minutes per epoch.

For the pre-training of the DialogLED parameters, the AMI and ICSI datasets were utilized. A total of 170 samples, comprising 117 from the AMI dataset and 53 from the ICSI dataset, were used as training data, while 20 from the AMI dataset and 25 from the ICSI dataset were used as the validation set. The input data was pre-processed to match the input format of the DialogLED model, with speakers and utterances represented using colons, such as "A: utterance."

During pre-processing, the MEDIQA-chat format, which presents the speaker in brackets followed by their utterance (e.g., "[doctor] utterance"), was transformed to match the conversation format learned by the DialogLED model (e.g., "doctor: utterance") using colons. In the second approach, where summaries were generated for each section, section names were heuristically defined and unified, as shown in Table 4, by converting synonymous section headers like 'CC:' and 'CHIEF COMPLAINT' to 'CHIEF COMPLAINT'. For the 'ASSESSMENT AND PLAN' section, instances where the section appeared as 'ASSESSMENT AND PLAN' or separately as 'ASSESSMENT' and 'PLAN', or only one of them appeared, were combined into a single 'ASSESSMENT AND PLAN' section.

5.2 Experimental Results

Since the ground truth to the test set in Task B is not open to the public, all evaluations were conducted on the validation set.

5.2.1 Quantitative Analysis

Table 5 presents the scores for generating full notes. The FT to Full Note model generates full notes given the entire conversation as input, while the models labeled PT partially generate notes based on prompt-based approach. As shown in Table 3, there are three styles for the prompts. PT contrastive refers to the model trained using a contrastive learning approach for prompt-based models. As the results indicate, the highest score for full notes is achieved by the FT to Full Note model, while the

Original section name	**Unified Section name**	**# of appearance**
ASSESSMENT	ASSESSMENT AND PLAN	29
ASSESSMENT AND PLAN	ASSESSMENT AND PLAN	34
PLAN	ASSESSMENT AND PLAN	32
EXAM	PHYSICAL EXAMINATION	4
PHYSICAL EXAM	PHYSICAL EXAMINATION	44
PHYSICAL EXAMINATION	PHYSICAL EXAMINATION	16
HISTORY OF PRESENT ILLNESS	HISTORY OF PRESENT ILLNESS	45
HPI:	HISTORY OF PRESENT ILLNESS	4
REVIEW OF SYSTEMS	REVIEW OF SYSTEMS	50
CC:	CHIEF COMPLAINT	4
CHIEF COMPLAINT	CHIEF COMPLAINT	59
RESULTS	RESULTS	52
CURRENT MEDICATIONS	MEDICATIONS	8
CURRENT MEDICATIONS:	MEDICATIONS	1
MEDICATIONS	MEDICATIONS	19
PAST HISTORY	HISTORY	9
PAST MEDICAL HISTORY:	HISTORY	1
MEDICAL HISTORY	HISTORY	18
SOCIAL HISTORY	HISTORY	28
SURGICAL HISTORY	HISTORY	7
FAMILY HISTORY	HISTORY	10
IMPRESSION	IMPRESSION	4
INSTRUCTIONS	INSTRUCTIONS	32
VITALS	VITALS	23
VITALS REVIEWED	VITALS	3

Table 4: Section Name Definition

Models	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	BLEURT
FT to Full Note	0.575	0.288	0.315	0.692	0.405
PT Style 1	0.431	0.232	0.207	0.662	0.336
PT Style 2	0.439	0.234	0.201	0.661	0.324
PT Style 3	0.447	0.235	0.208	0.66	0.342
PT Contrastive	0.414	0.215	0.192	0.669	0.350

Table 5: Full note generation score

Models	SH P	SH R	SH F1	Context Repetition Rate
FT to Full Note	0.91	0.47	0.62	0.08
PT Style 1	0.65	0.86	0.74	0.67
PT Style 2	0.67	0.87	0.76	0.71
PT Style 3	0.66	0.85	0.74	0.72
PT Contrastive	0.65	0.87	0.74	0.64

Table 6: Error statistic analysis. SH means Section Header.

Models	ROUGE-1	BERTScore	BLEURT
L to L	0.408	0.646	0.309
L to S	0.575	0.692	0.405
S to L	0.518	0.668	0.387
S to S	0.563	0.69	0.397

Table 7: Validation loss and score in fine-tuning to full note. L signifies the point at which the validation loss converges, and S denotes the point of validation score convergence. The prefix L and S pertain to the training phase on the AMI and ICSI datasets, while the suffix L and S refer to the training on Task B data. For instance, L to S indicates that the model was trained on Task B using the point at which the validation loss converges during the training on the AMI and ICSI datasets, and the recorded score corresponds to the highest validation score for that model.

PT models exhibit similar scores overall.

Table 6 provides statistics on the errors that occurred during the generation of full notes. Section Header P, R, and F1 represent the Precision, Recall, and F1 score calculated based on the presence of Section headers in both the Validation set and the model output. Specifically, True Positive (TP) occurs when a Section header appearing in the Validation set also appears in the model output, False Negative (FN) occurs when a Section header present in the Validation set does not appear in the model output, and False Positive (FP) occurs when the model output generates a Section header that does not exist in the Validation set. As shown, the FT to Full Note model exhibits high precision but low recall, leading to a reduced overall F1 score. In contrast, the PT models demonstrate better generation of section headers due to their higher recall.

The Context Repetition Rate refers to the proportion of summary content in a section that appears in other sections. In the FT to Full Note approach, the repetition rate is relatively low at 0.08, indicating that there is minimal repetition of content. In contrast, the PT approach exhibits a repetition rate near 0.7, suggesting that the majority of sections contain similar content to other sections. Although the PT method generates Section Headers more effectively than the FT to Full Note approach, the overall score is lower due to this repetition issue. To address this problem, contrastive learning was implemented; however, the PT Contrastive method still exhibits a repetition rate of 0.64. While this is a decrease in repetition, it still remains a relatively high value. The criterion for determining repetition is a cosine similarity of 0.5 or higher between the summary sentences of two sections.

Taking these factors into account, the FT to Full Note approach demonstrates a good ability to summarize the overall context but falls short in handling individual sections. As a result, it tends to generate frequently occurring sections and fill them with substantial content, leading to lower performance in section header evaluation. However, since the entire note is generated at once, content from one section does not appear in other sections, maintaining distinctiveness. On the other hand, while the prompt based partial generation method yields better evaluation results in section creation, the repetition issue remains inevitable, and even methods like contrastive learning cannot easily resolve it. This is because the model generates summaries

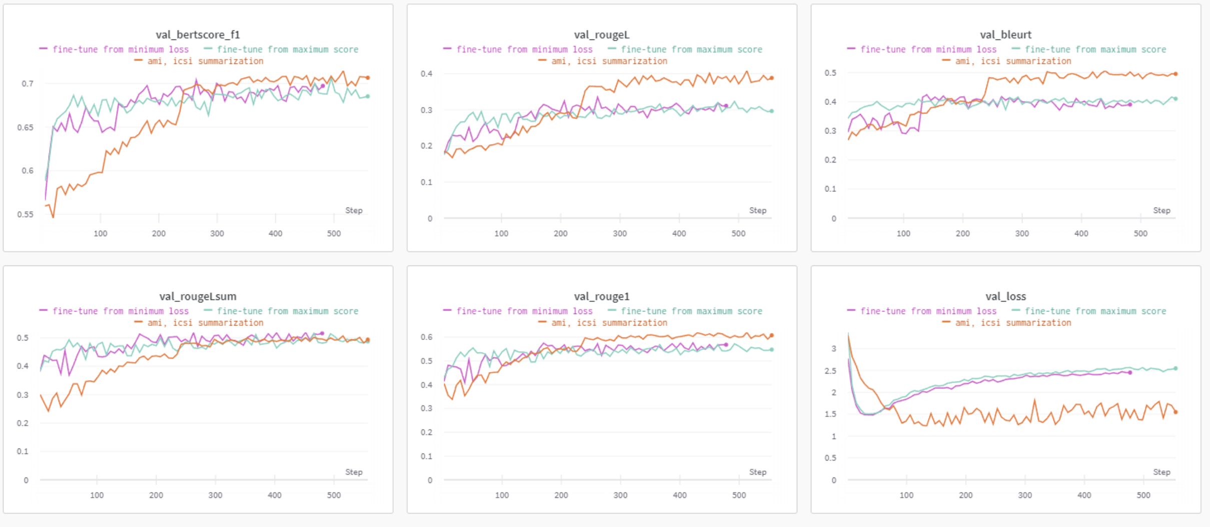

Figure 2: Training graph

separately for each section, so it does not know the content of summaries in other sections, leading to the learning of frequently occurring content repetition.

Additionally, during the learning process for Task B data after initial parameter tuning with the AMI and ICSI datasets, it was discovered that the points at which validation loss and validation score converge are different. As shown in Figure 2, when training to the AMI and ICSI data, the validation loss converged at 100 steps, but the validation score continued to rise, converging at approximately 300 steps. To analyze this, we evaluated two models: one trained on Task B data from the model at 100 steps and the other from the model at 300 steps. The results are shown in Table 7. When using a model trained on AMI and ICSI data, it can be seen that using a model from the point where validation loss converges performs better than using a model with a higher validation score. This suggests that the model may have started overfitting from the point where the validation loss converged. Therefore, a model from the point of validation loss convergence is more suitable for application to other domains like Task B, as it makes more general predictions and generates content. The mismatch between the timing of validation loss and validation score convergence also occurs when training on Task B data, which seems to be due to the small amount of data. In other words, when a large model learns from a small dataset, it appears to perform better when it is somewhat overfitted and dependent on the dataset, rather than at the point with the lowest validation loss, which seems to be more generally and appropriately trained.

Futhermore, as seen in Figure 2, the validation loss for the AMI and ICSI datasets converges and no longer increases, while for Task B, the validation loss increases again after converging. This more clearly shows the issue of overfitting. To address this issue, we can consider increasing the amount of dataset, constructing a model that is more suitable for semi-structured datasets to prevent overfitting, or modifying the loss function to be more appropriate for the dataset.

5.2.2 Qualitative Analysis

We also conducted a qualitative evaluation of the model results. We evaluated the model by comparing all the inferred results on the validation set. We compared the generated summaries with the correct summaries by section, examining whether the necessary sections were created, whether unnecessary sections were created, and what differences existed between the content of the created sections and the correct summary. First, we verified whether the results of the quantitative evaluation were consistent with those of the qualitative evaluation. Additionally, we qualitatively analyzed the phenomenon where the point of convergence for the validation score and the validation loss differed.

When evaluating the performance of the model qualitatively, the results were similar to those of the quantitative evaluation. The FT to Full Note model actually performed better than the PT model. In particular, it demonstrated high performance in frequently occurring and easy sections, such as the ’CHIEF COMPLAINT’ section. In the FT model, as in the quantitative analysis, there were many instances where the necessary sections were not created. Upon closer inspection, it was found that the content was often generated but not separated by section headers and instead placed in a single, generic section. This appears to be due to the imbalance in the frequency of section occurrences in the data, data scarcity, and the model's limitations in understanding the semi-structured structure. In the case of the PT model, 'content repetitions', in which the contents of other sections appear redundantly, was remarkably high, as would be evaluated in the quantitative analysis.

Secondly, we analyzed the qualitative differences between the points where the validation loss converged and the points where the validation score converged. As a result, in the case of the FT to Full Note model, when viewed qualitatively, it showed a better ability to create sections and fewer instances of empty content within the sections at the points where the score converged. In the case of the PT model, as the number of epochs increased, the overlap of content with other sections occurred more frequently. This suggests that the model may consider it more advantageous to create content, even if it is incorrect, rather than not create it by mistake, for content that appears most generically. In other words, it can be seen as overfitting to the correct answers in the data. However, in sections such as 'IMPRESSION', 'INSTRUCTIONS', and 'VITALS', the model tended to produce identical outputs at low epochs and different outputs as the number of epochs increased. This suggests that the model is gaining some understanding of the prompts as it learns.

Additionally, we analyzed the learning process for Task B after training on AMI and ICSI data. We qualitatively compared the differences between the models trained from the point where validation loss converged (L to S) and the models trained from the point with high validation scores (S to S). As a result, the S to S models tended to overfit to the AMI and ICSI data, causing many inappropriate words that do not exist in the MEDIQA data to appear. However, for patient information such as names and ages, the S to S models performed better than the L to S models. This suggests that the S to S models overfit to the AMI and ICSI data and learned to reference the original text more extensively. Although the L to S models achieved higher quantitative scores than the S to S models, the accuracy of information such as patient names and ages is more important for clinical notes. Therefore, the S to S models can be considered better suited for this task.

6 Conclusion

In this paper, we have conducted various experiments for Task B and analyzed the results. We utilized the DialogLED model to handle long inputs and employed additional data, such as the AMI and ICSI datasets, to address the limited data issue. As a result, the model generating the entire note at once achieved high scores for the full note but low scores for section header creation. In contrast, the approach of creating section summaries separately and then combining them for a complete note had high scores for section headers but low scores for the entire full note due to repetition issues. Analyzing these results, it appears that generating the entire content at once is a more appropriate approach since it is challenging to avoid repetitive content when creating section summaries separately. In conclusion, to learn the characteristics of Task B, which requires the creation of a semi-structured structure, a hybrid approach that combines generating the entire content at once and creating section headers separately is needed, rather than relying on simple fine-tuning.

Limitations

In the experiments conducted by the team, several limitations were observed. Firstly, the handling of the semi-structured structure was not accomplished perfectly. When generating the entire content at once, many sections did not appear, and when creating sections separately, the content of the notes was repetitive. Additionally, addressing the clinical domain properly was not achieved.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (2022-0-00078, Explainable Logical Reasoning for Medical Knowledge Generation)

References

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

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

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Maël Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska Masson, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre D. Wellner. 2005. The ami meeting corpus: A pre-announcement. In *Machine Learning for Multimodal Interaction*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805.

Xiaodong Gu, Kang Min Yoo, and Jung-Woo Ha. 2020. Dialogbert: Discourse-aware response generation via learning to recover and rank utterances. In *AAAI Conference on Artificial Intelligence*.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *ArXiv*, abs/1904.05342.

Adam L. Janin, Don Baron, Jane Edwards, Daniel P. W. Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The icsi meeting corpus. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, 1:I–I.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Annual Meeting of the Association for Computational Linguistics*.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *ArXiv*, abs/1908.08345.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Acibench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Submitted to Nature Scientific Data*.

Ming Zhong, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Dialoglm: Pre-trained model for long dialogue understanding and summarization. In *AAAI Conference on Artificial Intelligence*.