QUB-Cirdan at "Discharge Me!": Zero shot discharge letter generation by open-source LLM

Rui Guo[1,2] [*]   **Greg Farnan**[2] [†]   **Niall McLaughlin**[1] [‡]   **Barry Devereux**[1] [§]

[1] Queen's University Belfast
[2] Cirdan

Abstract

The BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation aims to reduce the administrative burden on clinicians by automating the creation of critical sections of patient discharge letters. This paper presents our approach using the Llama3 8B quantized model to generate the "Brief Hospital Course" and "Discharge Instructions" sections. We employ a zero-shot method combined with Retrieval-Augmented Generation (RAG) to produce concise, contextually accurate summaries. Our contributions include the development of a curated template-based approach to ensure reliability and consistency, as well as the integration of RAG for word count prediction. We also describe several unsuccessful experiments to provide insights into our pathway for the competition. Our results demonstrate the effectiveness and efficiency of our approach, achieving high scores across multiple evaluation metrics.

1 Introduction

The BioNLP ACL'24 Shared Task, "Discharge Me!" on Codabench (Xu et al., 2024), focuses on automating the creation of two crucial sections of patient discharge letters: "Brief Hospital Course" (BHC) and "Discharge Instructions" (DI). This initiative arises in response to significant time burdens on clinicians, highlighted by surveys of U.S. physicians. One study found that physicians spend twice as much time on Electronic Health Records (EHR) compared to direct patient interactions during clinical hours (Sinsky et al., 2016). Another survey involving 1,524 physicians revealed an average of 1.84 hours spent on EHR documentation outside office hours. Automating the generation of BHC and DI aims to significantly reduce the clerical load on healthcare providers, thereby improving patient service quality and potentially mitigating clinician burnout.

A discharge letter, or a discharge summary, is a critical document summarizing a patient's hospital visit from admission to discharge, serving as a bridge between hospital care and follow-up with outpatient providers. Among its several sections, the "Brief Hospital Course" outlines the patient's treatment and progress during the hospital stay, typically using clinical jargon best understood by healthcare professionals. Conversely, the "Discharge Instructions" are designed to guide patients and their caregivers once they leave the hospital, using layman's language to clearly explain follow-up care, medication regimens, and lifestyle recommendations.

Large Language Models (LLMs) offer a promising solution for automating medical documentation due to their ability to understand and generate human-like text (Singhal et al., 2023a; Zhang et al., 2023). Unlike traditional extractive summarization (El-Kassas et al., 2021), which predominantly involves concatenating snippets from existing texts, LLMs can enhance summarization by integrating both extractive and abstractive techniques. This has been applied to progress note summarization (Gao et al., 2022; Liu et al., 2023), similar to this Codabench challenge. With both proprietary LLMs such as ChatGPT (OpenAI, 2024) and open-source LLMs such as Llama3 (AI@Meta, 2024), the potential for creating accessible medical summaries is significant.

In this challenge, we propose a zero-shot approach utilizing the Llama3 8B quantized model, which is optimized for low computing resource usage without fine-tuning, and the result is in the top 10 in the final benchmark assessment. Our key contributions are:

---
[*]rui.guo@cirdan.com
[†]greg.farnan@cirdan.com
[‡]n.mclaughlin@qub.ac.uk
[§]b.devereux@qub.ac.uk

- Crafting specialized templates for the "Brief Hospital Course" and "Discharge Instructions" sections, with carefully designed prompts to ensure the generated text is medically reliable and stylistically consistent with the training dataset.

- Exploring various methods to estimate the total word count for the target sections, including:

    - Fitting a statistical distribution

    - Employing a random forest classifier

    - Implementing a context-based retrieval system

- Conducting all experiments using a T4 GPU, demonstrating that our approach is computationally efficient.

2 Related Work

The application of foundation models, pre-trained on billions of tokens from diverse data sources, is increasingly prevalent in healthcare (He et al., 2024). These models are pivotal in various domains, such as diagnosis generation (Gao et al., 2023b) and medical image analysis (Zhang et al., 2024). Within clinical text processing, large language models (LLMs) are employed for tasks including summarization (Van Veen et al., 2023; Gao et al., 2023a) and answering medical questions (Singhal et al., 2023b). Specifically, the "Discharge me!" challenge involves condensing extensive medical records into succinct discharge letters while retaining all critical information, making LLMs suited for this task.

Participants in the BioNLP 2023 Workshop's Problem List Summarization task often utilized T5 (Raffel et al., 2020) or BART (Lewis et al., 2019) models, enhancing these backbones either by further training on clinical texts or fine-tuning for specific clinical tasks (Gao et al., 2023a). This further pre-training introduces medical knowledge not originally present in the LLM while fine-tuning adapts the model to produce outputs in the correct format for the target task.

Several studies such as BioMistral (Labrak et al., 2024) and PMC-LLaMA (Wu et al., 2024) have adapted open-source LLMs by applying pre-training and fine-tuning sequentially. Conversely, Med-PaLM (Singhal et al., 2023a) bypasses additional pre-training, relying solely on fine-tuning from a vast pre-trained dataset. On a different note, BioMedLM (Bolton et al., 2024) focuses exclusively on medical texts, resulting in a smaller model but still competes effectively with models trained on larger, more general datasets.

Pre-training and fine-tuning LLMs require GPUs with significant memory capacities (often exceeding 16GB). Fine-tuning can take several days, even using Parameter-Efficient Fine Tuning (PEFT) methods like LoRA (Hu et al., 2021). However, modern LLMs can exhibit strong performance without additional fine-tuning if provided with the appropriate context and instructions. For instance, Almanac (Zakka et al., 2024) enhances its output by retrieving clinical question-related knowledge from curated sources, a technique known as Retrieval-augmented Generation (RAG) (Gao et al., 2023c). Additionally, Medagents (Tang et al., 2023) demonstrates that a zero-shot method, which deconstructs the question into distinct steps and assigns specific prompts and roles to the LLM for each stage, can achieve competitive results compared to more traditional few-shot approaches.

3 Methods

In this section, we introduce our zero-shot template-based approach, combined with RAG, to determine the target word count, which is both effective and resource-friendly. We adopted the Llama3 8B model with 8-bit quantization as the open-source model for this challenge. Figure 1 illustrates our approach:

1. Splitting the full discharge letter into different segments, such as "Chief Complaint" and "Brief Hospital Course". This allows us to selectively use relevant sections and discard or truncate those too lengthy to process.

2. Employing Retrieval-Augmented Generation (RAG) to find the most similar patient's target section, using that section's word count as the target for generation. Generating a similar word count to the target can help maintain the generated summaries' completeness and increase evaluation metrics such as BLEU, ROUGE, and METEOR.

3. Providing the target section's structure template and prompt to Llama3 along with

the patient's context and target word count.

4. Generating the result by Llama3 8B quantized model.

While GPT-4/3.5 models generally outperform open-source models such as Llama2 in understanding EHR data (Liu et al., 2024), the rules of this challenge discourage the use of proprietary model APIs (e.g., OpenAI's GPT-4). Consequently, we resorted to the state-of-the-art (SOTA) open-source model, Llama3 (AI@Meta, 2024). Our approach leverages the full text from the "text" field in the provided discharge.csv file, alongside aggregated fields from other MIMIC-IV tables, including patient information, diagnoses, and transfer history. We meticulously curated a template for each target section and designed prompts to guide the LLM in generating the required sections. In addition to our final approach, we documented several other zero-shot methods for target section generation and various approaches to predict the target section's word count. However, these were not adopted in our final solution.

3.1 Dataset Exploration

The dataset for this challenge is derived from MIMIC-IV's submodules, MIMIC-IV-Note (Johnson et al., 2023c) and MIMIC-IV-ED (Johnson et al., 2023a). All patients have visited the Emergency Department (ED), and the final target sections, "Brief Hospital Course" and "Discharge Instructions", are extracted from their discharge letters. Since patients can be admitted to the hospital after their initial ED visit, we also explored other tables from the MIMIC-IV hosp and ICU modules (Johnson et al., 2023b) to provide a comprehensive view of the patient's hospital stay beyond the ED information.

Due to limited context length, we could not simply pass all available information into the LLM. Therefore, we ranked all sections of the discharge letter to select a subset of the information. We segmented the discharge letter's "text" column from discharge.csv using regex and a template of keywords for different sections, as shown in the Section column of Table 1. Besides the information from the "text" column, we aggregated "Patient Admissions" information, including gender, race, age (calculated), "Diagnoses" (throughout the patient stay), and "Transfer Summary" from other MIMIC-IV tables. Since we compiled the patient's

diagnoses and transfer summary for the entire hospital stay using other MIMIC-IV tables rather than just the Emergency Department (ED) stay, we did not use the tables in the ED module, such as triage, edstays, and diagnosis, as they only cover part of the patient's stay. The content of "radiology" will be set to the content of the section "Imaging" if the "Imaging" section is empty in the discharge letter. We then calculated the average ranking of the metric score for each section relative to the target sections, using the provided evaluation metrics, including BLEU-4 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang et al., 2019), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). Each section was compared to the target sections, "Brief Hospital Course" (BHC) and "Discharge Instructions" (DI), with higher-ranking sections being more related to the target sections. Table 1 shows that "History of Present Illness" is most related to the BHC section, followed by imaging results, physical exams, past medical history, and diagnoses. BHC is most related to DI, followed by sections related to BHC.

Based on the ranking in Table 1 and the length of each section, we selected "History of Present Illness", "Imaging and Studies", "Past Medical History", "Patient Admissions", and "Chief Complaint" as the context for the BHC section. We used the generated BHC, "Discharge Medications", "Discharge Disposition", "Discharge Diagnoses", "Discharge Condition", and "Followup Instructions" for DI section. Other sections related to DI were excluded because they are also related to BHC. We truncated each section to the 95th percentile of its total length to remove outliers and potential segmentation errors.

3.2 Retrieval for the Target Section Word Count

Understanding the target section's word count is beneficial for generating the appropriate amount of text, thereby improving the evaluation metrics for this challenge. Figure 2 shows the word count distribution for the target sections in the training dataset. Both target sections have right-skewed distributions, and BHC also has a peak for word counts under 100. We hypothesize that patients with similar backgrounds may have similar target sections. These retrieved target sections from patients with similar backgrounds can be used
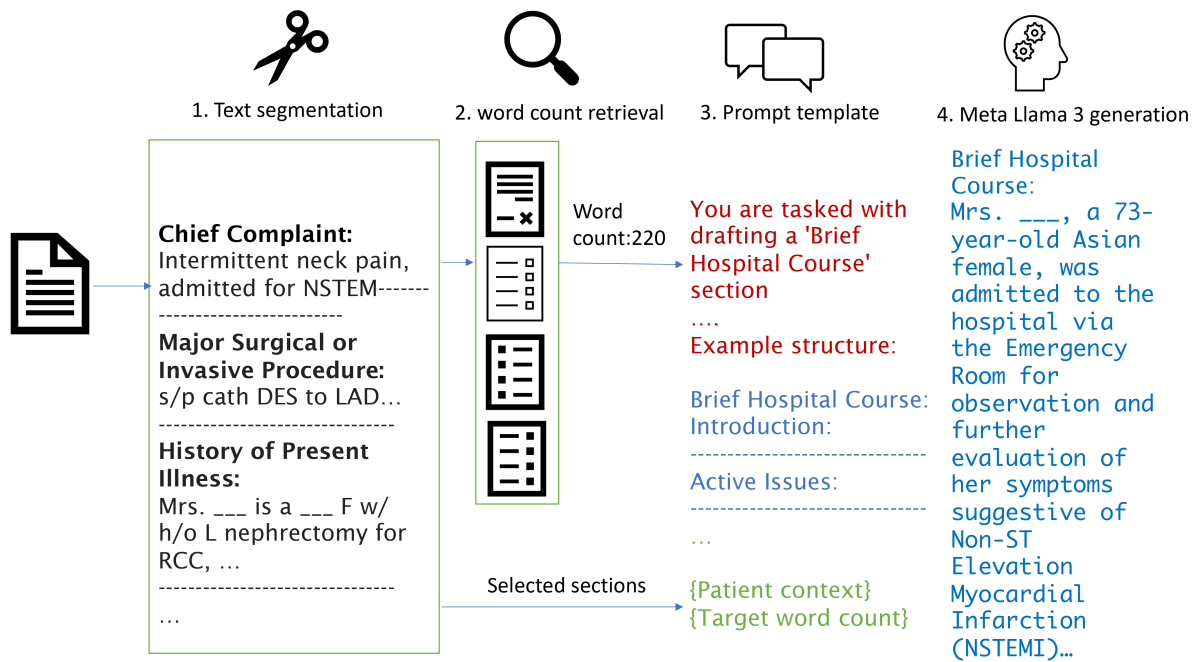
Figure 1: Overview of our solution. The figure illustrates our four-step approach: (1) Text Segmentation: splitting the discharge letter into sections such as "Chief Complaint" and "Brief Hospital Course"; (2) Retrieval-Augmented Generation (RAG): retrieving similar patient sections to determine word count; (3) Template and Prompt Design: providing structured templates and prompts to Llama3 with patient context and target word count; (4) Text Generation: generating the final output using Llama3.

as a starting point, providing a template or word count for further refinement. We selected "Chief Complaint", "Diagnoses", and "History of Present Illness" as inputs for retrieving the BHC section. We added "Admission medications", "Discharge Medications", "Discharge Disposition", "Discharge Diagnoses", and "Discharge Condition" for retrieving the DI section. We used the "sentence-transformers/all-MiniLM-L6-v2" model to create embeddings of the context information for each training dataset entry and FAISS for similarity search. The word count from the first retrieved document's target section was used in the prompt to LLM for the generation. We compared this word count selection strategy to using a fixed word count, and the results are presented in Section 4.

3.3 Target Section Structure Template and Prompt Creation

The target word count distribution varies, and we inspected several randomly chosen examples of target sections with different word counts. We selected examples with word counts over 180 to accommodate most cases for BHC template construction. Examples with word counts between 100-300 were chosen for the DI template

construction. The structure is in JSON format, with names and descriptions for each section.

The BHC structure template is:

```
1. Introduction:    Brief   introduction
   including    patient    demographics,
   significant   past   medical   history,
   and reason for hospitalization.

2. Active Issues: Details of the primary
   medical concerns addressed during the
   stay,  including  initial  assessments
   and management actions.

3. Chronic       Issues       (Optional):
   Management     of    known     chronic
   conditions    during    the    hospital
   stay.

4. Transitional     Issues     (Optional):
   Specific      follow-up       actions
   recommended     for     post-discharge
   care.

5. Additional       Notes       (Optional):
   Other    pertinent    information    or
```
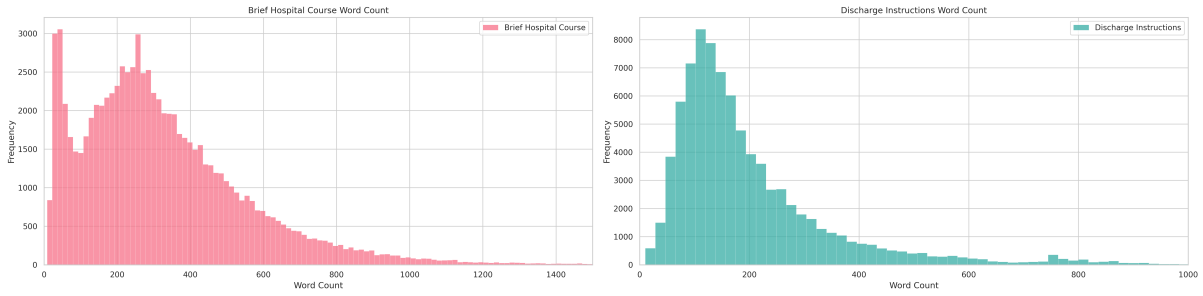
667

Figure 2: The target section word count distribution. Both BHC and DI have right-skewed distributions. BHC has two peaks, one below 100 words and one around 250 words.

Section	BHC	DI
Patient Admissions	13	21
Transfer Summary	15	23
Diagnoses	5	4
Service	11	12
Allergies	14	22
Attending	17	24
Chief Complaint	8	11
Major Surgical Procedure	9	17
History of Present Illness	1	2
Review of System	10	15
Past Medical History	4	9
Social History	16	25
Family History	12	16
Physical Exam	3	5
Pertinent Results	7	18
Imaging and Studies	2	3
Brief hospital course		1
Admission Medications		10
Discharge Medications		7
Discharge Disposition		14
Discharge Diagnoses		6
Discharge Condition		8
Followup Instructions		13
Provider		19
Code Status		20

Table 1: The ranking of different sections' relation to BHC/DI by averaging all the evaluation metrics provided by this challenge. We aggregated the patient's admission info, including gender, race, age (calculated), diagnosis, and transfer history from other MIMIC-IV tables.

considerations affecting patient care.

The template includes several optional sections not included in all the examples. The template will be fed to the prompt below as the "structure" variable. The prompt for BHC is:

```
As a medical professional, you are
tasked with drafting a "Brief Hospital
Course" section for a discharge letter.
Utilize the structure from a brief
hospital course example to guide your
composition. The goal is to write a
new, coherent, brief hospital course for
another patient based on the provided
structured template. The total word
count for the brief hospital course
should be {words} words.

BHC Instructions:

1. Follow the JSON template provided
   to structure the new brief hospital
   course. Each section should be filled
   according to the relevant patient
   information.

2. Omit the optional sections if they are
   irrelevant to the patient's case.

3. Omit the optional sections if the
   total word count is less than 100
   words.

4. Do not add a new section after
   Additional Notes.

5. Use placeholders "___" for any date,
   patient name, and location.

6. Use appropriate medical terminology
   and concise language to ensure
```

clarity and professionalism.

7. Do not be wordy; be concise if possible.

8. Do not include the word "optional" in the result if they are included. If they are not included, just omit those sections.

9. Do not copy patient information verbatim; paraphrase and use the structure template to fit in the details.

10. All the section headers must be from the template, not from the patient information.

11. Do not fabricate details not present in the patient information.

12. Use section headers for each major medical issue, starting with a hashtag #, do not use * for section header.

13. Use bullet points to highlight key actions, medication changes, or critical clinical decisions, starting with a hyphen -. Do not use * or +.

14. Ensure that each major issue or condition has its own section header if there is enough content related to it, even if briefly mentioned.

15. Write in a narrative style for each section, providing a detailed account of the patient's condition, treatment, and outcomes.

16. Employ medical abbreviations and terminology appropriately to convey information efficiently.

17. Start the output with "Brief hospital course:"

Example structure for the brief hospital course: {structure}.
Patient information: {context}.

The template for DI is below. This is fed to the DI prompt as the "structure" variable.

1. Greeting: "Dear [Title] ___,", "HospitalExperience": "It was a pleasure taking care of you at ___.",

2. AdmissionReason: "Title": "WHY WAS I ADMITTED TO THE HOSPITAL?", "Details": "[ReasonForAdmission]" ,

3. InHospitalActivities: "Title": "WHAT HAPPENED WHILE I WAS IN THE HOSPITAL?", "Details": "[ActivitiesDuringStay]" ,

4. DischargeAdvice: "Title": "WHAT SHOULD I DO WHEN I GO HOME?", "Instructions": "[PostDischargeInstructions]" ,

5. Closing: "We wish you the best!", "CareTeam": "Your ___ Team"

The prompt for DI is:

You are tasked with drafting a "Discharge Instructions" section for a patient's discharge letter as a medical professional. The instructions should succinctly summarize the key points of the patient's hospital stay and post-discharge care clearly and easily for the patient to follow.

DI Instructions:

1. Use the JSON template provided to structure the discharge instructions.

2. Do not include explicit section headers in the final text, such as "Greeting" or "Hospital Experience".

3. Do not include any placeholder such as "[]" in the result.

4. Include the title in the template.

5. Integrate medication information narratively, mentioning specific medications only when discussing their relevance to the patient's ongoing care and follow-up instructions.

6. Do not list medications; describe how they contribute to the patient's treatment plan.

7. The total word count should be around {words} words, focusing on essential instructions relevant to the patient's care.

8. Use "___" to anonymize any date, patient name, and location.

9. Clearly specify any medication changes, follow-up appointments, and additional care instructions using placeholders where specific details are to be inserted.

10. Employ a professional yet empathetic tone to ensure clarity and approachability.

11. Integrate medical terminology appropriately, ensuring it is understandable to a layperson.

12. Start the output with a polite greeting and conclude with well-wishes or a thank you message.

Example structure for the discharge instructions: {structure}.
Patient information: {context}.

4 Results

The Llama3 model was downloaded from the Ollama model repository with the model ID "llama3:8b-instruct-q8_0". We utilized the LangChain framework for retrieval, template building, and model calling. All experiments were conducted on a T4 GPU with 16GB memory, using the Microsoft Azure platform's "Standard NC4 as T4 v3 (4 vCPUs, 28 GiB memory)" configuration.

We compared several approaches:

1. *Baseline with Random Shuffling*: We shuffled the "hadm_id" column, a unique identifier for each patient's discharge letter, assigning a random target section to each "hadm_id". This random selection comes from the same distribution as the training data but without the actual content of the input text.

2. *Baseline with RAG Retrieval*: We used the retrieved target sections directly. This result can be similar to the target, but the details can differ from the real input.

3. *Fixed Target Word Count*: We set a fixed word count of 420 for BHC and 100-200 for DI in the prompt.

4. *Proposed Method*: Our method combines retrieved target word counts with a structured template.

Table 2 presents the evaluation metrics from the Codabench platform (Xu et al., 2024), including BLEU-4 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang et al., 2019), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). The random shuffle yielded the lowest scores across all metrics, indicating poor performance. Using the retrieved target section directly resulted in the highest BLEU score. The fixed word count approach achieved higher Align and MEDCON scores than the retrieved target section but had lower scores for other metrics. Our proposed method, which combines the retrieved word count and structured template, achieved the highest scores across all metrics except BLEU. The lower BLEU score for the proposed method is due to BLEU's heavy penalty for deviations from exact wording. In contrast, the higher ROUGE scores indicate our method effectively captures the essential content, even with varied wording. We also measured the generation time for each section. The average time to generate one BHC was 16.67 seconds, and one DI was 16 seconds.

5 Unsuccessful Attempts

We also explored several alternative approaches for this task, but they yielded unsatisfactory results:

1. *Style Transfer Using Retrieved Target Section*: We asked the LLM to use the style of the retrieved target section to fit the patient context. However, the Llama3 8B model often used the target section directly, failing to infer the style and remove the original content. This could be due to the weaker reasoning ability of the 8B model compared to the 70B model with better reasoning ability.

2. *Two-Step Style Transfer*:

   (a) Firstly, extract a template from the target section.

   (b) Secondly, fill in the patient content into the template (this step can also be split into several smaller steps).

	bleu	rouge1	rouge2	rougel	bertscore	meteor	align	medcon	overall
random shuffle	0.01	0.183	0.025	0.105	0.226	0.23	0.109	0.1	0.124
RAG retrieved target	**0.041**	0.286	0.061	0.172	0.293	0.297	0.167	0.203	0.19
fixed target word	0.017	0.296	0.055	0.159	0.256	0.285	0.187	0.221	0.185
retrieved word count	0.024	**0.377**	**0.106**	**0.205**	**0.3**	**0.332**	**0.174**	**0.254**	**0.221**

Table 2: The evaluation results from the Codabench platform. The random shuffle method yielded the lowest scores, while our final retrieval approach to determine the target word count achieved the highest scores across most metrics.

However, the extracted templates were not always reliable, and this method took twice as long as the curated template approach. Consequently, we opted to curate the templates rather than relying on the LLM manually.

3. *Predicting Target Section Word Count*: We tested several methods to predict the total word count of the target section, including fitting a random forest classifier by aggregating over 100 features from other MIMIC-IV tables and fitting log-normal distributions. These methods also proved inadequate. Table 3 shows the random forest classifier results for BHC with word count classes greater than 450, with an F1 score of 0.45. Figure 3 lists the top 10 features, including the number of lab tests, diagnoses, and total hospital duration. The classifier achieved an F1 score of 0.49 for word counts greater than 280 for the DI section, as shown in Table 4, with different section word counts being the top features in Figure 4.
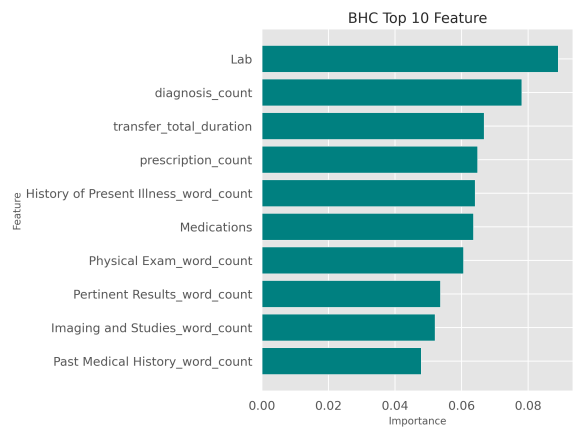

Figure 3: The top 10 features for the BHC classifier. WC: word count. The total number of lab tests, diagnosis, and total duration in the hospital are the top 3 features.
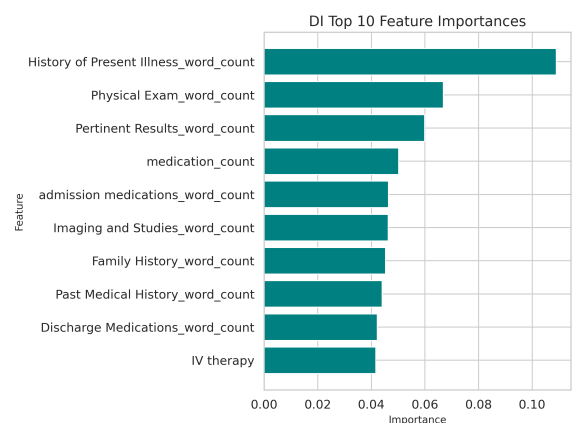
	precision	recall	f1-score	support
<450	0.818	0.926	0.869	18965
>450	0.610	0.359	0.452	6087

Table 3: BHC random forest classifier results for BHC word count above and below 450. The f1-score is 0.45 for the class with more than 450 words, which is not accurate enough.

6 Conclusion

In this paper, we present a resource-friendly approach to automating the generation of the "Brief Hospital Course" and "Discharge Instructions" sections in discharge letters using the Llama3 8B quantized model. Our zero-shot template-based method and Retrieval-Augmented Generation produce high-quality, contextually appropriate

Figure 4: The top 10 features for the DI classifier. WC: word count. The word count of different segments is ranking high.

	precision	recall	f1-score	support
<280	0.864	0.964	0.911	20143
>280	0.716	0.377	0.494	4909

Table 4: DI random forest classifier result for DI word count above and below 280. The f1-score is 0.49 for the class with more than 280 words, which is not accurate enough.

summaries. However, we observe a lower BLEU score due to the different wording between the method's result and the target sections. Ensuring the reliability and accuracy of generated content remains a significant challenge. Future work will focus on enhancing model reasoning capabilities, improving dynamic template extraction, and integrating robust validation mechanisms to verify medical accuracy. The code for this work is shared on `https://github.com/ruiguo-bio/discharge_me`, covering aggregating additional tables, segmentation of the discharge letters, RAG for the two target sections, and the random forest classifier for the target section words prediction.

7 Limitations and Future Work

1. We would like to perform a more thorough evaluation to ensure that the model's generated content is clinically relevant and does not include false or harmful information. This evaluation could be extended to understanding the strengths and weaknesses of language models for the challenge task.

2. We create a template by sampling target sections with word counts close to the median. However, the length and structure of real target sections can vary significantly from our template. Our approach could be improved by predicting the target word count more precisely or by sampling different templates depending on the word count.

3. We would like to test a wider range of language models and thoroughly compare different methods of providing relevant context to the language model, including different methods of Retrieval-Augmented Generation (RAG) and prompt engineering.

8 Ethical Statement

All the data used in the experiments are downloaded from the PhysioNet after completing the required CITI training and credentialing process. Beyond the general potential ethical considerations of using LLMs to automatically process and generate clinical text (including bias, fairness, transparency and accountability), there are no specific ethical issues raised by the particular methodologies or data presented in this research.

References

AI@Meta. 2024. Llama 3 model card. `https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md`. Accessed: 2024-05-12.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Elliot Bolton, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, Jonathan Frankle, Percy Liang, Michael Carbin, et al. 2024. Biomedlm: A 2.7 b parameter language model trained on biomedical text. *arXiv preprint arXiv:2403.18421*.

Wafaa S El-Kassas, Cherif R Salama, Ahmed A Rafea, and Hoda K Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert systems with applications*, 165:113679.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, Matthew M Churpek, and Majid Afshar. 2023a. Overview of the problem list summarization (probsum) 2023 shared task on summarizing patients' active diagnoses and problems from electronic health record progress notes. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2023, page 461. NIH Public Access.

Yanjun Gao, Ruizhe Li, John Caskey, Dmitriy Dligach, Timothy Miller, Matthew M Churpek, and Majid Afshar. 2023b. Leveraging a medical knowledge graph into large language models for diagnosis prediction. *arXiv preprint arXiv:2308.14321*.

Yanjun Gao, Timothy Miller, Dongfang Xu, Dmitriy Dligach, Matthew M Churpek, and Majid Afshar. 2022. Summarizing patients' problems from hospital progress notes using pre-trained sequence-to-sequence models. In *Proceedings of COLING. International Conference on Computational Linguistics*, volume 2022, page 2979. NIH Public Access.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023c. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Yuting He, Fuxiang Huang, Xinrui Jiang, Yuxiang Nie, Minghao Wang, Jiguang Wang, and Hao Chen. 2024.

Foundation model for advancing healthcare: Challenges, opportunities, and future directions. *arXiv preprint arXiv:2404.03264*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Leo Anthony Celi, Roger Mark, and Steven Horng. 2023a. Mimic-iv-ed (version 2.2).

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023b. Mimic-iv (version 2.2).

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023c. Mimic-iv-note: Deidentified free-text clinical notes (version 2.2).

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *arXiv preprint arXiv:2402.10373*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Darren Liu, Cheng Ding, Delgersuren Bold, Monique Bouvier, Jiaying Lu, Benjamin Shickel, Craig S Jabaley, Wenhui Zhang, Soojin Park, Michael J Young, et al. 2024. Evaluation of general large language models in contextually assessing semantic concepts extracted from adult critical care electronic health record notes. *arXiv preprint arXiv:2401.13588*.

Ming Liu, Dan Zhang, Weicong Tan, and He Zhang. 2023. Deakinnlp at probsum 2023: Clinical progress note summarization with rules and language modelsclinical progress note summarization with rules and languague models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 491–496.

OpenAI. 2024. Chatgpt. https://openai.com/chatgpt. Accessed: 2024-05-12.

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

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. *Journal of machine learning research*, 21(140):1–67.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023a. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023b. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Christine Sinsky, Lacey Colligan, Ling Li, Mirela Prgomet, Sam Reynolds, Lindsey Goeders, Johanna Westbrook, Michael Tutty, and George Blike. 2016. Allocation of physician time in ambulatory practice: a time and motion study in 4 specialties. *Annals of internal medicine*, 165(11):753–760.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2023. Medagents: Large language models as collaborators for zero-shot medical reasoning. *arXiv preprint arXiv:2311.10537*.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, et al. 2023. Clinical text summarization: Adapting large language models can outperform human experts. *Research Square*.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. 2024. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, page ocae045.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Cyril Zakka, Rohan Shad, Akash Chaurasia, Alex R. Dalal, Jennifer L. Kim, Michael Moor, Robyn Fong, Curran Phillips, Kevin Alexander, Euan Ashley, Jack Boyd, Kathleen Boyd, Karen Hirsch, Curt Langlotz, Rita Lee, Joanna Melia, Joanna Nelson, Karim Sallam, Stacey Tullis, Melissa Ann Vogelsong, John Patrick Cunningham, and William Hiesinger. 2024. Almanac — retrieval-augmented language models for clinical medicine. *NEJM AI*, 1(2):AIoa2300068.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Hongbo Zhang, Junying Chen, Feng Jiang, Fei Yu, Zhihong Chen, Guiming Chen, Jianquan Li, Xiangbo Wu, Zhang Zhiyi, Qingying Xiao, Xiang Wan, Benyou Wang, and Haizhou Li. 2023. HuatuoGPT, towards taming language model to be a doctor. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10859–10885, Singapore. Association for Computational Linguistics.

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

Zhenyu Zhang, Benlu Wang, Weijie Liang, Yizhi Li, Xuechen Guo, Guanhong Wang, Shiyan Li, and Gaoang Wang. 2024. Sam-guided enhanced fine-grained encoding with mixed semantic learning for medical image captioning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1731–1735. IEEE.