Ixa-Med at Discharge Me! Retrieval-Assisted Generation for Streamlining Discharge Documentation

Jordan Koontz
HiTZ Center - Ixa
UPV/EHU
jkoontz001@ikasle.ehu.eus

Maite Oronoz
HiTZ Center - Ixa
UPV/EHU
maite.oronoz@ehu.eus

Alicia Pérez
HiTZ Center - Ixa
UPV/EHU
alicia.perez@ehu.eus

Abstract

In this paper we present our system for the BioNLP ACL'24 "Discharge Me!" task on automating discharge summary section generation. Using Retrieval-Augmented Generation, we combine a Large Language Model (LLM) with external knowledge to guide the generation of the target sections. Our approach generates structured patient summaries from discharge notes using an instructed LLM, retrieves relevant "Brief Hospital Course" and "Discharge Instructions" examples via BM25 and SentenceBERT, and provides this context to a frozen LLM for generation. Our top system using SentenceBERT retrieval achieves an overall score of 0.183, outperforming zero-shot baselines. We analyze performance across different aspects, discussing limitations and future research directions.

1 Introduction

Generating detailed clinical notes in Electronic Health Records (EHRs) is a time-consuming task that can lead to clinician burnout and operational inefficiencies in healthcare systems. The BioNLP ACL'24 Shared Task, "Discharge Me!" (Xu et al., 2024), aims to automate the generation of critical discharge summary sections using natural language processing (NLP). While large language models (LLMs) like GPT-4 (OpenAI et al., 2024) and Llama-3 (Meta, 2024) have advanced NLP capabilities, they can produce hallucinations when encountering out-of-distribution queries (Zhang et al., 2023).

The Retrieval-Augmented Generation (RAG) framework aims to mitigate hallucinations in large language models (LLMs) by combining external knowledge retrieval with LLM generation (Lewis et al., 2020; Ma et al., 2023). A Naive RAG approach involves indexing data into vectors, retrieving relevant vectors for a given query, and providing the retrieved context to a frozen LLM. However, this naive implementation often suffers from limitations in retrieval precision, recall, and generation quality. Notwithstanding, we evaluate the efficacy of a Naive RAG framework for the "Discharge Me!" task. Section 3 describes our system methodology and presents our results. Section 4 analyzes the limits of our approach and outlines prospective research areas for improvement.

2 Task Description

The BioNLP ACL'24 Shared Task, 'Discharge Me!", focuses on streamlining the clinical documentation process by automating the generation of two critical sections in discharge summaries: "Brief Hospital Course" and "Discharge Instructions". By reducing the time and effort clinicians expend on writing these detailed notes in electronic health records (EHRs), we can alleviate administrative burden, minimize clinician burnout, and ultimately improve operational efficiencies and patient care quality.

2.1 Dataset Description

For this shared task, participants are provided with a dataset derived from the MIMIC-IV-Note and MIMIC-IV-ED submodules. The shared task dataset contains $109,168$ visits to the Emergency Department (ED). Each visit consists of chief complaints documented by ED physicians, ICD diagnosis codes (either ICD-9 or ICD-10), at least one associated radiology report, and the full discharge summary text which includes the "Brief Hospital Stay" and "Discharge Instructions" sections, among others. The dataset is split into training ($68,785$ samples), validation ($14,719$ samples), phase I testing ($14,702$ samples), and phase II testing ($10,962$ samples). The chief goal is to develop a system that can generate the two target sections given the available data for each visit.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 658–663
August 16, 2024. ©2024 Association for Computational Linguistics

2.2 Evaluation

For evaluating the participants' systems, a hidden subset of 250 samples from the test phase I and test phase II is used. The evaluation framework is composed of a diverse array or metrics that capture both textual similarity and factual correctness aspects of the generated texts. Concretely, the following metrics are used: BLEU-4 (Papineni et al., 2002), ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). The final overall system score is a composite measure derived by combining the scores across all evaluation metrics and both target sections.

3 Methods & Results

3.1 Structured Patient Summary Generation

The first step in our approach involved generating structured JSON summaries from the patient discharge summaries. This process extracted and organized relevant information for generating the "Brief Hospital Course" and "Discharge Instructions" sections, critical components of discharge documentation..

We leveraged the capabilities of an LLM, specifically[1] the `mistralai/Mistral-7B-Instruct-v0.2` model, to facilitate this preprocessing step. The `vllm` (Kwon et al., 2023) library was utilized for interacting with the LLM, while the `lmformatenforcer`[2] library ensured character-level parsing and schema enforcement.

Our pipeline consisted of the following steps:

1. **Data Masking**: To ensure that the LLM generated summaries based solely on the available information, we masked the "Discharge Instructions" and "Brief Hospital Course" sections from the input discharge summaries.

2. **Prompt and Schema Design**: A carefully crafted prompt template, presented in table 1, was designed to guide the LLM in generating structured JSON summaries. Additionally, we defined a Pydantic data model to serve as the schema for the desired JSON output format.

3. **LLM Inference**: For each masked discharge summary, we employed the LLM to generate two structured JSON summaries using the defined prompt template. One summary excluded the "Discharge Instructions section", while the other omitted the "Brief Hospital Course section".

The structured summaries (mentioned in step 2) aimed to captured essential patient information like demographics, medical history, reason for admission, findings, treatments, and discharge condition. This structured input aimed to reduce noise and prevent hallucinations in subsequent generation steps.

3.2 Zero-shot Generation

We first established a baseline by conducting experiments with a zero-shot generation approach, using the`mistralai/Mistral-7B-Instruct-v0.2` model. The primary objective was to generate "Discharge Instructions" and "Brief Hospital Course" texts directly from the patient information in JSON format, without relying on fine-tuning or RAG techniques.

To guide the language model, we designed two ad-hoc prompt templates: one for "Discharge Instructions" and another for "Brief Hospital Course" summaries. These templates, created by us and not defined by medical professionals, included detailed instructions and placeholders for the patient JSON data. The "Discharge Instructions" template provided guidelines for generating a 300-400 word summary, covering aspects like greeting the patient, summarizing the hospital course, listing medications, and providing follow-up instructions. The "Brief Hospital Course" template aimed to produce a 400-600 word text, organized by active and inactive issues or organ systems, summarizing diagnostic findings, treatments, procedures, and the patient's response to treatment.

One notable limitation of using these ad-hoc prompt templates was the lack of grounding in external knowledge sources. The model relied solely on the information provided in the patient JSON, which may not always be comprehensive or sufficient for generating accurate and detailed summaries. Consequently, the generated summaries could sometimes miss important details, include irrelevant information, or lack the necessary context for certain medical terms or procedures.

To address these limitations and enhance the quality of the generated summaries, we explored

[1] The LLM is available at: `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2`

[2] The library is available at: `https://github.com/noamgat/lm-format-enforcer`

Instruction: Generate a detailed "Brief Hospital Summary" in a structured format following a provided schema. The "Brief Hospital Summary" should include information about the patient's demographics, primary reason for admission, chief complaint, relevant past medical history, diagnostic findings, diagnosis, treatments provided, patient's response to treatment, consultations with other specialties, medication changes and adjustments, discharge condition and disposition, and follow-up plans, follow-up care instructions, and scheduled appointments.

Ensure that the extracted information is concise, clear, and captures the essential aspects of the patient's hospital course. Review the organized information for completeness and accuracy, and refine or rephrase any unclear or ambiguous information.

Schema: `patient_demographics`, `age`, `gender`, `name`, `reason_for_admission`, `chief_complaint`, `relevant_history`, `diagnosis`, `diagnostic_findings`, `imaging`, `labs`, `procedures`, `treatments`, `consultations`, `medications`, `medication_changes`, `discharge_condition`, `discharge_disposition`, `follow_up_instructions`.

Context:

[Clinical note(s) will be provided here]

Table 1: Patient Hospital Summary Prompt Template

RAG implementations, which are discussed in the subsequent section.

3.3 Retrieval Augmented Generation

To further enhance the LLM's generative capabilities, we sought to combine its parametric memory with non-parametric memory by enriching the prompt's context with relevant examples retrieved from an external dataset. Specifically, given an input $\mathbf{x}$ (a patient's JSON summary), we employ retrieval functions (defined later) to fetch the k most similar discharge instructions or brief hospital course texts from the Discharge Me training set $\mathcal{D}$. This process generates an k-shot prompt, thereby providing the LLM with additional context to inform its responses.

During the retrieval process, we calculate the relevance scores for all examples $d \in \mathcal{D}$ using two retrieval functions. For retrieval function A (BM25) (Robertson et al., 1995) the relevance score of a document d to a query x is calculated based on the frequency of query terms in the document, the document length, and the rarity of the query terms. For retrieval function B (SentenceBERT) (Reimers and Gurevych, 2019): the relevance score is computed as in expression (1).

$$s_B(\mathbf{x}, d) = \cos(\text{SentenceBERT}(\mathbf{x}), \text{SentenceBERT}(d)) \quad (1)$$

where SentenceBERT is a pre-trained model that encodes the input x and document d into dense vector representations. Specifically, we use the pre-trained `pritamdeka/S-PubMedBert-MS-MARCO` model (Deka et al., 2022)[3]. The cosine similarity between the two vector representations is used to measure the semantic similarity between the input and the document.

3.4 Results

We evaluated four different systems: FDS+SBERT-RAG, PS+Zero-shot, PS+SBERT-RAG, and PS+BM25-RAG. FDS+SBERT-RAG employed the full patient discharge summary (FDS) as input, along with the RAG framework and SentenceBERT (SBERT) for retrieval. PS+Zero-shot used the patient summary (PS) as input but performed inference using only the prompt instructions without RAG. PS+SBERT-RAG utilized the PS as input, also with the RAG framework and SBERT for retrieval. PS+BM25-RAG used the PS as input, with the RAG framework and BM25 as the retrieval function. Our top-performing system, PS+SBERT-RAG, attained an overall score of 0.183 at the competition deadline, exhibiting the potential of combining LLMs with RAG techniques for generating clinical notes. In contrast, our worst-performing system, PS+Zero-shot, obtained an overall score of 0.172, highlighting the performance uplift provided by our RAG methodology compared to the zero-shot approach. Table 2 presents our n-gram overlap metrics, table 3 our semantic similarity metrics, table 4 our factual alignment and clinical concept accuracy metrics, and table 5 our systems' overall scores.

[3] The model is available at: `https://huggingface.co/pritamdeka/S-PubMedBert-MS-MARCO`

System	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L
PS+Zero-shot	0.011	0.263	0.052	0.133
PS+SBERT-RAG	0.016	0.259	0.057	0.144
PS+BM25-RAG	0.018	0.244	0.051	0.141
FDS+SBERT-RAG	0.02	0.286	0.076	0.156

Table 2: N-gram Overlap Metrics

System	BERTScore	Meteor
PS+Zero-shot	0.238	0.275
PS+SBERT-RAG	0.282	0.284
PS+BM25-RAG	0.283	0.281
FDS+SBERT-RAG	0.261	0.290

Table 3: Semantic Similarity Metrics

System	AlignScore	MEDCON
PS+Zero-shot	0.210	0.196
PS+SBERT-RAG	0.210	0.215
PS+BM25-RAG	0.192	0.196
FDS+SBERT-RAG	0.170	0.219

Table 4: Factual Alignment and Clinical Concept Accuracy

System	Overall
PS+Zero-shot	0.172
PS+SBERT-RAG	0.183
PS+BM25-RAG	0.175
FDS+SBERT-RAG	0.185

Table 5: Overall Evaluation Results

In the context of n-gram overlap metrics, PS+SBERT-RAG exhibited suboptimal performance, achieving scores of 0.016 for BLEU-4, 0.259 for ROUGE-1, 0.057 for ROUGE-2, and 0.144 for ROUGE-L. These results suggest that the generated texts demonstrated limited lexical overlap with the reference summaries, implying potential challenges in accurately capturing relevant details and phrasing inherent in the gold standard.

On the other hand, PS+SBERT-RAG performed more favorably in semantic similarity metrics, achieving scores of 0.282 for BERTScore and 0.284 for Meteor. The BERTScore and Meteor results indicate that the generated texts from PS+SBERT-RAG exhibited high semantic equivalence with the reference summaries, suggesting its ability to capture the underlying meaning and context accurately, despite potential lexical differences.

Furthermore, PS+SBERT-RAG achieved a score of 0.21 for AlignScore, which evaluates the degree of factual alignment between the generated and reference texts. It also obtained a MEDCON score of 0.215, specifically gauging the accuracy and consistency of clinical concepts mentioned. These scores demonstrate the system's proficiency in generating clinically relevant and factually consistent content.

We also explored utilizing the full patient discharge summary (FDS) as input, along with the RAG framework, which we refer to as FDS+SBERT-RAG. We opted to use the Sentence-BERT retrieval function as it performed better than BM25 when using the PS inputs. Although we did not have the opportunity to finalize the results before the competition deadline, we found that FDS+SBERT-RAG achieved even better performance than PS+SBERT-RAG, with scores of 0.02 for BLEU-4, 0.286 for ROUGE-1, 0.076 for ROUGE-2, and 0.156 for ROUGE-L in the n-gram overlap metrics. FDS+SBERT-RAG also performed well in the semantic similarity metrics, scoring 0.261 for BERTScore, 0.29 for Meteor, 0.17 for AlignScore, and 0.219 for MEDCON. The improved performance of FDS+SBERT-RAG suggests that providing the model with more comprehensive patient information can further enhance its ability to generate accurate and clinically relevant summaries.

4 Conclusion

Our work explored a Retrieval-Augmented Generation approach for the "Discharge Me!" shared task on automating the generation of "Brief Hospital Course" and "Discharge Instructions" sections. We grounded a Large Language Model with structured patient summaries and retrieved relevant examples from the challenge training data set, aiming to mitigate hallucinations and enhance generation quality.

While our grounded approach demonstrated potential in generating coherent summaries, several areas exist for performance improvement. Fine-tuning the pipeline on "Brief Hospital Course" and "Discharge Instructions" sections could better align generated text with domain-specific language patterns. Incorporating constrained decoding or post-processing could improve n-gram overlap with references. Optimizing retrieval for stylistic similarity could indirectly benefit n-gram metrics. Moreover, metrics like MEDCON could be improved by retrieving Unified Medical Language System (UMLS) concept-rich examples or integrating UMLS databases during retrieval/generation. Exploring advanced RAG architectures with iterative retrieval and multi-step reasoning could address Naive RAG limitations.

Limitations

Our approach faced challenges due to maximum sequence length constraints. The retrieval encoder (SentenceBERT) had a 350-token limit, leading to the loss of relevant contextual information. Full discharge summaries exceeded the LLM's context length, resulting in omitted details, and likely hindered performance due to the loss of important contextual information. Additionally, our system did not effectively leverage the available radiology reports and ICD-9/10 diagnosis codes, which could potentially enhance the understanding of patient conditions and improve generation quality. The adhoc prompts, created without medical professionals' guidance, may have lacked necessary context and guidelines to generate accurate and comprehensive "Brief Hospital Course" and "Discharge Instructions" sections. The lack of domain adaptation for the LLM and SentenceBERT retrieval model could lead to issues understanding and generating domain-specific terminology and clinical concepts. By combining domain knowledge, task-specific fine-tuning, architectural enhancements, addressing sequence length limitations, and effectively integrating complementary data sources like radiology reports and diagnosis codes, we believe more accurate and reliable generation systems can be developed, contributing to improved patient care and reduced administrative burdens.

5 Acknowledgements

This work was elaborated within the framework of LOTU (TED2021-130398B-C22) funded by MCIN/AEI/10.13039/501100011033, European Commission (FEDER), and by the European Union "NextGenerationEU"/PRTR. Besides, this work was partially funded by the Spanish Ministry of Science and Innovation (EDHIA PID2022-136522OB-C22); by the Basque Government (IXA IT-1570-22) and by EXTEPA within Misiones Euskampus 2.0.

References

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, Ann Arbor, Michigan. Association for Computational Linguistics.

Pritam Deka, Anna Jurek-Loughrey, and P Deepak. 2022. Improved methods to aid unsupervised evidence-based fact checking for online health news. *Journal of Data Intelligence*, 3(4):474–504.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

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.

Xinbei Ma, Yeyun Gong, Pengcheng He, hai zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`. Accessed: 2024-05-10.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, and Ilge Akkaya et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Preprint*, arXiv:1908.10084.

Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at trec-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. Gaithersburg, MD: NIST.

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.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.