Generative Language Models for Personalized Information Understanding

Pengshan Cai
University of Massachusetts Amherst

Follow this and additional works at: https://scholarworks.umass.edu/dissertations_2

Part of the Computational Engineering Commons

Recommended Citation
https://doi.org/10.7275/36214447 https://scholarworks.umass.edu/dissertations_2/3036

This Open Access Dissertation is brought to you for free and open access by the Dissertations and Theses at ScholarWorks@UMass Amherst. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of ScholarWorks@UMass Amherst. For more information, please contact scholarworks@library.umass.edu.
GENERATIVE LANGUAGE MODELS FOR
PERSONALIZED INFORMATION UNDERSTANDING

A Dissertation Presented
by
PENGSHAN CAI

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

February 2024

Manning College of Information and Computer Sciences
GENERATIVE LANGUAGE MODELS FOR
PERSONALIZED INFORMATION UNDERSTANDING

A Dissertation Presented
by
PENGSHAN CAI

Approved as to style and content by:

____________________________________
Hong Yu, Chair

____________________________________
Mohit Iyyer, Member

____________________________________
Laure Thompson, Member

____________________________________
Fei Liu, Outside Member

____________________________________
Ramesh K. Sitaraman, Associate Dean for
Educational Programs and Teaching
Manning College of Information and Computer
Sciences
ABSTRACT

GENERATIVE LANGUAGE MODELS FOR PERSONALIZED INFORMATION UNDERSTANDING

FEBRUARY 2024

PENGSHAN CAI
B.E., WUHAN UNIVERSITY
M.Sc., UNIVERSITY OF CHINESE ACADEMY OF SCIENCES
Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Hong Yu

A major challenge in information understanding stems from the diverse nature of the audience, where individuals possess varying preferences, experiences, educational and cultural backgrounds. Consequently, adopting a one-size-fits-all approach to provide information may prove suboptimal. While prior research has predominantly focused on delivering pre-existing content to users with potential interests, this thesis explores generative language models for personalized information understanding. By harnessing the potential of generative language models, our objective is to generate novel personalize content for individual users. As a result, users from diverse backgrounds can be provided with content that are tailored for their need and better aligns with their interests.

The crux of this research hinges on addressing the following two aspects: 1. Personalized Content: How to harness user profiles to create tailored content for
individual users; 2. **Effective Communication**: How to engage with users in order to proficiently convey information. For the first aspect, i.e. personalized content, we explored personalized news headline generation. By analyzing users’ reading history, our proposed framework identifies perspectives that users are interested in, which can further guide generating news headlines that are attractive to users. For the second aspect, i.e. effective communication, we developed personalized reading assistive agent, which assist users understand complex information in news article or academic documents through conversations. Compared to reading, obtaining information through conversations is more interactive and requires shorter attention span.

We further incorporate the above aspects in personalized information systems in a real-life scenario, i.e. patient education. Specifically, we propose a novel after-visit summaries (AVS) writing assistant. After-visit summaries notes are documents given to patients to help them understand their clinical visits and disease self-management. Our approach not only automatically generates AVS drafts, but also detects potential errors in the generated drafts, allowing physicians to revise and produce AVS notes with higher efficiency and accuracy. Moreover, we present PaniniQA, a patient-centric interactive question answering system designed to help patients understand their discharge instructions. PaniniQA first identifies important clinical content from patients’ discharge instructions and then formulates personalized educational questions for distinctive patients. In addition, PaniniQA is also equipped with answer verification functionality to provide timely feedback to correct patients’ misunderstandings.

Overall, we aspire to contribute to the advancement of information dissemination techniques, promoting a more inclusive and effective means of communication in our information-driven world.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xii</td>
</tr>
<tr>
<td><strong>CHAPTER</strong></td>
<td></td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Generating User-Engaging News Headlines</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Learning as Conversation Reading Assistant</td>
<td>6</td>
</tr>
<tr>
<td>1.4 Assisting Physicians in Writing Patient After-Visit Summaries</td>
<td>9</td>
</tr>
<tr>
<td>1.5 Enhancing Patient Education Through Interactive Question Answering</td>
<td>11</td>
</tr>
<tr>
<td>2. LITERATURE REVIEW</td>
<td>15</td>
</tr>
<tr>
<td>2.1 Language Modeling</td>
<td>16</td>
</tr>
<tr>
<td>2.2 Applying LMs for Information Personalization</td>
<td>17</td>
</tr>
<tr>
<td>2.3 Applying LMs for Automatic Summarization</td>
<td>19</td>
</tr>
<tr>
<td>2.4 Applying LMs for Content Grounded Dialogue System</td>
<td>21</td>
</tr>
<tr>
<td>2.5 Applying LMs for Patient Education</td>
<td>22</td>
</tr>
<tr>
<td>3. GENERATING PERSONALIZED NEWS HEADLINES</td>
<td>24</td>
</tr>
<tr>
<td>3.1 Approach</td>
<td>24</td>
</tr>
<tr>
<td>3.1.1 Signature Phrases Identification</td>
<td>25</td>
</tr>
<tr>
<td>3.1.2 User Signature Selection</td>
<td>25</td>
</tr>
<tr>
<td>3.1.3 Signature-Oriented Headline Generation</td>
<td>27</td>
</tr>
<tr>
<td>3.2 Corpora Processing</td>
<td>27</td>
</tr>
<tr>
<td>3.3 Experiments</td>
<td>30</td>
</tr>
</tbody>
</table>
3.3.1 Baseline Methods ................................................. 30
3.3.2 Objective Evaluation ............................................ 30
3.3.3 Subjective Evaluation ............................................ 33
3.3.4 Ablation Study .................................................. 35

3.4 Conclusion .......................................................... 37

4. LEARNING AS CONVERSATION: DIALOGUE SYSTEMS
REINFORCED FOR INFORMATION ACQUISITION .............. 39

4.1 Approach ............................................................. 39
  4.1.1 Model Architecture ............................................. 40
  4.1.2 Phase 1: Supervised Pre-Tuning ............................... 41
  4.1.3 Phase 2: Unsupervised Self-Play Fine-tuning ............... 42
  4.1.4 Reward Functions ............................................. 43
    4.1.4.1 Coverage .................................................. 43
    4.1.4.2 Coherence .................................................. 43
    4.1.4.3 Mixed Reward ............................................. 44

4.2 Experimental Setting .............................................. 45
  4.2.1 Datasets ....................................................... 45
  4.2.2 Comparison Models .......................................... 46
  4.2.3 Evaluation Metrics .......................................... 47

4.3 Objective Results ................................................ 49
4.4 Subjective Results ............................................... 52
4.5 Conclusion .......................................................... 54

5. GENERATION OF PATIENT AFTER-VISIT SUMMARIES
TO SUPPORT PHYSICIANS .......................................... 56

5.1 Summarization ...................................................... 56
5.2 Error Detection ..................................................... 58
  5.2.1 Type I Error: Hallucination .................................. 58
  5.2.2 Type II Error: Missing Content ............................... 60

5.3 Experiments ........................................................ 62
  5.3.1 Dataset ......................................................... 62
  5.3.2 Evaluation Metrics ........................................... 63
  5.3.3 Summarization Results ....................................... 66
  5.3.4 Error Detection Results ...................................... 68
6. PATIENT-CENTRIC QUESTION ANSWERING FOR PATIENT EDUCATION .......................... 72

6.1 Question Answering in the GPT Era ................................................................. 72
6.2 Extracting Salient Medical Events ................................................................. 74
   6.2.1 Event and Relation Identification ............................................................... 75
   6.2.2 Question Generation ..................................................................................... 76
6.3 Data Annotation ................................................................................................. 77
6.4 Evaluating Information Extraction ..................................................................... 80
   6.4.1 IE Evaluation Settings ................................................................................... 80
   6.4.2 IE Evaluation Results ..................................................................................... 81
6.5 Evaluating Patient Education ............................................................................. 82
   6.5.1 Human Evaluation Settings .......................................................................... 82
   6.5.2 Patient Evaluation Measurements ............................................................... 84
   6.5.3 GPT-4’s Automatic Evaluation Settings ....................................................... 86
   6.5.4 Synthesized Dataset for Evaluation ............................................................. 86
   6.5.5 Physician Evaluation Results ....................................................................... 86
   6.5.6 Patient Evaluation Results ............................................................................ 88
   6.5.7 GTP-4’s Automatic Evaluation Results ....................................................... 90
   6.5.8 Heuristic Evaluation of Conversation Log .................................................. 90
6.6 Conclusion ......................................................................................................... 92
6.7 Ethical Considerations ....................................................................................... 92

7. CONCLUSIONS & FUTURE WORKS ................................................................. 94

8. APPENDIX ......................................................................................................... 96

BIBLIOGRAPHY .................................................................................................... 97
### LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Statistics of the datasets. For each corpus, the synthesized user dataset is used for training the signature phrase selection module and evaluating the entire system, while the headline generation dataset is used for training the headline generation module (it does not have a test set because the generation step is evaluated in the entire system using the test set of synthesized user dataset).</td>
<td>28</td>
</tr>
<tr>
<td>3.2 Objective evaluation results of all methods. “-F” means using the fine-tuned signature phrase encoder, headline encoder and user history encoder, while “-N” means using the naive DPR models as encoders. “REC Score” refers to recommendation score. Vanilla approaches do not consider human preference.</td>
<td>31</td>
</tr>
<tr>
<td>3.3 Examples of generated headlines.</td>
<td>34</td>
</tr>
<tr>
<td>3.4 The impact of different signature phrase selectors.</td>
<td>35</td>
</tr>
<tr>
<td>3.5 Result of generated headlines for newsroom articles when synthesized users have different number of interest phrases.</td>
<td>35</td>
</tr>
<tr>
<td>3.6 Performance of GPT-3 generated headlines compared to our $SP_{\text{individual-F}}$.</td>
<td>36</td>
</tr>
<tr>
<td>3.7 Two paradigms of applying GPT-3 in personalized headline generation. History Oriented uses GPT-3 to generate headlines for users based on their reading history. Topic Oriented first obtains focused signature phrases using our signature identification and selection modules, and then generates the headline based based on the focused topics using GPT-3.</td>
<td>36</td>
</tr>
<tr>
<td>4.1 Datasets statistics.</td>
<td>46</td>
</tr>
<tr>
<td>4.2 Details of the datasets used to train our neural coherence scoring models.</td>
<td>46</td>
</tr>
</tbody>
</table>
4.3 A scoring rubric provided to human evaluators. ......................... 47

4.4 We compare Teacher bots based on naive DoHA\textsuperscript{[128]} model to
variants fine-tuned using different reward functions, \textit{Avg len} refers
to average utterance length, \textit{WC} refers to \textit{WoW-coh}, \textit{IC} refers to
\textit{InferConv-Coh} .................................................. 47

4.5 Teacher bot’s performance when using different coherence score
models during fine-tuning ........................................... 52

4.6 A few example responses from different Teacher bots. In example
A, +\textit{COH} and \textit{FULL} abstractively generates \textit{Concord is famous for}
... to make the response more coherent to user’s question.
However, the underlined part in +\textit{COH}’s statement misrepresent
the input passage and is inaccurate. In example B, all responses
seem coherent because of the open-ended question. .................. 54

5.1 State-of-the-art summarization models investigated in this work for
generation of patient after-visit summaries ......................... 57

5.2 Semantic types used in this study ........................................ 58

5.3 Model variants +\textit{POS} and +\textit{TYPE} aim to inform the model about the
occurrences of events identified by MetaMap ......................... 62

5.4 Statistics of our dataset .................................................. 63

5.5 Instructions provided to physicians. The scoring scale for summary
evaluation is from 1 (worst) to 3 (best) ................................. 65

5.6 Quantitative evaluation of patient after-visit summaries produced by
state-of-the-art summarization models. LED shows best
performance among all tested abstractive models. It significantly
outperforms all other systems for all metrics ($p < 0.05$), with the
exception of BART in terms of R-1, according to a
non-parametric Wilcoxon signed rank test ................................. 65

5.7 Automatic evaluation of our hallucination detector (\textbf{H-Alert}) and
missing event detector (\textbf{M-Alert}). Both detectors strongly
outperform their baselines ................................................. 69
5.8 A direct comparison of summaries before and after physician revision. A post-study interview with physicians reveals that most revisions are related to missing key medical events (colored orange). They also spend substantial efforts explaining medical jargon to patients and fixing hallucinations (colored red).

6.1 Expert-written question templates are used to generate a question from each binary relation. This method enables us to create targeted questions about salient medical events. By posing questions about one event, we guide patients towards the other as potential answers. The placeholders are to be replaced with medical events detected from discharge instructions.

6.2 Question generation (QG) from a medical event (bottom) or a binary relation of events (top).

6.3 A hierarchy of salient medical events. We consider both medical events (E) and their binary relationships (R).

6.4 Results of fine-tuning four pretrained models on IE task: medical event extraction (Top) and event-relation identification (Bottom).

6.5 Automatic evaluation results of medical event identification per category with ClinicalRoBERTa model.

6.6 Event-Relation detection per category with ClinicalRoBERTa model.

6.7 Prompt presented to GPT-4 for evaluating the quality of generated questions and answer verification. GPT-4 is expected to output a score on each perspective directly.

6.8 Examples of the synthetic discharge instructions and generated questions.

6.9 Examples of chatbot’s answer feedback.

8.1 Abbreviation list.
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>4</td>
</tr>
<tr>
<td>1.2</td>
<td>8</td>
</tr>
<tr>
<td>1.3</td>
<td>10</td>
</tr>
<tr>
<td>1.4</td>
<td>14</td>
</tr>
<tr>
<td>3.1</td>
<td>29</td>
</tr>
</tbody>
</table>

1.1 An example of generating a personalized news headline using our framework (solid green line) as compared to generating general headlines directly from the news article (grey dotted line). Both headlines are appropriate for the news article, but the green headline is more attractive to users interested in the topic *Upper East Side, Manhattan.*

1.2 The teacher bot educates a user about "BERT" through conversation generated from the document.

1.3 An example after-visit summary generated from electronic health records (EHR) notes associated with a patient. A novel alerting mechanism is proposed in this work to report errors found in the summary, including missing medical events and hallucinated facts. We aim to build effective detectors with self-supervision on unlabeled data for error alerting.

1.4 An illustration of PaniniQA, our interactive question-answering system for patient education. It generates questions from discharge instructions, helping patients understand their health conditions through interactive question answering. An answer verification module confirms correct responses or expands feedbacks on partially correct ones. The final turn shows a GPT-generated question. Its answer is absent from the discharge instruction and it is deemed inappropriate for patient education.

3.1 Synthesizing user profiles. The synthesized user’s interests contain randomly selected interest phrases, i.e. *Stanford University, Diabetes, Boeing.* Some news headlines related to these phrases are chosen to represent the synthesized user’s reading history. During the inference stage, one news article containing the interest phrase *Stanford University* is selected as the source article for headline generation.
3.2 Result of human evaluation scores on the generated headlines w.r.t. text quality, headline appropriateness, and user adaptation. 34

4.1 Our two-phase training framework 40

4.2 ROUGE gain and utterance length tend to decrease as the number of turns increases 50

4.3 Coverage and coherence scores when applying different reward scores 51

5.1 One or two event nuggets are randomly masked out from a summary sentence (a). The masked sequence (b) is fed to a denoising auto-encoder to produce a synthesized sentence that may contain hallucinated medical events (c). 60

5.2 “abdominal pain” appears in both the clinical document and after-visit summary, with the same CUI. “nausea vomiting” and “nauseous” are aligned because there is an is-a relation between the two concepts. 61

5.3 Summaries are rated by medical practitioners along the dimensions of adequacy, faithfulness, readability and ease of revision. Their ratings are averaged for each summary. 67

6.1 Word cloud demonstrating the most frequent medical terminologies and their frequency in our annotations. The sizes of the terminologies refer to their frequency in our dataset. These terminologies are identified from annotated medical events using SciSpacy. 79

6.2 System UI of our human evaluation study: the left panel shows a discharge note (Condition None), and the right panel provides the question-answering interactions to the user via a chatbot. The bot can either present only questions (Condition Q in green boxes) or plus answer feedbacks (Condition QA in orange boxes). 82

6.3 Patient evaluation results, including Cloze Test accuracy and evaluator rankings’ MRR scores across four categories (higher is better). The methods are represented with color-coding: None-blue, Q (Human)-orange, QA (GPT)-yellow, QA (GPT+IE)-beige, QA (Human)-green. 87

6.4 GPT-4’s evaluation scores for question quality 89
CHAPTER 1
INTRODUCTION

1.1 Motivation

When presenting information to people from distinctive backgrounds, a one-size-fits-all approach may undermine individuals’ distinct interests, life experiences, educational attainments, and cultural context that shape their capacity to effectively grasp information. [46] Personalized information service has been investigated, with the goal of providing individuals with information that is relevant, meaningful, and beneficial to their unique circumstances. [61, 95, 43] However, existing studies pertaining to personalization predominantly concentrated on aligning existing content with individuals’ based on their preferences. These encompass personalized search engines [44] and recommendation systems [25].

Recent years have witnessed a rapid and remarkable advancement in generative language models [15, 92]. These models enable the creation of new content that is virtually indistinguishable from human-generated content, and support a wide spectrum of applications, from automated summarization to interactive dialogues and question answering. By harnessing the capabilities of generative language models, we have the potential to tailor the presentation of information to better suit individual needs and interests. [169]

This aspect gains particular significance in the realm of healthcare and patient education [75, 177], where individuals seek to comprehend their health status and adhere to self-care instructions. Given that each patient may contend with unique medical conditions and varying levels of health literacy, it becomes imperative that
the information imparted to them is meticulously personalized to facilitate accurate health status understanding and efficient patient self-management.

This thesis delve into the emerging application of generative language models to revolutionize personalized information access. Our primary objectives are to address two key questions: 1. How to analyze user profiles and leverage them for guiding the generation of personalized content; 2. How to proficiently convey information to users through conversational interfaces. To answer the first question, we present **Personalized News Headlines Generator**: This system aims to generate personalized news headlines tailored to individual users. By analyzing users’ browsing history, the system provides news summaries that align precisely with their information needs. This approach facilitates users’ efficient discovery of relevant information and enhances their overall information access experience. To answer the second question, we present **Interactive Dialog Agent for Reading**: We introduce an interactive dialog agent that engages users in conversation to assist reading. Instead of passive consumption, users can interact with the content, ask questions, and seek clarifications. The interactive nature of this agent fosters better comprehension and engagement, enabling users from diverse backgrounds to understand the content more easily. To assess the practicality of the aforementioned systems in real-world scenarios, we integrate them into patient education applications. Specifically, we create the following two systems: 1. **Personalized Discharge Instructions Generator**: This system is designed to assist professions in efficiently and accurately producing discharge instructions personalized for distinctive patients. This personalization empowers patients to better understand their health situation, and effectively manage their post-discharge activities. 2. **Dialogue Agent for Patient Education**: This dialogue system provides crucial support for patient education. Patients are engaged in a conversation with the agent, and required to answer a series of questions closely
related to their discharge instructions. Going through the conversation ensures patient have a clear understanding of their medical conditions and treatment plans.

These proposed systems hold the promise of transforming information access and understanding. By embracing the capabilities of generative language models, we envision a future where information delivery is personalized, interactive, and empowering, ultimately fostering a more inclusive and informed society. In the following sections, we provide a detailed exploration of each of these systems.

1.2 Generating User-Engaging News Headlines

Personalized news recommendation systems, such as Google News and Yahoo News, help users discover articles that align with their interests [71]. However, these systems often present the same article headline to all users, making it difficult for them to understand the connection between their interests and the recommended article, potentially reducing the effectiveness of the recommendation system. To address this, we propose a new framework for generating personalized, engaging headlines that clearly show the connection between a user’s reading history and a recommended article. Our framework has the potential to improve the efficacy of personalized news recommendations, and recommendations for short videos, articles, recipes, etc. [106, 69, 49]

Generating personalized headlines is a challenging task due to the constraints of conciseness and the need to capture the reader’s attention. A personalized headline should (a) effectively convey the main message of the article and (b) provide a clear link to the user’s reading history, using only about 10 words on average [11]. There are two main challenges in this task. First, a headline that entices users to click, but only presents limited information and fails to convey the essential story, becomes clickbait rather than a useful headline [13, 127]. Second, it is difficult to find large scale annotated datasets containing news articles, multiple personalized headlines,
Figure 1.1: An example of generating a personalized news headline using our framework (solid green line) as compared to generating general headlines directly from the news article (grey dotted line). Both headlines are appropriate for the news article, but the green headline is more attractive to users interested in the topic Upper East Side, Manhattan.

and associated user profiles. Such a dataset would be useful in developing personalized headlines, but it is currently unattainable.

The key to effective personalization is to develop a comprehensive framework that enables us to (a) understand users’ interests based on their reading histories, (b) produce personalized headlines, and (c) evaluate the effectiveness of these headlines in terms of user preference. Previous studies on headline generation have primarily focused on producing headlines that accurately summarize a given news article or its first sentence [154, 180, 107, 152, 70], but have not considered the potential benefits of personalization. In this study, we propose a pipeline that incorporates user profiling1 and a comprehensive synthesis of automated and human evaluation methods for user preference to produce personalized headlines that cater to a varied audience.

Our approach focuses on learning a relevance function that condenses a user’s reading history into a collection of signature phrases. This method for user profiling is both efficient and adaptable, as the signature phrases can be easily updated as

---

1We are interested in analyzing users’ reading histories, i.e., the sequence of news headlines they have recently browsed, to gain a deeper understanding of their interests and preferences. We do not have access to users’ demographic data.
the user’s interests evolve [7]. These signature phrases are derived from news article based on the user’s reading history through contrastive learning without the need for annotated data. For example, if the phrase Upper East Side frequently appears in the user’s reading history, it could become a signature phrase for that user (Figure 1.1). These signature phrases do not need to appear verbatim in the user’s reading history and can indicate broader interests, e.g., if the phrases Avengers and Hulk appear in the user’s reading history, it could indicate a love for Marvel movies and Marvel Studios could be a signature phrase that reflects this interest. We build a synthetic dataset that trains the model to generate personalized headlines for a news article. Using signature phrases, our model is able to create a connection between the recommended article and the user’s interests, resulting in personalized headlines that are both engaging and anchored to the article to avoid clickbait.

Evaluating personalized news headlines presents unique challenges [47]. It would be ideal to have human evaluators judge the effectiveness of system headlines. Indeed, we have conducted a human evaluation in this study. However, this process is time-consuming and costly, making it impractical during the system development phase. Thus, we propose a comprehensive synthesis of automated and human evaluation methods to assess headline relevance and user preference. By using signature phrases, we can synthesize user profiles of various types. We hypothesize that personalized headlines generated for these user profiles will be preferred by the same users over generic, non-personalized headlines according to recommender-driven metrics [72, 173]. We also experiment with a variety of automatic metrics to assess headline quality in terms of informativeness, relevance to the source article, and content accuracy [82, 37].

In this paper, we make the following contributions:

- we present a comprehensive framework for generating personalized news headlines that convey the essential message of the article and capture the reader’s attention
while also aligning with their interests. Our framework utilizes a learnable relevance function to derive signature phrases from users’ reading histories and uses them to personalize the headlines;

- we thoroughly synthesize automated and human evaluation methods to assess the effectiveness of headlines in terms of their accuracy and user preference. We further compare our proposed framework with strong headline generation baselines, present results on benchmark news datasets, and identify promising directions for future research through an in-depth analysis of system outputs.

1.3 Learning as Conversation Reading Assistant

Communication is the central process of education [27]. In learning as conversation [148], a student does not read a passage but gains information and knowledge through conversation with a teacher who reads the passage. Compared to the traditional learning by reading, learning as conversation has the advantages that conversation helps students stay engaged and that information is provided piece by piece, which helps strengthen learning with a shorter attention span.

The advantages of learning as conversation have been verified with educational evidence [113, 91, 48]. For example, studies have shown that when children read storybooks, parents’ guided conversation, e.g., posing questions and providing responsive feedback, substantially amplifies the learning benefits. While high-quality conversations with experts are not always available, it would be helpful if AI-empowered chat bots could be applied to facilitate users to gain information or knowledge.

In recent years, there has been significant research in content-grounded dialogue generation, where external passages are used to inspire knowledge intensive dialogues. However, these systems or datasets are either for chit chat [208, 29] or for goal-oriented information seeking [40, 21], little work has explored applying chat bots for the learning as conversation purpose.
In this work we propose a novel task for learning as conversation: information-acquisition-oriented dialogue generation. Given a passage, our chat bot actively engages with an end-user to form a coherent conversation, so that the user could gain knowledge without reading the passage. Our task has a broad range of potential application venues in which people traditionally rely on reading to obtain information, including:

- Education: Chat bot helps a user gain knowledge from books or research papers;
- News and Media: Through conversation, a user could be provided stories tailored for his/her preference;
- Tutorial: While reading an instruction book could be tedious and time-consuming, a chat bot could efficiently walk a user through the process.

As shown in Figure 1.2, for our task, a good conversation should have the following characteristics:

1. Coverage: The chat bot should try to convey as much information in the passage as possible, instead of mumbling about irrelevant information;
2. Coherence: The chat bot’s response should be coherent to the user’s utterance, making the user feel that his/her questions are followed and addressed.
In summary, we propose a novel framework which consists of the following two chat bot modules: 1) Teacher bot, which attempts to transfer the information in an input passage to a user through conversation; and 2) Student bot, which responds to Teacher bot to form coherent conversations during training. The two bots are trained in a two-phase manner: In Phase 1, we pre-tune the two chat bots on Wizard of Wikipedia [29] dataset, enabling both bots with the basic ability of conversing over a passage. In Phase 2, we fine-tune Teacher bot through self-play with Student bot, guided by reinforcement rewards. In this process, we enhance Teacher bot to be more informative while maintaining the ability to coherently address human users. Specifically, the fine-tuning phase is unsupervised, i.e. Teacher bot could be transferred to various domains or corpora without additional annotated dialogue datasets.
Our contributions include: 1) A novel task of information-acquisition-oriented dialogue system; 2) A novel unsupervised learning framework which enables a teacher bot to carry out informative and coherent conversations with human users for information acquisition purpose; 3) Extensive experiments with human evaluation demonstrate the effectiveness of our proposed approach.

1.4 Assisting Physicians in Writing Patient After-Visit Summaries

Studies have shown that the majority of patients do not understand their clinical visits [117]. After-visit summary note (AVS) is a summary given to patients after their clinical visit, it is intended to summarize patients’ clinical visits and help their disease self-management [39]. Compared to clinical notes, an after-visit summary has the following characteristics: 1) it is written in lay person language thus is easy for patients to read and comprehend; 2) it only contains information that patients should be aware of, leaving out redundant details, e.g. unimportant lab results. Studies have shown that around 36% of American adults have limited health literacy [83], and 94.4% of patients found that lay language after-visit summary helps them understand their clinical visits [123]. However, the implementation of after-visit summary is challenging. Many physicians face excessive workloads [170] and do not have time to complete the summaries in a timely manner [58]. Thus, there is a real need for—and this study contributes to—automatic generation of after-visit summaries to unburdening physicians with complex information workflows.

We explore best-performing neural abstractive summarizers to generate after-visit summaries from electronic health records (EHR) notes. The summaries are rated by physicians as concise and easy to read. However, they can not be presented directly to patients, as they frequently contain two types of errors: 1) **Missing content.** A summary often leaves out important details such as medication dosage and route, un-
Figure 1.3: An example after-visit summary generated from electronic health records (EHR) notes associated with a patient. A novel alerting mechanism is proposed in this work to report errors found in the summary, including missing medical events and hallucinated facts. We aim to build effective detectors with self-supervision on unlabeled data for error alerting.

1) Type I Error (Hallucination). Summaries contain hallucinated content or content not supported by the input documents. For example, an abstractive summary on kidney infection was generated from an input document that describes urine infection. These types of errors are not uncommon in abstractive summarization [86, 108, 120], but they could be disastrous to patients.

In this study, we build systems to facilitate detection and correction of those types of errors, allowing physicians to correct or edit system generated summaries. As illustrated in Figure 1.3, Summarization produces a system summary; Error Alerting automatically detects errors from the generated after-visit summary. Crucially, we build effective detectors with self-supervision on unlabeled data for error alerting. A novel dataset is constructed by synthesizing summaries containing medical events that are inconsistent with their source documents. Using this simulated dataset, we train a hallucination detection model, which alerts physicians of potential hallucination content. Further, by aligning medical events in EHR notes to those in after-visit summaries using MetaMap [6], we identify key events important to patients, and alert physicians of salient medical events not covered in the generated summaries as missing content.

The contributions of our research are as follows:
• We propose a new task that generates lay language AVS from EHR notes, build and evaluate state-of-the-art NLP models for this task. A novel alerting mechanism is proposed to report errors, including missing medical events and hallucinations. The training of our error detectors is self-supervised, using only unlabelled text.

• Clinical applications demand high performance. Existing automatic metrics are not adequate for evaluating the quality of generated AVS. Therefore, we conduct a qualitative assessment of system outputs with medical practitioners. Our findings show that the alerting mechanism could provide a promising avenue towards making the writing process easier for physicians.

1.5 Enhancing Patient Education Through Interactive Question Answering

Limited patient understanding of their medical conditions can lead to poor self-care at home. Upon hospital discharge, physicians often provide discharge instructions to aid in patients’ recovery and disease self-management [39]. However, some patients may have difficulty understanding and memorizing instructions due to low health literacy, limited memory, or an absence of supervision. For example, research shows that patients only retain a minimal amount of information from discharge instructions, with an immediate forgetting rate of up to 80% [73, 140]. Further, when instructions are misinterpreted by patients, there is often a lack of corrective intervention. Limitations in a patient’s understanding of their medical conditions hinder their prospects of recovery. It is imperative to investigate new methods of patient education to enhance health outcomes.

In this study, we explore a novel method inspired by Dialogic Reading [171] to educate patients through interactive question-answering. Dialogic Reading actively involves patients in the learning process by following the PEER sequence: Prompt, Evaluate, Expand, and Repeat, which enables patients to engage in a meaningful
dialogue, further strengthening their understanding and retention of the material. As illustrated in Figure 1.4, our dialog agent asks questions about key aspects of discharge instructions and encourages patients to read and understand the instructions to provide accurate answers thoroughly.

Crafting questions that effectively meet educational objectives is challenging [14, 32]. A suitable question should be based on the patient’s discharge instruction and aim to improve their understanding of health conditions, such as “What was the probable cause of your chest pain?” Conversely, the question “How does cardiac catheterization help treat a heart attack?” illustrated in Figure 1.4, may exceed the education scope, as it is unanswerable or requires knowledge beyond the provided discharge instruction. Such questions are considered unsuitable for patient education.

We introduce new question-generation methods that draw on the advancements of LLMs [15, 119, 118]. Utilizing OpenAI’s GPT-3.5 model, we generate informative questions from discharge instructions. Further, we combine LLMs with medical event and relation extraction to constrain the model, producing questions that target salient medical events identified in the discharge instructions. We create a new dataset with expert-annotated medical events and relations for discharge instructions from the MIMIC-III [65] database. While earlier efforts have annotated events that physicians would discuss during patient handoff [121, 89], our focus is on identifying pairs of medical events with correlational or causal relationships. By posing questions about one event, we guide patients toward the other as potential answers.

Our system further incorporates an answer verification module to provide instant patient feedback. When patients give correct answers, the bot confirms them, reinforcing their understanding. If answers are incorrect or partially correct, the bot clarifies misunderstandings and provides additional information. Extensive automatic and human evaluations demonstrate the efficacy of our question-generation methods
and show that PaniniQA holds great promise for promoting patient education. To summarize, our research contributions are as follows.

◊ We explore a new way of educating patients regarding their health conditions through interactive question-answering. Our approach aligns with the P.E.E.R. dialogic reading theory that promotes patients’ active participation in comprehending medical events.

◊ We compare questions generated using OpenAI’s GPT-3.5 model, our enhanced method with medical event extraction, and human-written questions tailored for patient education. We meticulously evaluated all questions, answers, and patients’ educational outcomes.

◊ Through comprehensive human evaluations, we demonstrate that PaniniQA holds promise for patient education. Future work includes controlling the difficulty of questions, prioritizing questions given patients’ health literacy, and enabling interactive learning of medical concepts.
“You were admitted to the hospital with chest pain and were found to have a heart attack. You underwent cardiac catheterization and two stents were placed in your left anterior descending artery…”

What is causing your symptom of chest pain?

I had a heart attack.

You are right about that!

How was your heart attack treated?

By a stent procedure.

Your answer is partially correct. You underwent cardiac catheterization and two stents were placed in your left anterior descending artery.

How does cardiac catheterization help treating heart attack?

Question is beyond the education scope

Figure 1.4: An illustration of PaniniQA, our interactive question-answering system for patient education. It generates questions from discharge instructions, helping patients understand their health conditions through interactive question answering. An answer verification module confirms correct responses or expands feedbacks on partially correct ones. The final turn shows a GPT-generated question. Its answer is absent from the discharge instruction and it is deemed inappropriate for patient education.
CHAPTER 2
LITERATURE REVIEW

In recent years, language models have demonstrated remarkable effectiveness in a wide array of text generation tasks, such as automatic summarization, dialog systems, and question answering. Leveraging the success of these text generation tasks, we can apply similar approaches to create personalized content that supports users in better understanding information. For instance, by analyzing users’ profile, we may better understand their interests and utilize automatic summarization techniques to condense complex content into simpler text snippets that cater to users needs. This enables users to grasp essential information quickly and effortlessly. Similarly, when dialogue agents have conversations with the users, by analyzing users’ dialogue history, the agents may obtain the users’ underlying intention and focus, thus may lead the conversation to address users’ concerns promptly and contextually, enhancing their overall information access experience. In the following chapter, we delve into prior research endeavors that are closely aligned with the focus of our work. Our discussion begins by elucidating the realm of language models and their manifold utility, followed by introduction previous researches concerning personalized information service. Subsequently, we focus on supporting technologies we applied for the thesis, including automatic summarization, dialogue generation and other technologies for patient education.
2.1 Language Modeling

Pre-trained Language Models Language models are a type of artificial intelligence model designed to understand and generate human language text. These models have gained significant attention and popularity in recent years due to their remarkable ability to model the probability of language and to process and generate natural language in a way that closely resembles human communication. Modern language models predominantly rely on the transformer architecture, initially introduced by Vaswani et al. [164]. These models can be tailored for specific purposes through variations in training data selection and the creation of synthesized training datasets.

Language models come in diverse structural forms. Some exclusively function as encoders, as seen in models like BERT [26], RoBERTa [103] and Transformer-XL [196]. These models transform a sequence of text into context vectors, which are usually utilized to support downstream tasks like token-level classification and sequence-level classification. Another category of language models, such as ELMo [125], GPT-3 [15], and GPT-4, operate solely as decoders. These models generate output text direct following the input text prompt. Additionally, there are models employing both encoder and decoder components, as observed in BART [92], Pegasus [196], and T5 [132]. These models first apply encoder to transform the input text into context vectors, and then apply decoder to generate output text based on the context vectors. The focus of the research in this thesis predominantly revolves around decoder-only or encoder-decoder structured language models, which are deemed more suitable for tasks involving text generation.

The performance of language models is profoundly influenced by the selection of pre-training corpora and task-specific objectives. Notably, language models pre-trained on biomedical literature, such as BioBERT [88], or clinical records, exemplified by ClinicalBERT [94], exhibit enhanced efficacy in bio-medical tasks compared to
models pre-trained on general-domain text. In our exploration, we also demonstrated that language models pre-trained on clinical documents consistently demonstrate better performance in patient education tasks. On the other hand, language models like Pegasus [196] and BART-CNN/Daily mail [92] that undergo pre-training for abstractive summarization also showcase superior performance in summarization tasks. Their advantages over ordinary pre-trained language models are also reflected in our experiments in news headline generation and medical document summarization.

**Large Language Models** Recent research reveals the substantial benefits of scaling up pre-trained language models [167]. Although scaling is mainly conducted in model size (with similar architectures and pre-training tasks), these large-sized language models display different behaviors from smaller language models, leading to significant improved performance across a spectrum of downstream tasks. For example, GPT-3 is capable of the ability of in-context learning and zero-shot learning, whereas GPT-2 can not do well. [203]

Particularly remarkable are the achievements of large language models fine-tuned through reinforcement learning with human guidance, as demonstrated by Wang et al. [165]. These models showcase exceptional proficiency in elevating the quality of generated text while mitigating issues like hallucination and the use of toxic language.

Significantly, the prowess of large language models extends to supporting various text generation tasks, including tasks such as automatic summarization [199, 162], dialogue systems [22, 100], and question answering [141, 55]. Notably, these tasks serve as foundational components for personalized information comprehension.

### 2.2 Applying LMs for Information Personalization

**Personalized Information Service** Personalized information service refers to service that tailors information and content to the individual preferences, needs, and interests of users. It leverages data analysis and user input to deliver relevant news,
recommendations, updates, or other information, ensuring a more customized and engaging user experience. This service can be found in various contexts, such as news apps, social media algorithms, recommendation engines, and more, all aimed at enhancing user satisfaction and engagement by delivering content that aligns with their unique preferences and behaviors. Current researches concerning personalized information service mostly focus on the following perspectives:

1. **User Profiling** is applied to create detailed and comprehensive profile of individual users based on various data points and information. This profile is used in various applications and services, to understand and predict user behavior, preferences, and characteristics. Previous researches have explored creating user profiles based on user behavior [34], search history [157], demographic information [80], and social information [98].

2. **Content Analysis** in the context of a personalized information service is a crucial process that involves examining and understanding the content or information available to the service. This analysis is essential for delivering tailored and relevant content recommendations to individual users. Content analysis include researches topics such as text embedding [72, 193], keyword extraction [42, 9], content categorization [3, 20].

3. **Recommendation Algorithms** are applied to match user profiles with relevant content. Traditional recommendation algorithms include collaborative filtering [159], content-based filtering [8], and hybrid approaches [90]. These algorithms identify patterns and similarities among users and content items to make personalized recommendations. In recent years, pretrained language models show remarkable performance in recommendation systems due to their accurate representation of textual content [176, 41, 175].
**Privacy Protection**  Privacy is a significant and complex issue when it comes to personalized information services. These services, which often rely on collecting and analyzing user data to provide tailored experiences, must strike a balance between customization and respecting users’ privacy rights. Researches concerning protecting user privacy mostly focus on the following perspectives:

1. **Data Security** is a paramount concern for personalized information services, which often deal with sensitive user data to deliver customized experiences. Ensuring the security of this data is essential to protect user privacy, maintain trust, and comply with data protection regulations. This includes topics such as data collection [64], data storage [183] and data sharing [206].

2. **Privacy-preserving Techniques** are used to safeguard sensitive information while still allowing for useful data analysis or processing. These algorithms are particularly important in contexts where data privacy and security are critical, such as in healthcare, finance, and personal information management. These approaches includes federated learning [97, 96], homomorphic encryption [172, 115], data masking [62, 187].

### 2.3 Applying LMs for Automatic Summarization

The goal of automatic summarization is to condense one or multiple passages into one short paragraph of text, which is expected to cover the major information in the input passages. This thesis focus on two aspects in two subtasks in automatic summarization: 1. Summarizing news articles to generate news headlines; 2. Summarizing clinical documents into after-visit summaries for patients.

**Generating News Headline**  Automatic headline generation aims at generating headlines that can capture the gist of the input articles. This task has made significant progress in recent years [107, 59, 85, 153, 50], thanks in part to the development of
large language models [92, 132, 195, 15, 23] and the availability of benchmark news datasets such as Gigaword, XSum, and Newsroom [143, 116, 53]. These datasets include a single headline for each news article, serving as the ground truth for the models. In contrast to previous works, we aim to personalize headline generation to improve content recommendations, where a personalized headline should convey the main points of the article and capture the user’s attention.

Specifically, evaluating personalized content is a largely under-explored area, partly due to the lack of ground truth for personalized content generation [47]. Without ground truth, it is challenging to apply commonly used text generation evaluation metrics such as ROUGE, BLEU, BERTScore, MoverScore, BLEURT. [99, 126, 198, 204, 147]. To leverage recent advances in data synthesis [122, 4, 105], we propose synthesizing user profiles of various types. We then evaluate system headlines against these profiles along multiple dimensions, including their alignment with user interests, relevance to the source article, and content accuracy.

**Clinical Document Summarization** Automatic summarization is the task of condensing a long passage into a concise paragraph. Compared to summarizing documents in general domain, summarizing clinical documents has higher requirements for accuracy and readability. In previous researches, Zhang et al. [201] propose to generate the impression section of a radiology report using seq2seq models. Miura et al. [111] perform image-to-text radiology report generation by optimizing entity-based rewards with reinforcement learning. Studies are performed for summarizing doctor-patient dialogues [67, 79] and evaluating system generated notes [114]. It is important for an after-visit summary generated from EHR notes to avoid type I and type II errors. A type I error (false positive) suggests that there is false or inaccurate information in the summary, due to hallucinations, incorrect grounding, etc. It is a challenging and lingering problem facing natural language generation [38, 87, 82, 107, 120, 163]. A more surprising observation is that the type II error
(false negative) is deemed particularly harmful to patients. When salient medical events such as diagnoses or treatments are left out of the after-visit summaries, it could have a detrimental effect on patients’ self-care after being discharged from hospitals [133, 156]. This empirically motivates our work, where we seek to effectively identify salient medical events in EHR notes and alert physicians of any missing events to help them avoid those errors. A distinguishing characteristic of after-visit summaries is that they are patient-oriented. The summaries provide relevant and actionable information to patients, such as reasons for visit, diagnoses and procedures. Differing from physician-oriented clinical notes, these summaries are written in an easy-to-understand language, and they remain understudied in NLP.

2.4 Applying LMs for Content Grounded Dialogue System

Content grounded dialogue generation is the task of using the information provided in external content (e.g. a passage) to guide dialogue generation. Compared to previous research, the task explored in this thesis has the following novelties. 1) Compared to content-grounded information-retrieval-oriented dialogue such as doc2dial [40] and ABCD [21] where the chat bot responds to user query in a passive way, we expect our chat bots to convey knowledge proactively. 2) Compared to chit chat-oriented dialogue such as [208, 29, 77, 179], our task is more focused on extensive conversation in a particular topic, and aims at helping the end user acquire knowledge or information from a given passage. 3) Contrasted to chat bots that are applied in a single domain [208, 112, 178], our chat bot could be transferred to other domains through self-talk based fine-tuning. Another line of research works focus on content grounded text generation models [128, 205]. Compared with ordinary text generation models (e.g. BART [92]), these models are specifically designed to model external content as an additional input, and achieve better performance on content grounded dialogue generation tasks including CMU DoG [205] and Wizard of Wikipedia [29]. There
have also been research works applying dialogue systems for educational purposes. Some chat bots are for language practice. Others are specially designed for education in a single domain or task, e.g. moral education, educational debate. Compared with previous educational dialogue systems, our system is for information acquisition without domain restriction. Our task is also related to conversational question answering (CQA), e.g. However, most existing CQA systems passively respond to user queries in single turn conversations, while our system actively engage with users in multi-turn conversations.

2.5 Applying LMs for Patient Education

There is a growing need to improve patients’ understanding regarding their hospital experiences. Lack of understanding can result in non-adherence to discharge instructions and readmission to the hospital due to poor self-care at home. Previous research has attempted to generate hospital course summaries for patients using lay language. This paper goes a step further by utilizing interactive question answering to communicate essential medical events from discharge instructions to patients, thus enhancing their understanding and retention of the material.

Our proposed method differs from existing clinical question-answering studies in several aspects. Most clinical QAs are designed to satisfy individuals’ information needs, with questions modeled after those that can be asked by physicians. These systems focus on improving the accuracy of their answers. In contrast, our goal is to educate patients and prompt them with questions that will enhance patients’ understanding of their doctors’ visits. A successful QA system should be comprehensive and exhaustive, asking all relevant questions and prioritizing them based on the patient’s medical history and health literacy.
Successful patient education requires effective questioning [129]. Particularly, question generation has been studied using template-based [57, 19, 36] and neural seq2seq models [30, 31, 74, 160, 149]. Instruction-tuned LLMs have demonstrated exceptional abilities in conversing with humans [15, 144, 119, 23, 104]. However, most research has been conducted using CommonCrawl, Wikipedia, and other generic texts. Considering the factuality issues of neural language models [108, 120], question generation in the medical domain remains challenging.

Learning through conversation can improve education outcomes [48, 200, 17, 184, 184, 182]. Dialogic Reading [171, 113, 91] has demonstrated that engaging children in a guided conversation with parents while reading storybooks can significantly enhance their learning outcomes. While engaging physicians in high-quality conversations may not always be feasible, the use of question answering facilitated by a chatbot could be a valuable means of helping patients acquire a deeper understanding of their health conditions.
CHAPTER 3
GENERATING PERSONALIZED NEWS HEADLINES

In this chapter we present a novel framework that addresses these challenges by incorporating user profiling to generate personalized headlines, and a combination of automated and human evaluation methods to determine user preference for personalized headlines. In Section 3.1, we introduce our framework which utilizes a learnable relevance function to assign personalized signature phrases to users based on their reading histories, which are then used to personalize headline generation. In Section 3.2 we introduce creating synthesizing users to enable training and evaluating the models’ performance. In Section 3.3, we demonstrate the effectiveness of our proposed framework in generating personalized headlines that meet the needs of a diverse audience through extensive evaluation. Our framework has the potential to improve the efficacy of news recommendations and facilitate creation of personalized content.

3.1 Approach

Our goal is to generate a user-engaging headline that conveys the main idea of a given news article \( d \) for a specific user \( u \). To achieve this, we have developed a three-step framework: (1) *Signature phrases identification*. Using a key-phrase generation module, we identify a set of candidate signature phrases \( Z_d = \{z_1, z_2, \ldots \} \) that cover various aspects of \( d \) (Section 3.1.1); (2) *User signature phrases selection*. From the set of candidate signature phrases, we select a subset \( Z^u_d \subseteq Z_d \) that relates to user \( u \)'s interests as the user signature phrases (Section 3.1.2); (3) *Signature-oriented headline
Based on the news article $d$ and the selected user signature phrases $Z_d^u$, we generate a headline that introduces the content of the article $d$ from the perspective of the user $u$'s personalized interests (Section 3.1.3).

### 3.1.1 Signature Phrases Identification

We approach this task as a conditional text generation problem, in which the model takes a news article or headline as input and outputs all candidate signature phrases in the input sequence, separated by semicolons. We use a BART model that has been pretrained on the KPTimes dataset\(^1\). KPTimes [45] is a large-scale dataset containing 279K news articles paired with editor-curated signature phrases. Unlike other datasets for signature phrase identification [109, 78] that focus on scientific research papers, KPTimes focuses on extracting signature phrases in news articles, making it well-suited for our task. The model is trained by minimizing the cross-entropy loss between the predicted signature phrase sequences and the human-curated signature phrase sequences.

### 3.1.2 User Signature Selection

In this step, we rank all candidate signature phrases in $Z_d$ based on their level of engagement with user $u$'s reading history $H_u$, and select the top $k$ candidate signature phrases as the user signature phrases. Suppose that the user’s history $H_u$ can be defined as a set of headlines of articles that the user has previously read, i.e., $H_u = \{t_1, t_2, \ldots\}$. We first convert each signature phrase $z_i \in Z_d$ into a dense vector $z_i$ using a signature phrase encoder. To calculate the user-engaging scores for each candidate signature phrase $z_i$, we consider two different encoding strategies for the user’s history:

\(^1\)https://huggingface.co/ankur310794/bart-base-keyphrase-generation-kpTimes
(1) **Holistic history encoding.** We concatenate all headlines in the user’s reading history $H_u$ with additional semicolons for headline separation. Then we encode the concatenated headlines into a dense vector $h_u$ using a holistic history encoder. The engaging score $S(z_i, H_u)$ of a signature phrase $z_i \in Z_d$ for user $u$ is obtained by the dot product of the two vectors:

$$S(z_i, H_u) = z_i^\top h_u. \quad (3.1)$$

(2) **Individual history encoding.** Each individual headline $t_j \in H_u$ is encoded as a dense vector $t_j$ using an individual headline encoder. The user-engaging score is then defined as the maximum dot-product relevance between the signature phrase $z_i$ and each individual headline in the reading history:

$$S(z_i, H_u) = \max_{t_j \in H_u} z_i^\top t_j. \quad (3.2)$$

In practice, we train the user signature phrase selection model using an in-batch contrastive learning approach [131]. We consider a batch of synthesized users $\{u_1, u_2, \cdots, u_{N_B}\}$ where $N_B$ is the batch size, and each user $u_i$ has exactly one user signature phrase $z_i$. The reading history $H_i$ for user $u_i$ is then constructed by randomly sampling news articles whose candidate signature phrases contain $z_i$, i.e., $H_i = \{d \mid z_i \in Z_d\}$. In this way, $(z_i, H_i)$ is considered as a positive pair, and $(z_i, H_j) \ (i \neq j)$ is considered as a negative pair. The contrastive loss for this batch is defined as follows:

$$L_{select} = \frac{1}{2} \left( \sum_{i=1}^{N_B} \log \frac{S(z_i, H_i)}{\sum_{j=1}^{N_B} S(z_i, H_j)} + \sum_{j=1}^{N_B} \log \frac{S(z_j, H_j)}{\sum_{i=1}^{N_B} S(z_i, H_j)} \right). \quad (3.3)$$
3.1.3 Signature-Oriented Headline Generation

We model the user-specific headline generation process as a conditional generation task. Given a news article $d$ and a user $u$, along with the user signature phrases $Z_u^d \subseteq Z_d$, our goal is to generate a headline $t = [w_1, w_2, \ldots]$ for $d$, where $w_i$ is the $i$-th token in $t$. The loss for this generation step is calculated as the negative log-likelihood of the conditional language generation:

$$L_{gen} = - \sum_i \log Pr(w_i \mid w_1, \cdots, w_{i-1}; Z_u^d, d)$$ (3.5)

Specifically, the input to the generator is the concatenation of the user signature phrases $Z_u^d$ and news article $d$, and the output is the signature-based headline $t$. During the training stage, $Z_u^d$ is identified from $t$, the ground-truth headline of $d$. During the inference stage, $Z_u^d$ is identified from $d$ itself and selected by user signature selection models, since the headline $t$ is not available before generation. We use BART here as the generator for headline generation.

3.2 Corpora Processing

In this section, we describe the corpora processing step, including the creation of synthesized users and the generation of signature phrase based headlines. Our data is sourced from two existing news corpora: Newsroom [53] and Gigaword [143, 51]. The Newsroom corpus contains 995,041 article-headline pairs in its training set, 108,837 in its validation set, and 108,862 in its test set. The Gigaword corpus contains 7,704,419 instances in its training set, 394,390 in its validation set, and 381,045 in its test set. For each corpus, we construct two datasets: a synthesized user dataset and a headline generation dataset. The first dataset is used for training the use signature phrase selection model (Section 3.1.2) and evaluating the entire system, while the
Table 3.1: Statistics of the datasets. For each corpus, the synthesized user dataset is used for training the signature phrase selection module and evaluating the entire system, while the headline generation dataset is used for training the headline generation module (it does not have a test set because the generation step is evaluated in the entire system using the test set of synthesized user dataset).

Synthesized User Creation. As real user data is not available, we generate synthesized users to mimic real users’ reading histories. The process for creating synthesized users is illustrated in Figure 3.1 and consists of the following steps: (1) Identification of signature phrases in all news articles of a corpus to build a candidate phrase pool; (2) Mapping of each signature phrase to a series of news articles that contain that phrase; (3) Random sampling of a subset of phrases from the candidate phrase pool as each synthesized user’s area of interest; (4) Random sampling of a set of news articles that contain each user’s chosen interest phrase using the phrase-article map established in step 2.

During the training stage of the signature phrase selector, each synthesized user is assigned only one interest phrase to enable contrastive learning (Eq. 3.4). However,
Figure 3.1: Synthesizing user profiles. The synthesized user’s interests contain randomly selected interest phrases, i.e. Stanford University, Diabetes, Boeing. Some news headlines related to these phrases are chosen to represent the synthesized user’s reading history. During the inference stage, one news article containing the interest phrase Stanford University is selected as the source article for headline generation.

when evaluating the model, each synthesized user is assigned 1 ~ 5 interest phrases to mimic real-world scenarios. It is important to note that it is easier to generate personalized headlines for users with simpler backgrounds (e.g. users whose reading histories only relate to one or two topics). To study the effect of the number of users’ interested phrases on the generated headlines, we create 2,000 synthesized users with 1 ~ 5 number of interested phrases respectively.

In general, headline personalizing is only effective when the source article content aligns with the user’s interests. To ensure relevancy, we randomly select one of the user signature phrases from each synthesized user, and then randomly choose one news article that contains the selected phrase as the input for the test case. This ensures that the news article whose headline needs to be generated is relevant to the user. The evaluation details are further explained in Section 3.3.

**Headline Generation.** In order to generate signature phrase oriented headlines, we use the signature phrases identification model to extract signature phrases from the original headlines. These generated phrases, along with the corresponding news
article contents, are then fed into the headline generation model to generate the original headlines. In our experiments, we truncate all news articles to a maximum of 512 tokens and only keep signature phrases that appear in more than 10 news articles. On average, around 10 candidate signature phrases are identified in each news article, providing a diverse range of perspectives for headline generation.

3.3 Experiments

We thoroughly evaluate our proposed system from different perspectives, including objective evaluation (3.3.2), subjective evaluation (3.3.3) and ablation studies (3.3.4), for personalized headline generation.

3.3.1 Baseline Methods

We compare the performance of our system with the following baseline approaches: (1) PENS-EBNR and (2) PENS-NRMS [5] are LSTM-based personalized headline generation models. Both were trained on the PENS dataset, but using different reading history encoding models; (3) Vanilla System is a BART-large model fine-tuned directly on headline generation datasets without using signature phrases; (4) Vanilla Human refers to original headline given by the author of the news article; (5) SP-headline uses signature phrases identified in the original human-written headline to guide headline generation; (6) SP-random randomly selects signature phrases in the news article to guide headline generation. (7) SP-holistic and (8) SP-individual were introduced in previous sections.

3.3.2 Objective Evaluation

We use various metrics to evaluate the entire personalized headline generation pipeline:

1) Relevance Metrics. We use pre-trained DPR [72] and Sentence-BERT [138] models to calculate the relevance score between texts. Specifically, we report dot-product
### Table 3.2: Objective evaluation results of all methods. “-F” means using the fine-tuned signature phrase encoder, headline encoder and user history encoder, while “-N” means using the naive DPR models as encoders. “REC Score” refers to recommendation score. Vanilla approaches do not consider human preference.

Similarity when using DPR, and cosine similarity when using Sentence-BERT. These relevance metrics are calculated for both the **headline-user relevance** and the **headline-article relevance**. For **headline-user relevance**, the score is calculated between the generated headline and the user signatures. For **headline-article relevance**, the score is calculated between the generated headline and the entire news article.

(2) **Recommendation Score.** Following [173], we train a news recommendation system using the MIND dataset [174]. The system takes in a user’s reading history and a headline of a news article, and outputs a score indicating the degree to which the system would recommend the news to the user.

(3) **Factual Consistency.** We apply the pre-trained FactCC model [82] to obtain the factual consistency score between the generated headline and the news article. We report the percentage of generated headlines that are predicted to be factually consistent with the news article by the FactCC model.

(4) **Surface Overlap.** We use ROUGE-L F1 and Extractive Coverage to evaluate the surface overlap between the generated headline and the reference headline/news article. ROUGE [99] scores are widely used to evaluate the surface level coverage of
generated summaries against golden standards. Specifically, ROUGE-L F1 measures the longest common sub-sequence between the generated output and reference. Extractive Coverage [53] is the percentage of words in the generated headline that are from the source news article, measuring the extent to which the summary is derived from the text.

Table 3.2 presents objective evaluation results for generated headlines. We elaborate our observations from the following perspectives:

**User Adaptation.** (1) The methods *SP holistic* and *SP individual* generally show better performance, indicating that our signature phrase based headline generation framework is able to generate more user-oriented headlines. In contrast, while *Vanilla System* and *SP Headline* achieve higher Rouge-L scores, they have lower scores in user adaptation, suggesting that they have higher similarity with the original headline but do not achieve personalization. (2) Comparing SP based methods, we observe that using selectors fine-tuned on our signature selection datasets (i.e. -F) leads to more user-preferred headlines than their naive counterparts (i.e. -N). This reflects the improvement of fine-tuning signature phrase selector. It is worth noting that the performance of *SP Random* is significantly lower than *SP holistic/individual*, and almost similar to *Vanilla System*, which suggests that user adaptation is only achieved when signature phrases of users’ interests are well-selected. (3) *SP individual* shows better performance than *SP holistic*, indicating that individual encoding better aligns users’ reading history with their interests.

**Article Loyalty.** (1) While *Vanilla System* generally achieves better performance in headline-article relevance, *SP individual-F* generates more headlines that are identified as factually consistent by FactCC. Our analysis found that headlines generated by our SP-based methods are usually anchored to news articles by the signature phrase, i.e. the generated headlines may contain content in the context of the signature phrase (as shown in the example in Figure 3.1). This keeps the generated headlines
related and factually consistent with the news article, thus avoiding click-bait headlines. (2) The extractive converge of the original human headlines is lower than all machine-generated headlines, which implies that human written headlines are more abstractive. This explains the original headlines’ low performance in article loyalty metrics. Note that ROUGE scores do measure our goal of headline personalization, we present the results only to show the generated headlines’ surface-level resemblance to the human written ones.

3.3.3 Subjective Evaluation

We conduct a two-step human evaluation using 16 evaluators who have high English proficiency. In the first step, we collected 2,260 news headlines from 113 common topics in Newsroom and Gigaword corpus. We presented the volunteers with the article headlines and corresponding topics and asked them to select around 20 headlines of their interests mimicking their interest phrases and reading histories. In the second step, we generated headlines for 12 randomly selected news articles containing the volunteers’ interested phrases (6 from Newsroom and 6 from Gigaword). We then asked the volunteers to evaluate the generated headlines through the following five approaches: (1) Vanilla Human; (2) Vanilla System; (3) SP-random; (4) SP-individual-N; (5) SP-individual-F. We evaluated the headlines from three perspectives: (1) User adaptation; (2) Headline appropriateness and (3) Text quality. The grading scale ranges from 1 (worst) to 3 (best).

According to Figure 3.2, our signature-oriented headline generation approaches, SP-Individual-F and SP-Individual-N, perform better than other baseline methods in terms of user adaptation. This is in line with the objective results that our signature-oriented framework generates headlines that cater more to users’ interests.

Further, the headlines generated by Vanilla System obtain the highest scores in headline appropriateness. However, after analyzing the generated headlines, we re-
Figure 3.2: Result of human evaluation scores on the generated headlines w.r.t. text quality, headline appropriateness, and user adaptation.

<table>
<thead>
<tr>
<th>User Signatures</th>
<th>Mark Zuckerberg; Bill Gates</th>
</tr>
</thead>
<tbody>
<tr>
<td>News Article</td>
<td>The Giving Pledge, invented by Bill and Melinda Gates and Warren Buffett to spur the philanthropy of billionaires, ... assuredly the coolest recruits are Facebook co-founders Mark Zuckerberg and Dustin Moskovitz, who each turned 27 in May ...</td>
</tr>
<tr>
<td>Generated Headline</td>
<td>The Giving Pledge: Zuckerberg and Gates at 27</td>
</tr>
</tbody>
</table>

1 User Signatures: The Force Awakens
User Interest Phrase: Star Wars
News Article: Star Wars: Episode 7 has revealed its full title - it will be called Star Wars: The Force Awakens
Generated Headline: Star Wars Episode 7 to be called Star Wars: The Force Awakens

2 User Signatures: Shanghai Composite Index
News Article: China stocks fell more than 1 percent on Tuesday morning ... the Shanghai Composite Index lost 1.4 percent ...
Generated Headline: Shanghai Composite Index falls 1.4% despite market-soothing measures

3 User Signatures: Photography
News Article: ... Self-publishing is not a new development in photography, but recently the trend to make, edit, design and produce ...
Generated Headline: Self-publish or be damned: why photographers are going it alone

4 Human Headline: Self-published photography books to be showcased at Photographers’ Gallery
Generated Headline: Self-published photography books to be showcased at Photographers’ Gallery

Table 3.3: Examples of generated headlines.

alized that some identified signature phrases did not correlate well with the article’s main point, thus diverging from the article. For example, in the third example in Table 3.3, the generated headline focuses on *Shanghai Index’s drop*, which is only a minor evidence to support the article’s main point, i.e. *China’s stock market crush*, and is therefore not appropriate to be included in the headline.

Moreover, the *Vanilla Human* did not receive the highest scores. We found some of the human written headlines are overly rhetorical and not easily understandable to ordinary readers (see the fourth example in Table 6.8). All NLP models achieve good performance (around 1.8 points) in text quality, which is similar to the scores of the human-written headlines.
### Table 3.4: The impact of different signature phrase selectors.

<table>
<thead>
<tr>
<th>Selector</th>
<th>Hit@1</th>
<th>Hit@3</th>
<th>Hit@5</th>
<th>Mean Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Newsroom</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>9.28</td>
<td>27.79</td>
<td>46.28</td>
<td>5.071</td>
</tr>
<tr>
<td>Holistic-N</td>
<td>18.30</td>
<td>41.82</td>
<td>57.95</td>
<td>4.395</td>
</tr>
<tr>
<td>Holistic-F</td>
<td>30.10</td>
<td>54.69</td>
<td>68.81</td>
<td>3.376</td>
</tr>
<tr>
<td>Individual-N</td>
<td>30.99</td>
<td>57.05</td>
<td>71.68</td>
<td>3.193</td>
</tr>
<tr>
<td>Individual-F</td>
<td><strong>40.34</strong></td>
<td><strong>67.57</strong></td>
<td><strong>79.64</strong></td>
<td><strong>2.395</strong></td>
</tr>
<tr>
<td>Gigaword</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>9.28</td>
<td>27.79</td>
<td>46.28</td>
<td>5.071</td>
</tr>
<tr>
<td>Holistic-N</td>
<td>16.91</td>
<td>39.56</td>
<td>58.31</td>
<td>4.142</td>
</tr>
<tr>
<td>Holistic-F</td>
<td>29.21</td>
<td>55.44</td>
<td>70.95</td>
<td>3.094</td>
</tr>
<tr>
<td>Individual-N</td>
<td>23.98</td>
<td>50.09</td>
<td>67.50</td>
<td>3.438</td>
</tr>
<tr>
<td>Individual-F</td>
<td><strong>34.05</strong></td>
<td><strong>64.01</strong></td>
<td><strong>79.71</strong></td>
<td><strong>2.426</strong></td>
</tr>
</tbody>
</table>

Table 3.5: Result of generated headlines for newsroom articles when synthesized users have different number of interest phrases.

### 3.3.4 Ablation Study

**Selectors Evaluation.** To evaluate the performance of signature selection, we rank all candidate signature phrases within an article for a synthesized user and report the following metrics: (1) Hit@K, which is the percentage of times that the correct signature phrase is ranked among the top K; (2) Mean rank, which is the average rank of the correct signature phrase. We use our synthesized user evaluation dataset to evaluate both headline generation and signature selection.

As shown in Table 3.4, *Individual-F* demonstrates the best performance among all selectors. This explains the high user adaptation scores of headlines generated by *SP individual-F*. We have observed that the selector does not always choose the gold user signature phrases, yet the generated headline still relates to user’s interests. For example, in the second example of Table 3.3, even though the user’s interested phrase...
Table 3.6: Performance of GPT-3 generated headlines compared to our SP individual-F.

<table>
<thead>
<tr>
<th>Methods</th>
<th>User Adaptation Metrics</th>
<th>Article Loyalty Metrics</th>
<th>Other Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H-U Relevance</td>
<td>REC Score</td>
<td>H-A Relevance</td>
</tr>
<tr>
<td>History Oriented (GPT-3)</td>
<td>51.76</td>
<td>0.277</td>
<td>4.277</td>
</tr>
<tr>
<td>Topic Oriented (GPT-3)</td>
<td>52.73</td>
<td>0.296</td>
<td>4.562</td>
</tr>
<tr>
<td>SP individual-F</td>
<td>54.75</td>
<td>0.330</td>
<td>4.618</td>
</tr>
</tbody>
</table>

Table 3.7: Two paradigms of applying GPT-3 in personalized headline generation. 
History Oriented uses GPT-3 to generate headlines for users based on their reading history. Topic Oriented first obtains focused signature phrases using our signature identification and selection modules, and then generates the headline based on the focused topics using GPT-3.

*Star War* was not chosen as the user signature, the generated headline is still relevant to *Star War*, as the selected signature phrase *The Force Awakens* is the subheading of a movie in the *Star War* movie series.

Factors Affecting Headline Generation. Through our experiments, we have identified that the following factors affect the quality of the generated headlines:
(1) Number of topics that the user is interested in. As shown in Table 3.5, the evaluation results of headlines generated from newsroom articles for synthesized users with varying number of interest phrases indicates that, as the number of interest phrases increases, the user adaptation scores decreases, while other scores remain roughly the same. This suggests that it is easier to generate personalized headlines for users who read news related to fewer interest phrases. However, even when the number of interest topics increases to 30, our proposed method still achieves better user adaptation scores than the vanilla systems, while showing similar performance in

---

2In this experiment, we additionally include 3 groups of synthesized users who has 10/20/30 interest topics, each single user has 50-60 news in their reading histories.
article loyalty metric. (2) Number of user signature phrases. Our analysis of generated headlines revealed that when the signature-oriented headline generator takes multiple user signature phrases as input, the generated headline may contain factual errors. This is because the generator is compelled to incorporate irrelevant signature phrases into a coherent headline, as seen in the first example in Table 3.3). As a result, we only use a single signature phrase to guide headline generation.

**Applying GPT-3 for Personalized Headline Generation.** Recently, GPT-3 [15] has been found to be effective in zero-shot prompting automatic summarization [50]. In this section, we investigate whether prompts can inspire GPT-3$^3$ to generate personalized headlines of good quality. To achieve this goal, we conduct experiment with 100 random samples from our newsroom test set using two paradigms, as shown in Table 3.7, and present the results in Table 3.6.

Our SP individual-$F$ method outperforms GPT-3 based methods in terms of user adaptation metrics and ROUGE-L score. This suggests that despite GPT-3’s strong ability in zero-shot setting, it is still incomparable to models that are specifically trained for our headline generation task. Specifically, the topic oriented method shows better performance in user adaptation metrics than the history oriented method, which implies that our topic selector effectively reveals users’ interests.

### 3.4 Conclusion

We investigate the generation of personalized headlines tailored to various users’ interests. We propose a topic-focused generation framework and methods for creating synthesized data to support the training of our framework without the need for human-annotated datasets. Additionally, we explore evaluation methods that enable

---

$^3$In our experiment, we use OpenAI’s text-davinci-003.
the automatic evaluation of the generated headlines from multiple perspectives. Our experiments demonstrate the effectiveness of our proposed approaches.
In this chapter we present novel AI-empowered chatbots for learning as conversation where a user does not read a passage but gains information and knowledge through conversation with a teacher bot. Our information-acquisition-oriented dialogue system employs a novel adaptation of reinforced self-play so that the system can be transferred to various domains without in-domain dialogue data, and can carry out conversations both informative and attentive to users (Section 4.1). In Section 4.2 we introduce our experiment settings. In Section 4.3 and Section 4.4 we report our objective and subjective results on three large public data corpora.

4.1 Approach

In order to obtain an informative and attentive teaching dialogue system, we propose a framework that consists of two chat bots in different roles, and leverage both supervised learning and unsupervised reinforcement learning, as illustrated in Figure 4.1. The unsupervised reinforcement learning enables the system to be fine-tuned on other text corpus where no annotation or dialogue data is required.
4.1.1 Model Architecture

Given a passage $P$, the conversation between Teacher bot $X$ and Student bot $Y$ can be denoted as a sequence of turns $C = \{U^X_1, U^Y_1, ..., U^X_N, U^Y_N\}$, where $N$ is the number of turns in the conversation. In order to mimic our use case, Teacher bot has access to $P$ whereas Student bot does not.

Teacher bot $X$ aims at transmitting the information in $P$ to the student. At the $n$th turn, $X$ takes as input $P$ and the conversation history $H^Y_n = \{U^X_1, U^Y_1, ..., U^X_{n-1}, U^Y_{n-1}\}$, and outputs $U^X_n$. Teacher bot $X$ adopts DoHA [128], a pre-trained model for document-grounded text generation, and is tuned in supervised phase and unsupervised self-play phase.

In order to fine-tune Teacher bot $X$ with reinforcement learning on full conversations, as a practical approach, we train a Student bot $Y$ to carry on conversations...
with $X$. Student bot $Y$ takes the conversation history $H_n^Y = \{U_1^X, U_1^Y, \ldots, U_n^X\}$ as input, and output $U_n^Y$. It adopts BART [92] model.

### 4.1.2 Phase 1: Supervised Pre-Tuning

This phase trains Teacher bot $X$ to initialize and carry out conversations based on a given passage $P$, and trains Student bot $Y$ to respond appropriately to $X$. To this end, we pre-tune both $X$ and $Y$ on the Wizard of Wikipedia (WoW) dataset. WoW was chosen as the pre-tuning dataset because of its two characteristics: 1) Open-domain: WoW contains conversations on a broad range of topics and domains across Wikipedia, thus the pre-tuned Teacher and Student bots have greater potentials to be successfully transferred to other domains during the fine-tuning stage; 2) Content-grounded: in WoW, the teaching bot’s utterances are grounded on passages, which is similar to our task. We present the gold passage to Teacher bot directly, though, different from the WoW’s original setting [29] where Teacher bot searches a large corpus for supporting passages.

We optimize the maximum-likelihood objective for Teacher bot by minimizing the following loss:

$$L_{mle}^X = -\sum_{n=1}^{N} \sum_{m=1}^{M_n} \log(p(x_m|x_1, \ldots, x_{m-1}, H_n^X, P))$$

where $N$ is the total number of turns in the conversation, $\{x_1, \ldots, x_{M_n}\}$ is Teacher bot’s response at the $n_{th}$ turn, $M_n$ is the number of words in $U_n^X$. The loss function for Student bot, $L_{mle}^Y$, is similar to $L_{mle}^X$, with the exception of not including a passage $P$ as input.

$$L_{mle}^Y = -\sum_{n=1}^{N} \sum_{m=1}^{M_n} \log(p(y_m|y_1, \ldots, y_{m-1}, H_n^Y))$$

where $\{y_1, \ldots, y_{M_n}\}$ is Student bot’s response at the $n_{th}$ turn.
4.1.3 Phase 2: Unsupervised Self-Play Fine-tuning

In this phase, we aim at improving Teacher bot’s ability to present informative and coherent conversations. This is achieved by reinforcement learning on Teacher bot with the help of Student bot, and could be applied in a novel target domain even where dialogue dataset is absent. We adopt a self-play approach, i.e. we let Teacher bot and Student bot chat with each other over a passage in the target domain to generate multiple turns of conversations. In this fine-tuning phase, we keep Student bot frozen, and reward Teacher bot when the generated conversation achieves higher scores. In order to reduce the variance of the gradient estimate, we apply self-critic reinforcement learning [139]. Specifically, at each turn, we let Teacher bot generate two separate utterances: 1) $U^s$, which is sampled from the model, i.e. $x^s_m \sim p(x|x^s_1, ..., x^s_{m-1})$, and 2) $U^*$, which is obtained by greedy search, i.e. $x^*_m = \arg \max_w p(x|x^*_1, ..., x^*_{m-1})$. We optimize the model by minimizing the following RL loss:

$$L_{rl}^X = - \sum_{n=1}^{N} (R(U^s_n) - R(U^*_n)) \sum_{m=1}^{M_n} \log(p(x^s_m|x^s_1, ..., x^s_{m-1}, H^X_n, P))$$

where $R()$ is the reward function, which we will cover in Section 4.1.4, and $P$ is a passage from the target domain corpus.

If not taking into account language modeling, optimizing RL loss alone would lead Teacher bot to generate inarticulate and even grammatically incorrect utterances. To keep the fluency of Teacher bots, we optimize a combined loss $L^X$ consisting of RL loss $L_{rl}^X$ on the new target domain data and MLE loss $L_{mle}^X$ on the pre-tuning dialogue dataset, so the language style acquired during the pre-tuning phase would not get lost during RL fine-tuning:

$$L^X = \gamma L_{rl}^X + (1 - \gamma) L_{mle}^X$$
where $\gamma \in (0,1)$ is a scaling factor accounting for the emphasis on $L_{n}^{X}$ and $L_{mle}^{X}$.

We note that while $L_{mle}^{X}$ should be obtained on an annotated content-grounded dialogue dataset (e.g. WoW), $L_{rl}^{X}$ could be obtained on any target domain passage corpus even without dialogue data. This enables our approach to be transferred to an unsupervised text corpus.

4.1.4 Reward Functions

4.1.4.1 Coverage

We define the coverage reward of a Teacher bot’s utterance $U^{X}$ as:

$$R_{cov} = \text{Rouge}_1(P, H + U^{X}) - \text{Rouge}_1(P, H)$$

where Rouge$_1(P, H)$ is the Rouge-1 F1 [99] score of the conversation history $H$ to the input passage $P$. Intuitively, this function favors utterances that cover more information in the passage and have less overlap with the conversation history.

4.1.4.2 Coherence

Dialogue coherence datasets We explore neural coherence scoring models trained on two open-domain dialogue coherence datasets:

1. **WoW-coherence dataset** We reuse the WoW dataset to heuristically build a dialogue coherence classification dataset. Specifically, for each multi-turn dialogue in WoW, we label the ground truth response to its conversation history as *coherent* response, and all later responses in the same dialogue as *incoherent* responses.

2. **InferConv dataset** [33] This is an open-domain dialogue coherence classification dataset built from PersonaChat conversational data [197]. The dataset casts a response as the hypothesis and the conversation history as the premise, thus convert dialogue coherence evaluation into an NLI task. The dataset classifies the relation-
ship between the response and the conversation history into three categories: entailed, neutral and contradict. Table 4.2 summarizes statistics of these datasets.

**Coherence scoring models** Based on the same pre-trained model BERT [26], we train two different coherence scoring models on the two dialogue coherence classification datasets respectively. Both models take the concatenation (with [SEP]) of the conversation history and a candidate response as input, and minimize the cross entropy loss between the predicted label and the gold label. We use different methods to attain the coherence reward $R_{coh}$ from the two models.

For model WoW-coherence, we define the coherence reward with softmax:

$$R_{coh} = \frac{e^{o_c}}{e^{o_c} + e^{o_i}}$$

where $o_c$ and $o_i$ are the logits for coherent and incoherent labels in the output layer.

For model InferConv, we observe some responses labeled as neutral are appropriate responses but are not closely related to conversation history (e.g. “That’s interesting!”), we thus heuristically assign constant scores $s_e$, $s_n$ and $s_c$ as coherence reward $R_{coh}$ when the response is predicted as entailed, neutral and contradict. In the remainder of the paper, we use WoW-coherence as the default coherence model, and compare it with InferConv in Section 4.3.

### 4.1.4.3 Mixed Reward

The coverage and coherence rewards are combined with a hyper-parameter $\beta$, yielding the final reward:

$$R = \beta R_{cov} + (1 - \beta) R_{coh}$$
4.2 Experimental Setting

We proceed by describing our datasets, comparison systems and evaluation metrics. We then show the performance of our proposed approach compared to state-of-the-art in Section 4.3.

4.2.1 Datasets

*Wizard of Wikipedia* [29] contains a total of 22,311 human-human conversations crowdsourced via Amazon’s Mechanical Turk. The conversations are grounded in Wikipedia passages covering a wide range of topics: *e-book, toga party, armadillo*, etc. Both Teacher and Student bots are pre-tuned on the WoW dataset during Phase 1. Different from WoW’s original setting, we present the gold passage to the Teacher bot directly, instead of searching a large corpus for supporting passages. This allows us to focus less on retrieval and more on creating a Teacher bot to deliver informative and attentive dialogues.

We consider knowledge sources of various sorts as Teacher bot’s target domain during fine-tuning. CNN/DailyMail contains a large collection of online news articles with an average of 781 tokens per article [146]. The full content of the article cannot be conveyed in a short conversation. Thus, we use the first 130 tokens of each article as a supporting passage, assuming it covers the most important content of the news article.

Academic papers have become an omnipresent source of knowledge. We create our own dataset containing papers published in recent years (2017–2021) at major venues, including ACL, EMNLP, NAACL, EACL, Findings and ICLR conferences. Similarly, we use paper abstracts as supporting passages instead of full articles. Moreover, we include Wikipedia passages from the WoW dataset, without conversations, as another source of knowledge. The CNN-DM, *Paper Abstracts* and *Wikipedia* datasets are
Table 4.1: Datasets statistics.

<table>
<thead>
<tr>
<th>Pre-Tuning</th>
<th>WoW-Train</th>
<th>WoW-Valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Utterances</td>
<td>166,787</td>
<td>17,715</td>
</tr>
<tr>
<td>#Dialogues</td>
<td>18,430</td>
<td>1,948</td>
</tr>
<tr>
<td>#Words/utterance</td>
<td>16.6</td>
<td>16.6</td>
</tr>
<tr>
<td>#Words/passage</td>
<td>110.3</td>
<td>109.8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>#Passages</td>
<td>50,000</td>
<td>50,000</td>
<td>22,512</td>
</tr>
<tr>
<td>#Words/passage</td>
<td>111.7</td>
<td>129.8</td>
<td>149.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test Set</th>
<th>Wikipedia</th>
<th>CNN-DM</th>
<th>Paper Abs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Passages</td>
<td>1,000</td>
<td>1,000</td>
<td>500</td>
</tr>
<tr>
<td>#Words/passage</td>
<td>112.7</td>
<td>129.9</td>
<td>148.1</td>
</tr>
</tbody>
</table>

Table 4.2: Details of the datasets used to train our neural coherence scoring models.

used in Phase 2 of unsupervised self-play fine-tuning. Statistics of these datasets are summarized in Table 4.1.

4.2.2 Comparison Models

Our baseline Teacher bot builds on the state-of-the art content-grounded dialogue generation model: DoHA [128]. It includes two improvements to the architectures of pre-trained encoder-decoder models [92]: building context-driven representation of the supporting document, and enabling document-headed attention to acquire information from the document. DoHA has demonstrated strong performance in document-grounded generation. All DoHA models are pre-tuned on the WoW dataset.

Our FULL Teacher bot is created to converse in an informative and coherent manner. It extends DoHA by incorporating both coverage and coherence rewards in
Coherence
3 TeacherBot provides coherent responses to the evaluator’s input.
2 TeacherBot provides largely coherent responses (with minor coherency issues) to the evaluator’s input.
1 TeacherBot does not respond properly to the evaluator’s input.

Readability
3 TeacherBot’s responses are easy to read, containing no grammatical or semantic errors.
2 TeacherBot’s responses read smoothly but may contain 1-2 grammatical or semantic errors.
1 TeacherBot’s responses contain > 2 grammatical or semantic errors, or are nonsensical.

Overall Quality
An initial score of 3 is given to a dialogue, then 1 point is deducted for each of the following issues, with a minimum score of 0.
- Uninformative, i.e. it provides < 2 correct answers during QA.
- Incoherent, i.e. the average coherence score is < 2 points.
- Low readability, i.e. the average readability score is < 2.5 points.
- Any other issues that could lead to an ineffective conversation e.g. words are repeated between turns.

Table 4.3: A scoring rubric provided to human evaluators.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>Coverage Metrics</th>
<th>Coherence Metrics</th>
<th>Subjective Metrics</th>
<th>Avg Len</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>R-1</td>
<td>R-2</td>
<td>R-L</td>
<td>QA_Cov</td>
</tr>
<tr>
<td>Wiki</td>
<td>DoHA</td>
<td>48.99</td>
<td>41.27</td>
<td>44.31</td>
<td>19.76</td>
</tr>
<tr>
<td></td>
<td>+Cov</td>
<td>74.02</td>
<td>71.06</td>
<td>72.25</td>
<td>30.90</td>
</tr>
<tr>
<td></td>
<td>Full</td>
<td>62.78</td>
<td>58.83</td>
<td>60.61</td>
<td>25.94</td>
</tr>
<tr>
<td>CNN-DM</td>
<td>DoHA</td>
<td>38.89</td>
<td>30.09</td>
<td>32.91</td>
<td>15.90</td>
</tr>
<tr>
<td></td>
<td>+Cov</td>
<td>81.52</td>
<td>78.52</td>
<td>73.73</td>
<td>30.08</td>
</tr>
<tr>
<td></td>
<td>Full</td>
<td>65.77</td>
<td>60.45</td>
<td>57.83</td>
<td>25.5</td>
</tr>
<tr>
<td>Papers</td>
<td>DoHA</td>
<td>36.27</td>
<td>28.20</td>
<td>30.60</td>
<td>10.29</td>
</tr>
<tr>
<td></td>
<td>+Cov</td>
<td>72.63</td>
<td>69.69</td>
<td>49.18</td>
<td>20.32</td>
</tr>
<tr>
<td></td>
<td>Full</td>
<td>59.65</td>
<td>54.22</td>
<td>47.88</td>
<td>15.46</td>
</tr>
</tbody>
</table>

Table 4.4: We compare Teacher bots based on naive DoHA[128] model to variants fine-tuned using different reward functions, Avg len refers to average utterance length, WC refers to WoW-coh, IC refers to InferConv-Coh

unsupervised self-play fine-tuning. Additionally, we ablate Full model by removing each of the two rewards: +Cov uses only the coverage reward for fine-tuning, i.e. setting $\beta = 1$ in our reward function (Section 4.1.4). +Coh utilizes only the WoW-coherence reward, i.e. setting $\beta = 0$. Please refer to appendix for more implementation details and hyper-parameters.

4.2.3 Evaluation Metrics

We investigate a wide range of metrics to evaluate Teacher bot’s performance. Objective metrics measure the content coverage and coherence of Teacher bot’s utterances. Subjective metrics, devised with human-in-the-loop, provide a holistic eval-
uation of a conversation, focusing on its overall effectiveness and various aspects of linguistic quality.

**Objective Metrics.** Teacher bot converses with Student bot over a passage for three turns. That is, Teacher bot initiates the dialogue and provides two responses to Student bot. We objectively evaluate Teacher bot’s utterances in terms of information coverage and coherence as follows.

- **ROUGE** [99] is one of the most widely used metrics for measuring information coverage. We consider three variants in this study: R-1, R-2 and R-L, which respectively measure the overlap of unigrams, bigrams and the longest common subsequence between the given passage and Teacher bot’s utterances.

- **QA\(F_1\) and QA\(CONF\)** are two variants of SummaQA [145], a question answering-based evaluation metric. If a conversation is rich in information, it could be used as a surrogate for the passage to answering important questions. To this end, SummaQA generates Cloze-style questions from a passage by masking out entities, then employs a QA system to answer those questions based on a conversation. A higher QA performance suggests the conversation has better coverage. Particularly, QA\(F_1\) reports the F1 score for question answering; QA\(CONF\) measures the confidence of the QA system in predicting answers.

- **Wow-Coherence** and **InferConv** are neural coherence scoring models (Section 4.1.4.2) repurposed for evaluation. These models quantitatively assess if Teacher bot has produced a coherent response given the conversation history, or not.

- **DPR\(_{RELV}\)** provides a new perspective on dialogue coherence evaluation [194]. It draws on the Dense Passage Retriever model (DPR; Karpukhin et al., 2020) to predict if a Teacher bot’s response is relevant to Student bot’s input. A higher relevance score means the input and response share the same topic, suggesting a coherent conversation.
Subjective Metrics. We recruit 24 human evaluators to interact with Teacher bots. Each evaluator is asked to converse with bots over four passages. For each passage, the evaluator chats with four different Teacher bots for three turns, where Teacher bot initiates the conversation and responds twice to the evaluator’s input. We randomly select 48 passages for evaluation, i.e., 16 passages from each of the three test sets. To evaluate conversations produced from Paper Abstracts, we require evaluators, 8 in total, to be either PhD students or have obtained a PhD degree. For fair comparison, we shuffle and hide the order of Teacher bots presented to evaluators. Human evaluators were suggested to feed the same or similar inputs across Teacher bots on the same passage whenever possible. Throughout the conversation, the passage was not shown to the evaluators. After the conversation, human evaluators were asked to complete the following evaluation tasks:

- $QAH_{\text{HUMAN}}$: Five sentences are randomly selected from each passage and one important entity is masked out in each sentence. The evaluators are presented with each corrupted sentence and asked if the sentence could be recovered by referencing the conversation with Teacher bot. We report the ratio of sentences that could be correctly recovered.

- Linguistic Quality: We ask human evaluators to rate each conversation along three dimensions: Coherence: Does Teacher bot provide coherent responses to the evaluator’s input? Readability: Are Teacher bot’s utterances easy to read, containing no grammatical or semantic errors? Overall Quality: How will the conversation score in terms of informativeness, coherence, readability and all aspects considered? The scoring rubric provided to human evaluators is shown in Table 5.5.

4.3 Objective Results

Results on Test Sets. Table 4.4 presents objective evaluation results obtained for various Teacher bots on three test sets: CNN-DM, Paper Abstracts and Wikipedia.
Figure 4.2: ROUGE gain and utterance length tend to decrease as the number of turns increases.

We observe that our Full Teacher bot is able to substantially outperform the baseline system DoHA on all datasets and across all objective metrics. It strikes a fine balance between delivering information-rich conversations and ensuring those conversations are coherent and attentive. Further, we find that optimizing a single reward, whether it be coverage or coherence, produces suboptimal results. For instance, $+\text{Cov}$ tends to produce longer utterances than other variants. It improves information coverage, but yields low coherence scores, leading to a performance even inferior to the baseline DoHA. Our findings suggest that it is important for the reinforcer $L_{rl}^X$ to learn with both coverage or coherence rewards.

**Trading off Coverage for Coherence.** In Figure 4.3, we plot the learning curves of coverage and coherence scores when the reinforcer adopts a single reward ($+\text{Cov}$, $+\text{Coh}$) or both (Full). We use $R_{\text{cov}}$ and $R_{\text{coh}}$ to approximate coverage and coherence scores. These plots are generated using 50 validation instances from the Paper Abstracts dataset. We observe that with only the coverage reward ($+\text{Cov}$), Teacher bot tends to aggressively copy content from the passage, while disregarding the conversation history. This inevitably leads to incoherent conversations. Conversely, $+\text{Coh}$ can improve on coherence, but falls short on delivering informative conversations. Finally, our Full Teacher bot trades off coverage for substantially
Figure 4.3: Coverage and coherence scores when applying different reward scores

higher coherence, thus achieving a significant improvement over the baseline DoHA model.

**Measure of Information Gain** We are curious to know the amount of information brought by each utterance produced by Teacher bot. To this end, we define information gain $\mathcal{I}_G(\cdot)$ as the improvement of ROUGE scores brought by an utterance $U$:

$$\mathcal{I}_G(U) = \text{ROUGE}(P, H + U) - \text{ROUGE}(P, H),$$

where $P$ is the supporting passage, and $H$ represents the conversation history. We consider three ROUGE variants, R-1, R-2 and R-L, respectively. Figure 4.2 illustrates the gain of information for each of the three turns. The average ROUGE gain is reported for each turn, using conversations produced for the *Paper Abstracts* dataset. We observe that there is a general tendency across turns that information gain is decreasing. This is in part because that at the beginning of a conversation, Teacher bot has no constraints regarding content selection, it could rephrase any content selected from the supporting passage to initiate a dialogue. In subsequent turns, Teacher bot has to exercise caution in response generation considering both the conversation history and overall coherence of the conversation. Consequently, we find that the average length of the utterances also decreases in subsequent turns.
Table 4.5: Teacher bot’s performance when using different coherence score models during fine-tuning

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>Coverage $Q_A^{CoV}$</th>
<th>$Q_A^{F1}$</th>
<th>Coherence WoW-Coh</th>
<th>IC</th>
<th>DPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wikipedia</td>
<td>WoW</td>
<td>18.96</td>
<td>17.17</td>
<td>0.807</td>
<td>0.694</td>
<td>0.578</td>
</tr>
<tr>
<td></td>
<td>InferCov</td>
<td>20.41</td>
<td>19.61</td>
<td>0.647</td>
<td>0.676</td>
<td>0.561</td>
</tr>
<tr>
<td>CNN-DM</td>
<td>WoW</td>
<td>13.31</td>
<td>11.31</td>
<td>0.845</td>
<td>0.692</td>
<td>0.561</td>
</tr>
<tr>
<td></td>
<td>InferCov</td>
<td>14.95</td>
<td>14.07</td>
<td>0.719</td>
<td>0.728</td>
<td>0.550</td>
</tr>
<tr>
<td>Papers</td>
<td>WoW</td>
<td>8.18</td>
<td>3.95</td>
<td>0.806</td>
<td>0.547</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td>InferCov</td>
<td>9.47</td>
<td>4.31</td>
<td>0.737</td>
<td>0.554</td>
<td>0.558</td>
</tr>
</tbody>
</table>

A Comparison of Coherence Scoring Models. We compare two Teacher bots fine-tuned only with coherence reward from different coherence scoring models (i.e. WoW-coherence and InferConv). We demonstrate their objective results in Table 4.5. According to the results, Teacher bot fine-tuned with InferConv achieves slightly better coverage metrics. However, in terms of coherence metrics, Teacher bot fine-tuned with WoW-coherence model generally achieves better performance. Based on this observation, WoW-coherence scoring model better measures coherence in conversations.

4.4 Subjective Results

We demonstrate the subjective evaluation results in Table 4.4 and have the following observations:

1. For question answering, $+\text{COV}$ achieves the best performance on all three datasets. This again proves that the coverage reward helps make the conversation more informative;

2. For coherence scores, $+\text{COH}$ achieves the best performance on Wikipedia. However, on the other two datasets it was outperformed by FULL.

3. For readability scores, on CNN-DM and Paper Abstracts, $+\text{COH}$ achieves the lowest performance while $+\text{COV}$ achieves the highest.
4. For overall scores, **FULL** demonstrates the best performance. This suggests **FULL** delivers conversations that are more balanced in coverage, coherence and readability.

5. The *Paper Abstracts* corpus is the most challenging among all the test corpora, as our Teacher bots generally show worse performance in coherence, readability and overall scores. We found passages in *Paper Abstracts* contain volumes of professional vocabularies thus are more complicated for people to understand. In addition, it’s also more difficult for Student bot to respond appropriately during self-play fine-tuning (See examples in appendix). As a result, transferring Teacher bots to this domain is more challenging.

**A Case Study.** We show a few Teacher bots’ responses to users in Table 4.6. After analyzing the cases, we have the following observations:

1. The coverage reward encourages Teacher bots to directly copy content from the input passage, while the coherence reward encourages abstractively generating new content: As shown in Example A, +Cov directly extracts a part of the original passage as response, regardless of the user’s question, while +Coh and FULL abstractively rewrite the response to make it more coherent.

2. Putting too much weight on coherence reward could make Teacher bot become so abstractive that it misrepresents the original passage and lead to incoherence and semantic/grammar errors. (See +Coh’s response in example A and C) This explains +Coh’s low coherence and readability scores on CNN-DM and Paper Abstracts. This observation suggests the necessity to carefully choose the weight for coherence rewards, and to coupling coherence reward with coverage reward, which could make the chat bot less abstractive.

3. Generally, user utterances could be classified into two categories: **Information-seeking queries** which request certain information (e.g. the user’s utterance in
Table 4.6: A few example responses from different Teacher bots. In example A, +COH and FULL abstractively generates *Concord is famous for ...* to make the response more coherent to user’s question. However, the underlined part in +COH’s statement misrepresent the input passage and is inaccurate. In example B, all responses seem coherent because of the open-ended question.

example A): **Open statements** which do not have specific requests (e.g. the user’s utterance in example B). We found evaluators tend to give high coherence scores to response to open statements, as they could be addressed by a wider range of responses.

### 4.5 Conclusion

We propose an information-acquisition-oriented dialogue system that transfers information and knowledge in passages to users through conversations. An unsupervised self-talk approach is introduced leveraging novel rewards to enable Teacher bots to deliver informative and attentive conversations. Experiments with automatic and human evaluations demonstrate the effectiveness of our approach. Some inter-
esting future directions include extending the conversations to be based on a set of documents and specializing our dialogue systems for specific domains, e.g. patient education.
CHAPTER 5
GENERATION OF PATIENT AFTER-VISIT
SUMMARIES TO SUPPORT PHYSICIANS

In this chapter, we study the problem of automatic generation of after-visit summaries and examine whether those summaries can convey the gist of clinical visits (Section 5.1). Crucially, we introduce a feedback mechanism that alerts physicians when an automatic summary fails to capture the important details of the clinical notes or when it contains hallucinated facts that are potentially detrimental to the summary quality (Section 5.2). We report our findings on a new clinical dataset that contains a large number of electronic health record (EHR) notes and their associated summaries (Section 5.3). We further discuss the characteristics of this task in Section 5.4.

5.1 Summarization

Our method generates an after-visit summary from EHR notes concerning a patient. It is modelled as a single-document summarization task as the EHR notes were collapsed into a single document by the hospital and we were unable to recover individual EHR notes. We use $S=${$w_1, ..., w_{n_S}$} to denote tokens of the source document and $T=${$w_1, ..., w_{n_T}$} tokens of the target summary, $n_S$ and $n_T$ are length of the sequences.

We explore a variety of summarization models to generate after-visit summaries. They are detailed in Table 5.1. Particularly, an abstractive summarizer employs the standard Transformer-based encoder-decoder model to generate a summary $P(T|S)$. 

56
• **BART** [92] uses the standard encoder-decoder architecture. It was pretrained as a denoising autoencoder to learn to reconstruct the original text. Our input to the BART model consists of a clinical document and its output is an abstractive summary.

• **PEGASUS** [195] explores a new pretraining objective tailored for abstractive summarization. Important sentences are masked out from the input document and the model learns to generate the sentences as an output sequence, akin to an extractive model. The system has been shown to perform well in a low-resource scenario where few examples are available for fine-tuning.

• **LED** [10] is the Longformer-Encoder-Decoder model. It is an extension to Longformer to support text generation. LED uses a local windowed attention which makes it computational feasible to encode a long input document. We favor the LED model because, compared to news articles, there is more risk involved in truncating long clinical documents to a certain length.

• **BertSum** [102] employs the BERT model to identify summary-worthy sentences. It uses a flat architecture to encode the input document, then adds a Transformer layer on top of the sentence representations to model inter-sentence relationship. The final output layer is a sigmoid classifier used to predict if the sentence is to be included in the summary.

• **TextRank** [110] and **LexRank** [35] are graph-based models that extract relevant sentences based on eigenvector centrality.

• **Oracle Top-K** [2] is a method introduced by Adams et al. which represents the upper bound for sentence extraction. It ranks all document sentences according to their averaged R-1 and R-2 scores with respect to the reference summary. It then continues to add sentences yielding the highest scores to the summary until the target token count is reached.

<table>
<thead>
<tr>
<th>Abstractive Summarization Model</th>
<th>Extractive Summarization Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>• <strong>BART</strong> [92] uses the standard encoder-decoder architecture. It was pretrained as a denoising autoencoder to learn to reconstruct the original text. Our input to the BART model consists of a clinical document and its output is an abstractive summary.</td>
<td>• <strong>BertSum</strong> [102] employs the BERT model to identify summary-worthy sentences. It uses a flat architecture to encode the input document, then adds a Transformer layer on top of the sentence representations to model inter-sentence relationship. The final output layer is a sigmoid classifier used to predict if the sentence is to be included in the summary.</td>
</tr>
<tr>
<td>• <strong>PEGASUS</strong> [195] explores a new pretraining objective tailored for abstractive summarization. Important sentences are masked out from the input document and the model learns to generate the sentences as an output sequence, akin to an extractive model. The system has been shown to perform well in a low-resource scenario where few examples are available for fine-tuning.</td>
<td>• <strong>TextRank</strong> [110] and <strong>LexRank</strong> [35] are graph-based models that extract relevant sentences based on eigenvector centrality.</td>
</tr>
<tr>
<td>• <strong>LED</strong> [10] is the Longformer-Encoder-Decoder model. It is an extension to Longformer to support text generation. LED uses a local windowed attention which makes it computational feasible to encode a long input document. We favor the LED model because, compared to news articles, there is more risk involved in truncating long clinical documents to a certain length.</td>
<td>• <strong>Oracle Top-K</strong> [2] is a method introduced by Adams et al. which represents the upper bound for sentence extraction. It ranks all document sentences according to their averaged R-1 and R-2 scores with respect to the reference summary. It then continues to add sentences yielding the highest scores to the summary until the target token count is reached.</td>
</tr>
</tbody>
</table>

Table 5.1: State-of-the-art summarization models investigated in this work for generation of patient after-visit summaries.

An extractive summarizer selects important sentences to add to the summary until a length threshold has been reached. These systems are used off-the-shelf and have achieved some of the highest reported scores on summarization. We assess their ability to navigate complex medical terrain for generation of after-visit summaries.

Clinical notes are complex and full of references to medical events. However, the summary given to the patient is simple and clear. We are thus curious to know how medical events manifest themselves in the context of summarization. Events are especially important for this task, as salient events happening at each medical encounter must be included in the after-visit summary.

We define *event nugget* as a word or multi-word phrase that clearly expresses the occurrence of a medical event. Event nuggets are identified by MetaMap [6], an open-source software tool designed to discover medical concepts referred to in a text. Each occurrence of the concept is assigned a concept unique identifier (CUI) and its associated words are tagged in the text. In Figure 5.2, we show an example of medical concepts identified by MetaMap. Further, those medical concepts are categorized into

57
Various semantic types. We focus on concepts pertaining to a selected set of entity and event types (Table 5.2), which are deemed relevant by medical experts. The other types are excluded from consideration.

### 5.2 Error Detection

Event nuggets are associated with type I and type II errors frequently found in the summary. A **type I error** indicates a summary contains a hallucinated fact or event that is not present in the source document. A **type II error** suggests that an important medical event has been mistakenly left out of the summary, hampering its usability. In this section, we describe novel methods to detect likely errors and flag them in the text to alert clinicians.

#### 5.2.1 Type I Error: Hallucination

A **hallucination detector** aims to recognize any hallucinated content in a summary. Flagging errors is helpful because physicians can be alerted about any anomalies and it is especially appreciated in medical domain [150]. Our detector uses the BigBird model [191], which is an encoder-only architecture equipped with sparse attention to reduce Transformer’s quadratic complexity to linear, and capable of encoding thousands of tokens. The model takes as input a source document ($S$) and its system summary ($T$), and outputs a sequence of binary labels, one for each summary token, where 1 represents the token is considered hallucinated and 0 otherwise.
A key factor to the success of our model is its self-supervised training, where a large number of training instances are constructed from unlabeled data. Each training instance is a synthesized summary whose hallucinated tokens are flagged. We adapt the model of Zhou et al. [207], initially proposed for MT, to create our training instances. Our method differs from theirs in that, synthesized summaries are required to contain hallucinated medical events that are inconsistent with or unjustified by the source document.

**Synthesizing Erroneous Summaries.** The procedure for generating synthesized summaries is illustrated in Figure 5.1. Given a summary sentence, we mask out one or two of its event nuggets. It is then fed to a denoising auto-encoder [92] to produce an output sentence, whose masked-out positions are refilled with medical events that are “hallucinated” by the model. If the output is substantially different from the input, e.g., with < 50% token overlap, it is called a synthesized sentence with hallucinations. Tokens of the synthesized sentence, which cannot be aligned to the original sentence using an edit-distance-style algorithm, are flagged. E.g., “cardiac catheterization” and “abnormalities” in our example are clearly hallucinated facts. This procedure is repeated for all sentences of the reference summary to create a synthesized summary. We provide examples of synthesized sentences in the Supplementary.

Importantly, the model is fine-tuned to enable it to produce plausible synthesized summaries. We partition the training data into $K$ folds ($K=5$) of roughly equal size. The BART model is fine-tuned on the union of the $K$-1 folds, then applied to the remaining fold to generate synthesized summaries. The method transforms each reference summary of the dataset to a synthesize summary, which together with the source document, is used to train and test our type I error detector.

---

1If a summary sentence does not contain any medical event, it is left as-is in the synthesized summary. The original summary sentence is otherwise replaced by a synthesized sentence.
Figure 5.1: One or two event nuggets are randomly masked out from a summary sentence (a). The masked sequence (b) is fed to a denoising auto-encoder to produce a synthesized sentence that may contain hallucinated medical events (c).

5.2.2 Type II Error: Missing Content

Our missing content detector seeks to accomplish two objectives: 1) to detect salient medical events on a clinical document, and 2) to flag salient events that are missed by the summary. It is a non-trivial task to fulfill these objectives. Even though clinical notes are full of references to medical events, only a selective portion of them (≈18%) are included in after-visit summaries. As such, we formulate the problem as a classification task. An event nugget is assigned a label of 1 if it is salient, 0 otherwise. Our detector leverages self-supervised learning to identify salient events on EHR notes. It then alerts clinicians if the summary fails to include any of the salient events.

Pseudo-Annotations for Salient Events.

We create pseudo-annotations for salient events by aligning each source event with one of the target events. As shown in Figure 5.2, the medial events are identified by MetaMap [6]. Each occurrence of the event is associated with a concept unique identifier (CUI). Under the strict matching criterion, an event of the clinical document is labeled as 1 if an exact match (with the same CUI) is found in the summary. However, a large number of events are not well-aligned under this criterion due to distinct expressions used in clinical notes and summaries. This discrepancy in language use
Figure 5.2: “abdominal pain” appears in both the clinical document and after-visit summary, with the same CUI. “nausea vomiting” and “nauseous” are aligned because there is an is-a relation between the two concepts.

has its origin—clinical notes are physician-oriented, whereas after-visit summaries are patient-oriented. We explore lenient matching to alleviate mismatch. If a source event can reach any of the target event via a single hop on the UMLS semantic graph, the source event is labeled as salient. In Figure 5.2, source event “nauseous” is leniently matched to target event “nausea vomiting,” because there exists an is-a relation between the two events.

We fine-tune the BigBird model [191] to detect salient events. Being an encoder-only model, BigBird constructs contextualized representations for all tokens of a clinical document. It does not directly produce event representations. To address this issue, we let the model predict salient tokens during training. If a token is part of a salient source event, its gold-standard label is 1. At test time, the model generates token-level predictions. A source event is considered salient if any of its tokens is labeled as 1.

2https://www.nlm.nih.gov/research/umls/META3_current_relations.html
BASE +Pos [E] CT scan [/E] showed worsening of his [E] diverticulitis [/E] with a 5.6 x 3.9cm multiloculated fluid collection in his abdomen.

BASE +Type [Type1] CT scan [/Type1] showed worsening of his [Type2] diverticulitis [/Type2] with a 5.6 x 3.9cm multiloculated fluid collection in his abdomen.

Table 5.3: Model variants +Pos and +Type aim to inform the model about the occurrences of events identified by MetaMap.

We explore two variants of the model to allow it to better capture events. Both variants aim to inform the model about the occurrences of event nuggets identified by MetaMap. The first variant, +Pos, modifies the source sequence by inserting special tokens respectively at the beginning and end of a candidate event. The second variant, +Type, inserts different special tokens such that they correspond to the semantic types of the events. We conjecture that certain event types, e.g., body substance, are more likely be considered insignificant. In Table 5.3, we provide examples comparing the source sequences used by model variants.

5.3 Experiments

In this section, we describe our dataset, perform in-depth analyses on our models, and discuss feedback from physicians who participated in our qualitative evaluation.

5.3.1 Dataset

Through a collaboration with University of Massachusetts Chan Medical School, we are able to use their electronic health record database, which gives us access to 31,895 EHR notes and their physician-written summaries. All medical records are de-identified to protect patient privacy. These patients were admitted to the medical and surgical services of the hospital from October 2017 to March 2020.

Table 5.4 summarizes the statistics of our dataset. It is divided into train, validation, and test sets containing 28,157, 1,884 and 1,854 instances, respectively. The dataset has unique characteristics. We observe that the source documents are sub-
Train / Validation / Test Split  
28,157 / 1,884 / 1,854

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of words per clinical document</td>
<td>523.6 ± 464.3</td>
<td>Number of words per after-visit summary</td>
</tr>
<tr>
<td>Event per clinical document</td>
<td>42.8 ± 32.0</td>
<td>Event per after-visit summary</td>
</tr>
<tr>
<td>Nuggets occurring in both (lenient match)</td>
<td>7.9 ± 6.1</td>
<td>Nuggets occurring in both (strict match)</td>
</tr>
</tbody>
</table>

Table 5.4: Statistics of our dataset.

substantially longer and contain more medical events than their summaries. This is because most hospitalization details are omitted for patients. In addition, the length of clinical documents varies considerably, so is the case for summaries. A long clinical document could be the result of an extended hospital stay. An after-visit summary could be long or short depending on the patient’s medical conditions. In contrast, variation in length is less significant in other genres such as news and scientific articles.

We find that an average summary contains 12.3 medical events, yet only 7.9 of them can be linked to events of the clinical document. The gap is partially due to using MetaMap for medical event identification [137], which has a reported F-Score of 0.88 and may miss out-of-vocabulary event tokens. Additionally, physicians may add their instructions directly to patient’s after-visit summaries, and such content is not grounded in clinical documents.

5.3.2 Evaluation Metrics

**Quantitative Measures.** We evaluate the performance of our summarization and error alert models with a variety of quantitative measures.

- **ROUGE** [99] is the standard measure for summarization evaluation. It assigns a high score to a system summary if it has lexical overlap with the reference summary.
• **BERTScore** [198] is one of the new evaluation metrics for natural language generation that are built on contextualized representations produced by BERT and similar models.

• **SARI** [181] is widely used for simplification. It counts how often a system summary correctly keeps, deletes, and adds n-grams.

• **DaleChall** [24] calculates the readability of the summary based on its sentence length and number of difficult words in it. It is an improvement upon Flesch’s reading ease score.

• **P/R/F** scores are reported for error alert models on successful detection of missing medical events and detection of hallucinated summary tokens.

**Qualitative Measures.** In high-stake scenarios, automatic metrics alone cannot guarantee a good system. Thus, we need expert assessments by medical practitioners in this study. We recruit six human evaluators: five of them are physicians with M.D., one is a M.D. student. Owing to budget constraints, we select a random set of 18 clinical documents and their best system summaries for qualitative assessment. The system summaries are produced by the LED model, they are abstractive. Each summary is judged by two human evaluators, who perform two tasks on a summary:

• **Scoring.** A summary is rated along four dimensions. *Adequacy:* Does the summary contain all necessary information for the patient to know? *Faithfulness:* Does the summary faithfully convey the content of the clinical document? *Readability:* Is the summary easy to read for a lay person? *Ease of Revision:* How long might it take for a physician to revise the summary to meet the expectations of standard AVS? The scoring scale is from 1 (worst) to 3 (best). Their interpretations are provided in Table 5.5.

• **Revision.** We ask human evaluators to edit the summary until it meets the expectations of standard after-visit summaries. We report the edit distance between
### Adequacy
3. AVS contains all the information the patient needs to know
2. AVS misses some (1-3) points the patient needs to know
1. AVS misses more than 3 points

### Faithfulness
3. AVS contains no or only a few errors that are ignorable
2. AVS contains some (1-3) factual errors
1. AVS contains more than 3 factual errors

### Readability
3. AVS is easy to read for a lay person
2. AVS has some (1-3) points hard to be understood by the patient
1. AVS has more than 3 points hard to be understood by the patient

### Ease of Revision
3. Physician may spend \(<=2\) minutes to revise the AVS
2. Physician may spend \(>2\) minutes to revise the AVS
1. Physician prefers to not revise the AVS but rewrite from scratch

<table>
<thead>
<tr>
<th>Model</th>
<th>Model</th>
<th>R-1</th>
<th>R-2</th>
<th>R-3</th>
<th>R-4</th>
<th>R-L</th>
<th>BertS</th>
<th>SARI</th>
<th>DaleC↓</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEXT-RANK</td>
<td></td>
<td>25.71</td>
<td>7.36</td>
<td>3.92</td>
<td>2.64</td>
<td>13.83</td>
<td>54.37</td>
<td>34.33</td>
<td>12.56</td>
<td>150.01</td>
</tr>
<tr>
<td>LEX-RANK</td>
<td>[35]</td>
<td>25.57</td>
<td>7.26</td>
<td>4.01</td>
<td>2.71</td>
<td>12.51</td>
<td>54.34</td>
<td>34.19</td>
<td>12.38</td>
<td>152.21</td>
</tr>
<tr>
<td>BERT-SUM</td>
<td>[102]</td>
<td>26.22</td>
<td>7.42</td>
<td>4.43</td>
<td>2.90</td>
<td>14.56</td>
<td>55.62</td>
<td>35.57</td>
<td>11.21</td>
<td>149.73</td>
</tr>
<tr>
<td>BART</td>
<td>[92]</td>
<td>41.67</td>
<td>21.05</td>
<td>14.20</td>
<td>10.80</td>
<td>30.20</td>
<td>62.80</td>
<td>44.36</td>
<td>9.97</td>
<td>144.29</td>
</tr>
<tr>
<td>PEGASUS</td>
<td>[195]</td>
<td>37.02</td>
<td>19.68</td>
<td>14.02</td>
<td>10.93</td>
<td>28.44</td>
<td>60.91</td>
<td>41.89</td>
<td>10.53</td>
<td>134.26</td>
</tr>
<tr>
<td>LED</td>
<td>[10]</td>
<td>41.96</td>
<td>21.80†</td>
<td>15.01†</td>
<td>11.58†</td>
<td>31.49†</td>
<td>63.31†</td>
<td>45.06†</td>
<td>9.58</td>
<td>148.03</td>
</tr>
</tbody>
</table>

Table 5.5: Instructions provided to physicians. The scoring scale for summary evaluation is from 1 (worst) to 3 (best).

Table 5.6: Quantitative evaluation of patient after-visit summaries produced by state-of-the-art summarization models. LED shows best performance among all tested abstractive models. It significantly outperforms all other systems for all metrics \((p < 0.05)\), with the exception of BART in terms of R-1, according to a non-parametric Wilcoxon signed rank test.

For alert evaluation, we ask the evaluators to first label missing medical events on the clinical document, and hallucinations on the system summary. The evaluators are then given the alerts produced by our models, and they proceed to judging the correctness of each alert. This allows us to report precision, recall and F1 scores of our error alert models with human judgment.
5.3.3 Summarization Results

**Quantitative.** Table 5.6 provides a quantitative evaluation of after-visit summaries produced by state-of-the-art models. Our aim in this work is not to present new methods, but rather to thoroughly evaluate state-of-the-art models on this challenging task to identify areas for improvement. We observe that BERTSum achieves the highest scores among all extractive models. Further gain is provided by an oracle model developed by Adams et al. [2] that improves R-2 F-score from 7.42% to 13.55% by greedily extracting sentences yielding highest similarity scores with the reference summary. The method gives an upper bound on ROUGE scores obtainable by an extractive model.

We find all abstractive models to perform substantially better than their extractive counterparts. LED has shown best performance among all tested abstractive models, possibly due to its exceptional ability to encode long documents. With regards to evaluation metrics, we include less commonly used R-3 and R-4 F-scores, as they have been shown to correlate better with human judgment than other variants [52, 81]. Our results suggest that generation of patient after-visit summaries is highly abstractive. For this reason, an abstractive model would suit our task best. Extractive summaries are verbose and they may potentially overwhelm patients with unnecessary detail.

**Expert Scoring.** Two medical experts are asked to rate each summary produced by our best abstractive model (LED) along the dimensions of adequacy, faithfulness, readability and ease of revision. Their ratings are averaged for each summary and results are presented in Figure 5.3. All summaries are divided into five bins, their average ratings are 1/1.5/2/2.5/3, respectively. We observe that generating adequate summaries remains a challenge for the abstractive model. Only 5.5% of the summaries obtain a full score (3 points). Per our physicians, the remaining summaries have, to

---

3We provide inter-annotator analysis among physicians in the supplementary materials.
a varying degree, missed important medical events that patients need to know. Our findings suggest that future studies should incorporate expert knowledge in selecting medical events to add to the summary.

Efforts could be made to also improve the readability and understandability of abstractive summaries. We observe that 38.8% of the summaries obtain a full score on readability. A closer analysis reveals that a portion of the summaries contain abbreviated medical terminology or jargon that are familiar to physicians but may be difficult for non-experts. E.g., in “minimal PO intake,” PO is from the Latin “per os” and means “by mouth.” The summaries are also believed to have less hallucination issues when comparing to missing medical events. 72.2% of the summaries obtain 2.5 points or higher. Further, > 75% of the summaries receive an average score of 2.5 or higher on ease-of-revision. The results indicate that, physicians may be guided to revise system-produced summaries to meet the standards of medical practice, as opposed to starting from scratch.

**Expert Revision.** Table 5.8 shows a direct comparison of summaries before and after expert revision (more examples are in the supplementary). Our physicians have revised 43.5 words on average for each summary, corresponding to 47.2% of the summary length. Even though there is still room for improvement, the results are
positive. For 4 out of 18 cases, physicians only minimally revised the summaries, with less than 15% of the words edited. For 3 out of 18 cases, the summaries are nearly rewritten, where 90% of the words are edited. The results suggest that certain noisy clinical documents can cause disastrous summaries. It is crucial for summarizers to degrade gracefully as noise increases.

5.3.4 Error Detection Results

Our detectors are evaluated using both automatic metrics and human judgment. Results are reported in Table 5.7. **H-Alert** is our hallucination detector. It is evaluated on the test set with synthesized hallucinations (Section 5.2.1). Baseline-RAND samples a label for each summary token from a Bernoulli distribution $t_j \sim \text{Bernoulli}(p)$. Here, $p$ is the probability that an average summary token is hallucinated, computed on training data. **MostFreq5** and **MostFreq10** examine the semantic types of events (Table 5.2). If an event type is frequently hallucinated, all of its tokens are labeled as 1. As seen in the table, we find our H-Alert can not only outperform the baselines, but it obtains a high recall score (71.66%).

**M-Alert** is our missing event detector. It predicts source medical events that are missed by the summary. Baseline-RAND samples a label for each source event, $e_i \sim \text{Bernoulli}(q)$, where $q$ is the probability an average event is missed, computed on training data. We find that M-Alert produces better precision scores than all baselines. The best performance is achieved by the model variant +TYPE, which injects event types to the BigBird model to help detection of missing events. We note that identifying key medical events remains a challenging task and graph neural networks may help model inter-event relations.

**Expert P/R/F.** On expert-annotated summaries, we report scores for both detectors. System alerts have been manually verified. The micro-averaged P/R/F scores for H-Alert is 17.24/58.82/26.66, and the scores for M-Alert is 30.65/53.84/39.06.
<table>
<thead>
<tr>
<th>Model</th>
<th>P(%)</th>
<th>R(%)</th>
<th>F1(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline-RAND</td>
<td>6.52</td>
<td>3.22</td>
<td>3.82</td>
</tr>
<tr>
<td>Baseline-MOSTFREQ5</td>
<td>17.80</td>
<td>46.04</td>
<td>21.79</td>
</tr>
<tr>
<td>Baseline-MOSTFREQ10</td>
<td>20.86</td>
<td>76.31</td>
<td>29.19</td>
</tr>
<tr>
<td><strong>H-Alert (Ours)</strong></td>
<td><strong>44.96</strong></td>
<td><strong>71.66</strong></td>
<td><strong>55.25</strong></td>
</tr>
<tr>
<td>Baseline-RAND</td>
<td>3.99</td>
<td>13.56</td>
<td>6.06</td>
</tr>
<tr>
<td>Baseline-MOSTFREQ5</td>
<td>9.74</td>
<td>34.72</td>
<td>13.70</td>
</tr>
<tr>
<td>Baseline-MOSTFREQ10</td>
<td>9.71</td>
<td>49.14</td>
<td>15.04</td>
</tr>
<tr>
<td><strong>M-Alert (Ours)</strong></td>
<td><strong>49.22</strong></td>
<td><strong>43.65</strong></td>
<td><strong>41.71</strong></td>
</tr>
<tr>
<td><strong>M-Alert +POS (Ours)</strong></td>
<td><strong>51.03</strong></td>
<td><strong>45.98</strong></td>
<td><strong>43.80</strong></td>
</tr>
<tr>
<td><strong>M-Alert +TYPE (Ours)</strong></td>
<td><strong>50.69</strong></td>
<td><strong>49.88</strong></td>
<td><strong>45.51</strong></td>
</tr>
</tbody>
</table>

Table 5.7: Automatic evaluation of our hallucination detector (**H-Alert**) and missing event detector (**M-Alert**). Both detectors strongly outperform their baselines.

These results are positive because both detectors are able to attain high recall scores, indicating errors could be effectively flagged and passed on to physicians for further review.

5.4 Discussion

We discuss our findings from interviewing physicians and underline some of the key areas that are indispensable for further progress on this task.

- **Medical jargon.** Owing to time constraints and the literacy of physicians who create the clinical notes, the data we received are of varying quality. It is not uncommon to find jargon or ambiguous information, e.g., “Patient presents w/ < 24 hours abdominal pain nausea and non-bloody V/D,” here, “V/D” refers to “vomit and diarrhea.”

- **Style difference in clinical notes.** The notes could be: 1) procedure-oriented, i.e., they are narratives describing medical procedures performed on the patient, including treatment, medication, care plans, etc., 2) disease-oriented, i.e.,
A System Generated Summary:
You were admitted for dizziness. You had a CT scan of your head which showed some thickening in the sinuses of your sinuses. You were seen by the ear nose and throat doctor who recommended that you take an antibiotic called Unasyn while you are in the hospital. You also had an MRI of your brain which did not show any stroke. You are doing better and can go home today.

After Physician’s Revision:
You were admitted for dizziness. You had a CT scan of your head which showed some thickening in your sinuses and mastoid. This could be suggestive of an infection but your white cells and temperature were normal. You were seen by the ear nose and throat doctor who recommended that you take an antibiotic called Unasyn while you are in the hospital. You also had an MRI of your brain which did not show any stroke. You are doing better and can go home today.

Table 5.8: A direct comparison of summaries before and after physician revision. A post-study interview with physicians reveals that most revisions are related to missing key medical events (colored orange). They also spend substantial efforts explaining medical jargon to patients and fixing hallucinations (colored red).

each of the patient’s diseases is addressed in a separable section, or 3) organ-oriented, i.e., each organ is addressed in a separable section.

- **Improper grounding.** An after-visit summary states “We did test you for the coronavirus which was negative.” However, the “coronavirus test” was nowhere to be found in the source document. Similar grounding issue was identified in 5 out of 18 summaries during expert revision. Sometimes physicians directly include their knowledge about the patients into after-visit summaries without referring to clinical notes, causing a summarizer fine-tuned on such data to also “hallucinate” content.

- **High variance in length.** It would be unwise to truncate clinical notes, despite that most neural models use a fixed maximum length. E.g., a patient who underwent a heart transplant has a high risk of multiple medical comorbidities. It can lead to a large volume of EHR notes. Interestingly, physicians tend to include more content in after-visit summaries if they believe patients have high medical literacy and are able to understand and act upon complex instructions. This indicates that future systems may produce summaries of varying length per patients’ needs.
5.5 Conclusion

We tackle the problem of generation of patient after-visit summaries. We compared state-of-the-art summarization models for this task and introduced a novel alerting mechanism to predict two types of errors, including missing medical events and hallucinations in summaries. Extensive experiments using automatic metrics and expert evaluation show the effectiveness of our proposed approach.

5.6 Ethical Considerations

Data. Data used in this study are obtained from a comprehensive inpatient medical facility. They are electronic dismissal notes created by physicians to record a patient’s hospital stay or a series of treatments performed on a patient. These EHR notes are information-dense and full of technical terms. They need to be rewritten and summarized to generate after-visit summaries. The purpose of using patient medical records is to fine-tune abstractive summarization systems and quantitatively evaluate the truthfulness and adequacy of system summaries. These medical records are not for non-academic uses and intents. All medical records are deidentified by the hospital to protect patient privacy.

Summarization Models. Models for abstractive summarization have a tendency to hallucinate information that is not present in the input documents. This is because abstractive models carry inductive biases rooted in the data they are pretrained on. The data encode prior knowledge of natural language, they may also contain a non-negligible amount of toxic and abusive content. Despite our best efforts to alert clinicians of potential errors, some of them could be almost unnoticeable by non-physicians. We thus caution our users to carefully consider the ethical issues specific to abstractive summarization and natural language generation models.
Patient portal allows discharged patients to access their personalized discharge instructions in electronic health records (EHRs). However, many patients have difficulty understanding or memorizing their discharge instructions [202]. In this paper, we present PaniniQA, a patient-centric interactive question answering system designed to help patients understand their discharge instructions. PaniniQA first identifies important clinical content from patients’ discharge instructions and then formulates patient-specific educational questions. In addition, PaniniQA is also equipped with answer verification functionality to provide timely feedback to correct patients’ misunderstandings. Our comprehensive automatic & human evaluation results demonstrate our PaniniQA is capable of improving patients’ mastery of their medical instructions through effective interactions.

In Section 6.1 we reflect on generating questions in the GPT era, and introduce our approach in Section 6.2. We introduce our annotated dataset in Section 6.3. We report our evaluation results in Section 6.4 and Section 6.5.

6.1 Question Answering in the GPT Era

Large language models (LLMs) such as ChatGPT have led to significant advancements in generative AI [15, 144, 23, 104, 118]. Fine-tuning neural models on specific tasks often yields superior results. Furthermore, LLMs acquire emergent abilities through instruction tuning and reinforcement learning using human feedback [119].
This allows them to generalize to new tasks effectively. Common human-LLM interactions include (a) zero-shot prompting, where users provide a prompt for the LLM to complete, and (b) in-context learning, where users give task examples and ask the LLM to solve a new case, potentially involving a multi-step reasoning process [168]. In this study, we focus on zero-shot prompting to assess the LLM’s ability to comprehend discharge instructions.

LLMs possess vast world knowledge, and their performance on knowledge-intensive tasks correlates with training data and model size [12]. However, it remains unclear whether LLMs have enough domain knowledge to facilitate patient education. For example, GPT-3, with its 175 billion parameters, is trained on general data sources such as Common Crawl, WebText2, Books, and Wikipedia [15]. Yet, the model still generates factually inconsistent errors within their output. Our study presents an initial evaluation of GPT models’ potential in interactive patient education. Following the P.E.E.R. framework of dialogic reading, we employ GPT models to perform the following tasks:

**Question Generation.** We use OpenAI’s GPT-3.5 model (text-davinci-003) to generate informative questions from a discharge instruction. The questions aim at helping patients understand crucial medical events. Our prompt is “Generate N questions to help the patient understand crucial medical events in the above discharge instruction.” Similar to a teacher designing exam questions, we anticipate the GPT model to produce a set of questions all at once rather than incrementally. The questions must collectively cover the salient events identified in the discharge instruction while minimizing redundancy.

**Answer Verification.** Useful feedback is essential for improving patient comprehension of the material. To perform this task, we prompt the GPT model with “As a physician, your goal in the conversation is to help your patient better understand the discharge instructions before they leave the hospital.” Utilizing OpenAI’s API,
Table 6.1: Expert-written question templates are used to generate a question from each binary relation. This method enables us to create targeted questions about salient medical events. By posing questions about one event, we guide patients towards the other as potential answers. The placeholders are to be replaced with medical events detected from discharge instructions.

We also provide the original discharge instruction, interaction history, and current question-answer pair as key-value pairs for the model. We then instruct the model to “verify if the patient’s answer is correct, incorrect, or partially correct, and generate a suitable response to improve the patient’s comprehension of this question.”

We empirically compared two GPT models, text-davinci-003 and gpt-3.5-turbo (ChatGPT), and selected ChatGPT for answer verification as it is optimized for chat and generally produces higher quality responses.

6.2 Extracting Salient Medical Events

In this section, we present our question-answering system that emphasizes identifying salient medical events and their relations. We generate targeted questions using them and apply the same answer verification module described previously.

A typical discharge instruction includes Visit Recap, which recaps a patient’s clinical visit, including symptoms, diagnoses, treatments, and test results. Patients are expected to understand the relationships among these medical events, such as how the treatment ERCP relates to cholangitis as illustrated in Table 6.2 (top). Detailed Instructions include medication and aftercare instructions (bottom). They may be easy to understand but contain trivial details that patients may overlook, potentially
You were found to have an infection of your bile ducts called cholangitis. You had a procedure called an ERCP where a stent was placed to relieve the obstruction...

**Relation:** ERCP (Procedure) – cholangitis (Treatment Goal)

**Question:** What is the goal of treatment ERCP?

---

We made the following changes to your medication regimen: 1. We started you on a new medication called Toprol XL 25mg by mouth twice a day ...

**Event:** twice a day (Medicine Frequency)

**Question:** How often should Toprol XL be taken?

Table 6.2: Question generation (QG) from a medical event (bottom) or a binary relation of events (top).

hindering their self-care at home. We propose automatically extracting key medical events and relations from them (S6.2.1). Given their unique characteristics, we apply two distinct information extraction and question generation strategies for Visit Recap and Detailed Instructions to produce targeted questions (S6.2.2).

### 6.2.1 Event and Relation Identification

Key event and relation identification are conducted on Visit Recap. Event identification is framed as a *sequence labeling* task, where we assign a label to each token of the discharge note, representing its event type. We define 11 event types in this study, detailed in Table 6.3, including symptoms, diseases, complications, tests, test goals/results/implications, procedures, medicines, treatment goals and results.

We fine-tune pre-trained sequence labeling models on our dataset, optimizing the cross-entropy loss of gold standard labels.

Relation identification is framed as a *sequence classification* task. We focus on binary relations consisting of two medical events. We evaluate all pairwise combinations of identified medical events as candidates, provided their event types align with the six event relations defined in Table 6.1. Special tokens are inserted before and after each identified event to indicate both its position and event type.¹ The

¹E.g., the sentence “You were admitted for diverticulitis and treated with antibiotics” was modified as “You were admitted for *diverticulitis* and treated with *antibiotics*,

---

75
sequence, enhanced with special tokens, is fed into a sequence classification model to predict a binary label, where 1 indicates a relation between the two events, and 0 otherwise. We fine-tune pre-trained sequence classification models on our dataset by optimizing the cross-entropy loss for gold-standard labels.

We perform key event identification on Detailed Instructions using a different tool, as they contain medication and aftercare specifics that patients might overlook. We use an existing high-performing medical NER system to extract medical entities. This model was pre-trained on the MACCROBAT dataset \cite{pascal2012} and can identify 84 biomedical entities within clinical narratives. We limit the model to identify 7 entity types: Medicine Dosage, Medicine Frequency, Medicine Duration, Medication Name, Sign & Symptom, Diagnostic Procedure, Upcoming Appointment. Relation identification is not performed on detailed instructions.

6.2.2 Question Generation

Visit Recap. We generate a question from each identified binary relation. Different relation types are mapped to specific questions using templates provided by physicians according to their domain knowledge (see Table 6.1). Using a template-based approach allows us to create questions targeting salient medical events. By asking questions about one event, we guide patients towards the other as potential answers.

Detailed Instructions. We generate a question for each identified medical entity by creating a fill-in-the-blank question, which is then converted into a natural language question using the GPT model. An example is shown in Table 6.2. Although cloze-style questions can serve educational purposes, we want to prevent patients from using string matching to find answers. Instead, natural language questions require

where the special tokens /dsyn\ and /dsyn/ indicates the start and end position of this event, and dsyn reflects the event belongs to the category Disease.

\footnote{https://pypi.org/project/Bio-Epidemiology-NER/}
patients to have a deeper understanding of the discharge note, thus fulfilling our education objective. When selecting medical entities as triggers, we prioritize four categories: *Medicine Dosage*, *Medicine Frequency*, *Medicine Duration*, and *Upcoming Appointment*, as they are informative and better guide patient comprehension. To convert a cloze-style question into a natural question, we provide this prompt to the GPT model: *[Fill-in-the-Blank Sentence] Generate a simple question targeting the blank in the above sentence.*

### 6.3 Data Annotation

We seek to annotate discharge instructions from the MIMIC-III database (v1.4) [66] with key medical events that are important for patients to understand. MIMIC-III is a publicly available repository of de-identified health records of over 40,000 patients collected from the Beth Israel Deaconess Medical Center in Massachusetts. Our aim is to identify text snippets in discharge instructions that correspond to significant medical events, including *symptoms, diseases, test results, and treatments*. We annotate not only individual events but also their relationships. They are organized into a *hierarchy* as outlined in the schema shown in Table 6.3. Consistent with Lehman et al. [89]’s approach, we utilize events and their relationships as *triggers* that prompt the generation of questions.

We recruited five medical experts to create a sizable dataset. They are M.D. students at a reputable U.S. medical school and have a high level of language proficiency. Each expert is given 150 discharge notes to annotate. It is possible to skip some notes due to low text quality. Annotators were also given detailed instructions and examples. We developed a web-based interface to facilitate the annotation process, which has been iteratively improved to meet the needs of this study. Due to budget constraints, we assign one annotator to each discharge note. In total, we completed 458 discharge notes with medical event annotations.
Our annotation consists of two phases. In the first phase, an expert selects text snippets from the discharge instruction corresponding to medical events that the patient needs to understand. Each snippet is assigned a coarse event category, such as a *medical issue*, *laboratory test*, *treatment*. The expert further refines it by assigning a fine-grained event type, resulting in a schema with 11 event types (Table 6.3). In the second phase, the expert identifies relationships between medical events using a set of 6 pre-defined relationships, such as “[Symptom] … caused by [Disease].” We show a distribution of medical events in Figure 6.1.

A key distinction between our work and earlier dataset curation efforts [121, 190, 89] is that the earlier efforts aim to annotate *questions* that physicians would ask during patient hand-off, which may be informal and unanswerable based on the discharge instruction. In contrast, our focus is on annotating *salient medical events* that are essential to patient’s understanding of their medical conditions.
Figure 6.1: Word cloud demonstrating the most frequent medical terminologies and their frequency in our annotations. The sizes of the terminologies refer to their frequency in our dataset. These terminologies are identified from annotated medical events using SciSpacy.

<table>
<thead>
<tr>
<th>Pretrained Model</th>
<th>P(%)</th>
<th>R(%)</th>
<th>F1(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Medical Events</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bert</td>
<td>31.38</td>
<td>44.58</td>
<td>36.83</td>
</tr>
<tr>
<td>BioBert</td>
<td>40.43</td>
<td>51.63</td>
<td>45.35</td>
</tr>
<tr>
<td>PubmedBERT</td>
<td>42.70</td>
<td>50.12</td>
<td>46.11</td>
</tr>
<tr>
<td>ClinicalRoBERTa</td>
<td><strong>44.28</strong></td>
<td><strong>54.03</strong></td>
<td><strong>48.67</strong></td>
</tr>
</tbody>
</table>

| **Event Relations** |      |      |       |
| Bert               | 57.48| 75.31| 65.21 |
| BioBert            | 73.41| 80.37| 76.73 |
| PubmedBERT         | 72.56| 75.31| 73.91 |
| ClinicalRoBERTa   | **74.28**| **82.27**| **78.07**|

Table 6.4: Results of fine-tuning four pretrained models on IE task: medical event extraction (Top) and event-relation identification (Bottom).

We split our annotated data into train / validation / test splits, which contain 338 / 60 / 60 discharge instructions, respectively. For relation identification, we use the event pairs from the human-annotated relations as positive relations and all other medical event pairs of compliant types (e.g., the event pair types in Table 6.3) as negative relations. We collect all negative event pairs as negative cases. Overall, our medical relation dataset contains 2530 / 399 / 332 instances in the train / validation / test set, respectively; 28.7% instances are positive relations.
<table>
<thead>
<tr>
<th>Medical Event</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symptom</td>
<td>50.8</td>
<td>78.2</td>
<td>61.6</td>
</tr>
<tr>
<td>Disease</td>
<td>54.3</td>
<td>73.5</td>
<td>62.5</td>
</tr>
<tr>
<td>Complication</td>
<td>25.0</td>
<td>23.5</td>
<td>24.2</td>
</tr>
<tr>
<td>Test</td>
<td>65.9</td>
<td>82.8</td>
<td>73.4</td>
</tr>
<tr>
<td>Test goal</td>
<td>25.0</td>
<td>25.0</td>
<td>25.0</td>
</tr>
<tr>
<td>Test result</td>
<td>36.3</td>
<td>30.0</td>
<td>32.8</td>
</tr>
<tr>
<td>Test implication</td>
<td>16.6</td>
<td>16.6</td>
<td>16.6</td>
</tr>
<tr>
<td>Procedure</td>
<td>38.5</td>
<td>53.6</td>
<td>44.8</td>
</tr>
<tr>
<td>Medicine</td>
<td>42.3</td>
<td>42.3</td>
<td>42.3</td>
</tr>
<tr>
<td>Treatment goal</td>
<td>16.6</td>
<td>28.5</td>
<td>28.0</td>
</tr>
<tr>
<td>Treatment result</td>
<td>19.0</td>
<td>22.8</td>
<td>20.7</td>
</tr>
<tr>
<td>Overall</td>
<td>44.2</td>
<td>54.0</td>
<td>48.6</td>
</tr>
</tbody>
</table>

Table 6.5: Automatic evaluation results of medical event identification per category with ClinicalRoBERTa model.

<table>
<thead>
<tr>
<th>Medical Event Relation</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Symptom] caused by [Disease]</td>
<td>81.25</td>
<td>79.59</td>
<td>80.41</td>
</tr>
<tr>
<td>[Test] goal: [Test-Goal]</td>
<td>100.0</td>
<td>60.0</td>
<td>75.0</td>
</tr>
<tr>
<td>[Test] result: [Test-Result]</td>
<td>61.54</td>
<td>92.31</td>
<td>73.85</td>
</tr>
<tr>
<td>[Test] implication: [Test-Implication]</td>
<td>57.14</td>
<td>66.67</td>
<td>61.54</td>
</tr>
<tr>
<td>[Treatment] goal: [Treatment-Goal]</td>
<td>81.82</td>
<td>81.82</td>
<td>81.82</td>
</tr>
<tr>
<td>[Treatment result: [Treatment-Result]</td>
<td>70.59</td>
<td>85.71</td>
<td>77.42</td>
</tr>
<tr>
<td>Overall</td>
<td>74.28</td>
<td>82.27</td>
<td>78.07</td>
</tr>
</tbody>
</table>

Table 6.6: Event-Relation detection per category with ClinicalRoBERTa model.

6.4 Evaluating Information Extraction

To improve LLMs’ ability to generate educationally effective questions for patient education, we designed an Information Extraction (IE) module (medical event/relation identification) to guide question generation. We report automatic evaluation results for different IE methods in this section.

6.4.1 IE Evaluation Settings

We fine-tune four pre-trained language models on our annotated dataset for key medical event and relation identification in Section 6.3. These models are obtained from HuggingFace: 1) BERT-large [26]; 2) BioBERT [88]; 3) PubmedBERT [54]; 4) ClinicalRoBERTa [93]; All four pre-trained models have the same scale of parameters.

---

3I.e. No relationship exist between the event pair, in addition, the types of the two events are restricted by Table 6.1
(345 million). The later three language models were pre-trained on different biomedical or clinical corpora, thus are better transferable to our patient education task [161, 185, 186]. The models are trained on a single RTX 6000 GPU with 24G memory. The average training time for the relation identification model is around 20 minutes.\(^4\) For evaluation metrics, we report the model’s Micro-average precision, recall, and F-1 score.

### 6.4.2 IE Evaluation Results

The performance of four evaluated models is in Table 6.4. The results suggest: The models pre-trained with biomedical or clinical corpus show better performance than the naive Bert model. For both tasks, ClinicalRoberta achieves the best performance, so we report only this model’s performance in following category-wise performance analysis.

We further report more fine-grained results of the medical event extraction per category in Table 6.5, and the Symptom, Disease, Test, Procedure and Medicine categories generally achieve better performance, as we suspect it is due to a more abundant training data. Table 6.6 shows the fine-grained performance of event-relation identification per category. The F-1 scores of most relations are around 80\%, implying fair performance. The relation Test goal achieves 100\% in precision because our test set contains eight Test goal instances.

To explore the generalization ability of the model, we compare the model’s performance on seen and unseen medical events during training. Specifically, seen events are events that appear in the training set, while unseen events are not. We observed that 15.21\% of the test instances are unseen medical events. For the medical-event extraction task, the F1 score of seen events is 49.36\%, and the F1 score of unseen

\(^4\)Due to data sparsity, when training both the medical event and relation identification models, we first explore the optimal hyper-parameter set using the validation set. We then combine the validation set into the train set to train our models.
events is 44.82%. For the event-relation identification task, if both event in the event pair are seen events, the model achieves 78.72% in, otherwise the model achieves 74.50% in F1. As a consequence, the model only shows slight drop in performance when encountering unseen medical events during training.

6.5 Evaluating Patient Education

In order to comprehensively evaluate patient education outcome, we conducted human evaluations from both the patient’s and the physician’s perspectives, as well as a GPT-4 powered automatic evaluation. These evaluations focus on two main aspects: 1) The generated question’s quality of different models (GPT, GPT+IE, and human ground-truth); 2) The preference of different designs of the interaction experience (None of support, Question only, and Question and Answer).

6.5.1 Human Evaluation Settings

The goal of physician evaluation is to have human domain experts evaluate whether these machine-generated questions are comparable to the human-crafted
questions or not. To do so, we recruited 3 medical practitioners\(^5\) and their tasks are to read the discharge instructions, and provide qualitative feedback on if these machine-generated questions have educational effective to the patients; if not, how should they be improved?

The goal of **patient evaluation** is to have the general public users interact with and provide ratings on the different combinations of the question-generation models and the interaction designs. We also designed a post-experiment evaluation task (i.e., Cloze Test) to quantitatively measure their understanding outcome. We recruit 30 human evaluators to participate in our patient education experiment. All the evaluators have bachelor’s degrees but do not have any medical education background.

In our study, we have the following three options for the user interaction experience design:

1. **Condition None**: The evaluator only sees the discharge instruction, no question-answer interaction. This is today’s baseline.

2. **Condition Q**: The evaluator reads the discharge instruction, and interact with the chatbot, which can only ask questions but do not to provide feedback to users’ answers.

3. **Condition QA**: The evaluator reads the discharge instruction, and interact with the chatbot, which can ask questions and provide answer feedback to the user.

The questions asked by the chatbots can come from following three sources:

1. **Human**: Expert-written questions based on discharge instructions. We ask an MD student to read each discharge instruction and write down all questions she would ask a patient about this discharge instruction for patient-education purposes.

\(^5\)Two licensed physicians and one medical student with hospital internship experience
2. **GPT**: We utilize GPT-3 model to generate a series of questions (at least four) directly from the discharge instruction. Specifically, we use the following prompt:

\[
[\text{Discharge Instruction}] \text{ Generate at least four questions to help the patient understand crucial medical events in the above discharge instruction.}
\]

3. **GPT+IE**: Our question generation model enhances by the information extraction technique described in Sec 6.2.

The average number of questions from approach Human / GPT / GPT+IE are 7.5 / 6.17 / 6.1. When combining the variety of interaction designs and question-generation methods, there are five different conditions: 1) None; 2) Q (Human); 3) QA (GPT); 4) QA (GPT+IE); 5) QA (Human). We perform a within-subject experiment setup, where each of the 30 human evaluators should experience all five conditions using different discharge instructions. In total, we have 150 data points (30 per each condition). The order of the five conditions are shuffled so that each condition appears six times at each of the five orders.

### 6.5.2 Patient Evaluation Measurements

We use two measurements to evaluate patient’s educational outcome and preference.

1) **Cloze Test**: We recruited an MD student to identify 5-7 important medical events that she thinks the patient should be aware of, and replace them with blanks. We use these cloze tests as a post-study evaluation to ask each participant to try their best to fill in the blanks using their memory. The more blanks they fill in correctly, the better the patient’s education outcome is. We report the participant’s accuracy rate as the primary evaluation outcome.

---

\(^6\)We have tried a collection of prompts for the similar purpose, and do not observe significant differences in the quality of generated questions. We used the chosen prompt as it is naive to understand and leads to more succinct questions. Specifically, we instruct GPT-3 to generate at least four questions to benchmark against the least number of questions from the human annotator.
2) **Preference Ranking:** We ask evaluators to rank their experience using the following four questionnaire items (Evaluators are allowed to rank two conditions as tied):

- **Coverage:** Does the conversation cover the cloze test in the evaluation?
- **Appropriateness:** Are the questions properly raised, and appropriate for patient education?
- **Education Outcome:** How do you think the learning experience improves your understanding of discharge instructions?
- **Overall:** How do you like the general learning experience considering the above aspects?

We report the Mean Reciprocal Rank (MRR) [130] of each model’s final ranking. Generally, a higher MRR value implies the evaluators have more preference over an approach.

---

You are a physician who wants to evaluate how helpful an AI model is for educating patients. The model asks the patient questions, then verifies the patient’s answers, in order to help patients memorize their discharge instructions.

Four evaluation aspects for AI model’s question quality includes:

- **Coverage:** Does the conversation cover the cloze test in the evaluation?
- **Question Appropriateness:** Are the answers to the questions contained in the discharge instruction?
- **Education Outcome:** Do you think the chatbot helps patients understand their discharge instructions?
- **Overall:** How do you like the general experience with the chatbot considering the above aspects?

Two evaluation aspects of the AI model’s feedback includes:

- **Correctness:** Are the responses from the AI model factually correct?
- **Education Potential:** Do the AI model’s responses provide helpful information for educating patients?

5-point Likert scale:
1: very low rating
2: low rating
3: neutral or medium rating
4: higher rating
5: very highly rating

The patient’s discharge instructions: *The Patient’s Discharge Instruction*

The conversation between the patient and the AI model: *The Conversation History*

Give the 5-point Likert scale of the AI model’s question quality (four aspects) and answer feedback (two aspects) one by one. Return the scores as dictionary objects, adhering to the following structure: "Coverage": ..., "Question Appropriateness": ..., Please provide your response solely in the dictionary format without including any additional text.

**Table 6.7:** Prompt presented to GPT-4 for evaluating the quality of generated questions and answer verification. GPT-4 is expected to output a score on each perspective directly.
6.5.3 GPT-4’s Automatic Evaluation Settings

Following recent practice of applying large language models in evaluating dialogue tasks [101], we utilize GPT-4 as the evaluation model to automatically measure the quality of AI generated questions and feedback. Similar to patient evaluation in Section 6.5.2, we evaluate the quality of generated questions from the four perspective (i.e. Coverage, Question Appropriateness, Education Outcome and Overall). Additionally, we also evaluate the quality of AI models’ feedback from two perspectives, i.e. Correctness and Education Potential. Our prompt to the evaluation model is shown in Table 6.7. We collect evaluation model’s responses and report the average score of each perspective.

6.5.4 Synthesized Dataset for Evaluation

Directly presenting real health records to LLMs or participants can lead to data privacy violation.7 Thus, we created 30 synthesized discharge instructions for our human evaluation study. We randomly sampled 30 hospital course notes (a part of EHR data) from the MIMIC-III database, and converted them into synthetic discharge instructions following a neural abstractive summarization method proposed by [16]. Our physician collaborators reviewed these synthesized discharge instructions to ensure content validity and anonymity.

We then apply the various ways (human, GPT, GPT+IE) to created question-answer pairs for these anonymized synthesized data. We demonstrate some sampled discharge instructions and corresponding generated questions in Table 6.8.

6.5.5 Physician Evaluation Results

We interview three physician participants with following questions: 1) Do you think the questions are effective for patients to understand the important info in the

7https://physionet.org/content/mimiciii/view-dua/1.4/
Figure 6.3: Patient evaluation results, including Cloze Test accuracy and evaluator rankings’ MRR scores across four categories (higher is better). The methods are represented with color-coding: None-blue, Q (Human)-orange, QA (GPT)-yellow, QA (GPT+IE)-beige, QA (Human)-green.

discharge instruction? If not, what questions would you ask? 2) How do you like the questions generated from GPT and GPT+IE?

Physician participants all believe that GPT-generated questions tend to target content that patients do not need to be aware of (e.g., asking why heart attack could cause chest pain is a medical-domain-specific knowledge not suitable for patient’s education). Sometimes the answers to the GPT-generated questions do not even exist in the discharge instruction. Take example 1 in Table 6.8, the question asks what the patient should expect in their follow-up visits, but this information is not mentioned in the discharge instruction. These qualitative findings may explain why GPT-generated questions’ are rated by patient participants as low accuracy score in the Cloze Test metric, as well as ranked lower in Coverage, Appropriateness, and Education Outcome in the Section 6.5.6.

Worth noting, in some cases where the answers are not in the discharge instructions, physician participants actually believe those questions could be useful for patient education. In example 2 in Table 6.8, although the discharge instruction does not contain information on how to maintain the stent, physicians still think it is a question they would ask their patients, as it would motivate patients to have better self-managed recovery activities.
For questions generated by GPT+IE, most questions were perceived by the physicians as appropriate (e.g. example 3). However, the GPT-IE may still generate improper questions due to errors in the medical event-relation identification. As shown in example 4, the information extraction model identifies the symptom “swelling in your throat” as a disease, which leads to improper questions.

Physician participants also suggested that some GPT-IE-generated questions lack language fluency. As shown in example 5, the generated question seems redundant and can be better rephrased as “How long do you need to take Prednisone?”

<table>
<thead>
<tr>
<th>Example</th>
<th>Question</th>
<th>Analyzed</th>
<th>Corrected</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Your symptoms improved and you were discharged to home with close follow-up with your primary care physician and an allergist ... (GPT) Q: When is your follow-up appointment with your primary care physician and allergist and what should you expect during these visits?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>We also found that you have a condition called tracheobronchomalacia, which is a blockage of your airways. You had a stent placed in your airway to help keep it open ... (GPT) Q: How should you maintain the stent in your airway?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>You were admitted to the hospital with fevers. You were found to have pneumonia, and you were treated with antibiotics ... (GPT+IE) Q1: What is the cause of your symptom fevers? Q2: What treatment is applied to disease pneumonia?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>You were admitted to the hospital with swelling in your throat ... You were treated with steroids, benadryl, famotidine and epinephrine ... (GPT+IE) Q: What treatment is applied to disease swelling in your throat?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>The following changes have been made to your medications: START Prednisone 40mg daily for 5 days ... (GPT+IE) Q: What is the recommended duration for taking Prednisone at 40mg daily?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.8: Examples of the synthetic discharge instructions and generated questions

### 6.5.6 Patient Evaluation Results

We summarize the patient evaluation results in Figure 6.3. From the (a) Cloze Test chart, we observe that having a chatbot interact with patient participants (regardless of only with Q or with both QA) can indeed improve their performance over the baseline condition None, which suggests our proposed interactive question-answering design is a promising for patient education. In terms of whether having an answer feedback is helpful or not, the 92.7% accuracy of QA (Human) significantly outperforms the 80.6% accuracy performance of Q(Human), this implies the importance of validating patients’ answers and presenting feedback, thus we decided
to always including an answer feedback when conducting further comparison analysis regarding the $GPT$ v.s. $GPT+IE$ question generation algorithms. The result shows that $QA (GPT+IE)$ $88.3\%$ achieves higher accuracy than $QA (GPT) 74.1\%$. This demonstrates the improvement by applying enhancements to LLMs for patient education purposes.

The result related to Evaluator Ranking shows (plots (b, c, d, e) in Figure 6.3): 1) Considering the Overall ranking of three sets of questions using QA interactive approach, $Human$ questions performs better than AI generated questions. This suggests machine-generated questions are still not comparable to human ones. 2) Comparing the three interactive approaches, we observe $QA (Human) > Q(Human) > None$, which is in line with the findings of Cloze Test. 3) In terms of Appropriateness, and Education outcome, $GPT$ achieves the lowest ranking. According to our observation, many $GPT$-generated questions ask the evaluators about content not existing in the discharge instruction. As a result, evaluators think the questions are inappropriate and do not help patient education. 4) $QA(Human)$ has higher ranking in Coverage than $Q(Human)$, despite they use the same questions. This suggests much benefit is provided to patients through the answer feedback interaction.
6.5.7 GTP-4’s Automatic Evaluation Results

In terms of question quality (as shown in Figure 6.4), we observe GPT-4’s evaluation scores generally follow the same pattern of patient evaluation results, where questions from \( Q(Enhanced) \) are deemed better than \( Q(Direct) \). In addition, we observe the scores of all approaches are close or higher than 4, this implies GPT-4 judges the generated questions are of good quality in four perspectives. In terms of answer verification, as all interactive conditions all share the same verification method, we only present the average Correctness and Education Potential score. Specifically, GPT-4 gives 4.14 on Correctness and 4.01 on Education Potential. Both scores are above four, indicating GPT-4 judges feedback from our AI agents’ feedback as high quality.

6.5.8 Heuristic Evaluation of Conversation Log

We further conducted a heuristic evaluation to explore the deficiency of AI-generated responses and potential improvements. Specifically, we asked an MD student to evaluate the conversation log data of all patient participants.\(^8\) Overall, we collect 192 responses from 30 conversations between the participants mimicking patients and the AI model.

We ask our MD-background human evaluator to grade each of the AI model’s answer feedback, we apply the same evaluation metric, i.e., correctness and education potential as introduced in Section 6.5.3. We apply binary coding, i.e. evaluator judge response as positive or negative.

The positive rate for Correctness is 86.4\%, and the positive rate for Education Potential is 74.1\%. This suggests that most responses are factually correct and provide helpful information to patients.

\(^{8}\)The conversation logs are re-used from the patient evaluation described in section 6.5.6.
Table 6.9: Examples of chatbot’s answer feedback.

Table 6.9 shows some examples of the answer feedback from the chatbot, and we have following design suggestions for future research to improve the quality of the answer feedback: 1) Most responses are helpful for patients in reviewing their discharge instructions (example 1). But, some responses are factually incorrect and may confuse patients. The AI model may state that the patient’s answer is incorrect or partially correct (example 3), while the patient’s response is actually completely correct. 2) While the responses are generally helpful, they still have a deficiency in providing sufficient and attentive responses in educating patients like a human physician. As shown in example 4, a physician will provide more information about the distinctions between the two medications, including the specific diseases for which they are prescribed.
6.6 Conclusion

In this study, we present PaniniQA, a patient-centric interactive question answering system designed to help patients understand and memorize their discharge instructions. PaniniQA generates educational questions from discharge instructions after identifying salient medical events and event relations. LLMs with prompting is promising for question-answer generation, but sometimes hallucinating. Extensive evaluations highlight the importance of providing answer feedback.

6.7 Ethical Considerations

We would like to note a few ethical considerations of this work:

**Biases.** Large language models trained on vast amounts of text data can pick up biases present in data. For example, when generating patient education-oriented questions, they may prefer certain questions related to Aspirin or even associate certain health conditions with specific groups of people. They may also perpetuate misinformation and provide incorrect information. In addition, people who participated in our evaluation have different levels of language proficiency and medical background. These biases may be mitigated by enhancing model alignment with each individual’s background and health literacy level.

**Broader Impacts.** We have performed a preliminary study to educate patients on discharge instructions using interactive question answering. Although we evaluated our system using the MIMIC III dataset, which represents an intensive care unit (ICU) setting, the system should be generalizable to other settings, including perioperative care (from preparation before the surgery to recovery after the surgery), cancer treatment, and chronic condition management. Our system may help patients receive customized information that is tailored to their individual needs and preferences.

**Social Influence.** Our system has two pillars. First, it is grounded in discharge notes, where we identify important medical events and their relationships that pa-
tients should know. Second, it serves an education purpose. For that, we explore the P.E.E.R sequence to prompt the patient, evaluate, extend and ask them to repeat the answer to reinforce their understanding. Additionally, social influence strategies such as small talk, empathy, persuasion can be explored in the future to shape, reinforce, or change a patient’s behavior and promote engagement.

**Privacy Implications.** LLMs can present privacy concerns in patient education when health records are used, potentially violating the HIPPA regulations. However, in this study, we handle data usage with great care. We conduct all experiments on open-sourced real patient data and present an approach to synthetic patient discharge notes. Each synthetic discharge note used in this study has been reviewed by physicians to ensure their validity. We strictly limit our API usage to synthetic data.
CHAPTER 7
CONCLUSIONS & FUTURE WORKS

In this thesis, we delved into applying generative language models for personalized information understanding. By analyzing the users’ profile and interactions, we are able to better capture their interests and intention, thus provide more personalized information service, making understanding complex information a piece of cake. Specifically, we apply our approaches to the patient education domain, i.e. help patient better understand their health situation by providing them with personalized medical instructions and feeding information through interactive question answering.

In our future works, we will continue along the exploration of utilizing chatbots for personalized information service, with specific focuses on following subdomains:

- **Personalized News Information Acquisition** Presently, news aggregator applications rely on users’ reading histories as integral features to improve the recommendation of news content aligned with their interests. Our future research will expand on personalized information acquisition in two key directions. Firstly, instead of providing users with entire documents, we aim to deliver more succinct and accurate information through interactive dialogues. Secondly, we intend to refine user profiling by analyzing conversational interactions with users.

- **Personalized Literature and Movie Discussion** A deeper understanding of literature and movies often arises through in-depth discussions with others, involving knowledge sharing and the exchange of opinions. Currently, users turn
to literature or movie review platforms such as *Rotten Tomatoes* and *Douban* to glean insights from reviews. Our research will explore the integration of chatbots into these discussions. These chatbots are designed to comprehend user comments, identify similar opinions from other users, synthesize these viewpoints, and facilitate coherent conversations with users. Unlike traditional reviews, this approach offers real-time feedback on literature or movies through dynamic discussions. Furthermore, these discussions can closely align with the users’ perspectives, making them more engaging and relevant.
## APPENDIX

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVS</td>
<td>After-visit summary</td>
</tr>
<tr>
<td>EHR</td>
<td>Electronic health record</td>
</tr>
<tr>
<td>LM</td>
<td>Language model</td>
</tr>
<tr>
<td>LLM</td>
<td>Large language model</td>
</tr>
<tr>
<td>WoW</td>
<td>Wizard of Wikipedia</td>
</tr>
<tr>
<td>LSTM</td>
<td>Long short-term memory</td>
</tr>
<tr>
<td>DPR</td>
<td>Dense Passage Retriever</td>
</tr>
<tr>
<td>LED</td>
<td>Longformer-Encoder-Decoder</td>
</tr>
<tr>
<td>RL</td>
<td>Reinforcement learning</td>
</tr>
<tr>
<td>CUI</td>
<td>Concept unique identifier</td>
</tr>
<tr>
<td>ICU</td>
<td>Intensive care unit</td>
</tr>
<tr>
<td>P.E.E.R</td>
<td>Prompt, Evaluate, Expand, Repeat</td>
</tr>
<tr>
<td>IE</td>
<td>Information extraction</td>
</tr>
<tr>
<td>HIPAA</td>
<td>Health insurance portability and accountability act</td>
</tr>
</tbody>
</table>

Table 8.1: Abbreviation list
BIBLIOGRAPHY


[78] Krapivin, Mikalai, Autaeu, Aliaksandr, and Marchese, Maurizio. Large dataset for keyphrases extraction.


