PACE-RAG: Patient-Aware Contextual and Evidence-Constrained RAG for Clinical Drug Recommendation

Anonymous ACL submission

Abstract

Drug recommendation requires a deep understanding of individual patient context, especially for complex conditions like Parkinson's disease. While LLMs possess broad medical knowledge, they fail to capture the subtle nuances of actual prescribing patterns. Existing RAG methods also struggle with these complexities because guideline-based retrieval remains too generic and similar-patient retrieval often replicates majority patterns without accounting for the unique clinical nuances of individual patients. To bridge this gap, we propose **PACE-RAG** (**P**atient-**A**ware **C**ontextual and **E**vidence-Constrained **RAG**). Rather than directly copying frequent medications from retrieved patients, PACE-RAG personalizes recommendations by first extracting patient-specific clinical features, retrieving cases around these features, and then refining the final prescription using the patient's current symptoms, active medication history, and focus-specific prescribing tendencies. By analyzing treatment patterns tailored to specific clinical features, PACE-RAG generates patient-specific medication recommendations along with an explainable clinical summary. Evaluated on a Parkinson's cohort and the MIMIC-IV benchmark using Llama-3.1-8B and Qwen3-8B, PACE-RAG achieved state-of-the-art performance, reaching F1 scores of 80.84% and 47.22%, respectively. These results suggest that PACE-RAG is a robust and clinically grounded framework for personalized decision support. Our code is available at: `https://anonymous.4open.science/r/PACE-RAG-227C/`.

1 Introduction

Drug recommendation represents a complex clinical challenge that necessitates a highly personalized strategy tailored to the unique clinical presentation of each patient (Twala, 2025). Although Large Language Models (LLMs) possess extensive medical knowledge and achieve expert-level performance on standardized benchmarks (Singhal et al., 2023a; Nori et al., 2023), they frequently struggle with the nuanced, context-aware decision-making required in complex clinical practice (Deik, 2023). While physicians derive clinical judgments by synthesizing a patient's unique clinical condition with their own accumulated clinical experience, LLMs struggle to capture this real-world experiential knowledge. Therefore, it is essential to develop methodologies that effectively incorporate practical clinical insights.

	Medical Knowledge	Evidence-based Medicine	Personalized Medicine
Guideline RAG	✓	✗	✗
TreatRAG	✓	✓	✗
PACE-RAG (ours)	✓	✓	✓

Figure 1: **Comparison of medical RAG approaches.** Guideline RAG provides general medical knowledge, whereas TreatRAG retrieves similar patient cases but remains limited in personalization. **PACE-RAG** achieves personalized medicine by integrating a verification process with patient-similarity retrieval.

To address this, Retrieval-Augmented Generation (RAG) frameworks utilize external knowledge retrieved from clinical guidelines or medical textbooks (Zhao et al., 2025; Wang et al., 2024). However, these methods often suffer from semantic fragmentation during document chunking and fail to capture the intuitive judgment characteristic of expert physicians. Specifically, as illustrated in Figure 1, *Guideline RAG*, which retrieves standardized clinical documents containing standardized disease management guidelines, generates overly generalized reasoning that fails to account for the heterogeneous clinical trajectories of individual patients (Armstrong and Okun, 2020). To overcome the rigidity of general guidelines, recent research retrieved similar historical patient cases (Gao et al.,

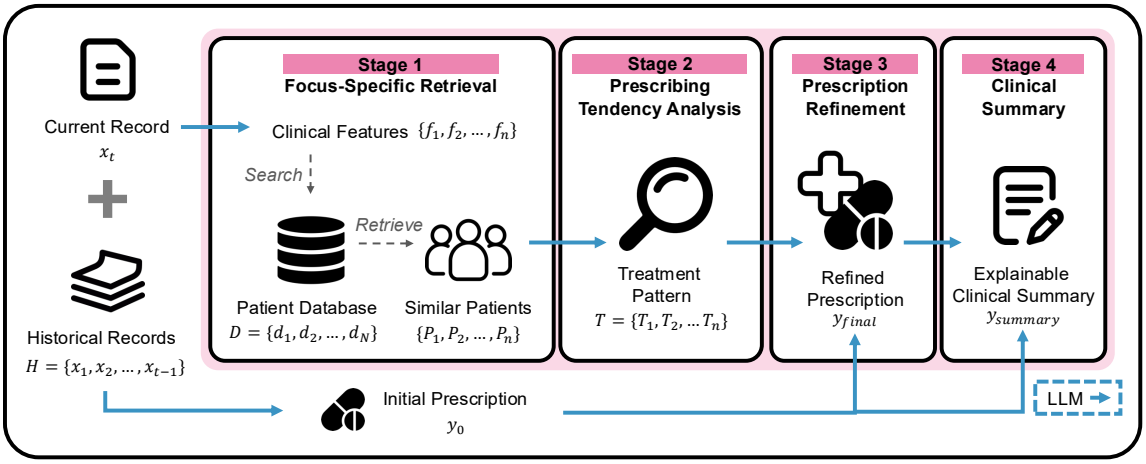

Figure 2: **Overview of the PACE-RAG framework architecture.** The pipeline consists of four stages: (1) Focus-Specific Retrieval to identify similar patients from x_t; (2) Prescribing Tendency Analysis to extract medication patterns from retrieved cases; (3) Prescribing Refinement to verify the initial prescription against identified tendencies; and (4) Clinical Summary to synthesize the final treatment rationale. The LLM performs each generation step, from focus extraction to final summary generation.

2024; Liu et al., 2025; Chen et al., 2026). While retrieving similar patient treatment cases (*Similar Patient RAG*) such as TreatRAG (Liu et al., 2025) offers more specific analysis than Guideline RAG, it remains susceptible to 'majority bias,' in which the model simply replicates the most frequent prescription patterns observed across the retrieved cohort rather than identifying the clinical rationale for each case. The major limitation of this approach is that patient-level similarity alone does not guarantee personalized prescribing.

To address these limitations, we propose **PACE-RAG** (**P**atient-**A**ware **C**ontextual and **E**vidence-Constrained RAG), a novel framework designed to capture each patient's context and leverage evidence-based retrieval to model the implicit prescribing tendencies of clinical experts. As shown in Fig. 2, our framework employs a four-stage reasoning process to identify patient-specific medication recommendations for each patient based on their specific clinical presentation.

First, we employ *Focus-Specific Retrieval*, which retrieves similar patients using symptom-level clinical features rather than the full clinical note, thereby focusing retrieval on patient-specific clinical signals. Second, we implement *Prescribing Tendency Analysis*. The model extracts treatment patterns from the retrieved cohort, effectively modeling how physicians typically respond to specific symptom clusters. Third, we introduce *Evidence-Constrained Prescription Refinement*, which verifies the initial medication plan against the prescribing tendencies derived from similar patient cases. Finally, the *Explainable Clinical Summary* synthesizes the decision trajectory into an interpretable report to ensure clinical trust and utility.

We evaluate our framework using a real-world Parkinson's Disease dataset. Our results exhibit superior efficacy compared to alternative retrieval-based methods and verification frameworks. To demonstrate generalizability, we extended our evaluation to the MIMIC-IV benchmark (Johnson et al., 2023). These results suggest that our method handles the complexities of multi-symptom clinical records. Crucially, our framework built on a small open-source model surpasses large-scale models such as Qwen2.5-72B, demonstrating a privacy-preserving and cost-effective alternative without relying on large closed-source models.

In summary, our contributions are as follows:

- We propose **PACE-RAG**, which overcomes the over-generalization of guideline-based retrieval and the majority-bias of similar-patient retrieval by identifying symptom-specific rationales within individual patient contexts.
- We introduce a four-stage methodology that extracts focus-specific clinical signals to perform keyword-centric retrieval of similar cases and analyzes prescribing tendencies to generate context-aware medication plans.
- We demonstrate that our framework generalizes beyond Parkinson's Disease to the MIMIC-IV dataset, demonstrating its robustness in handling heterogeneous, multi-symptom clinical contexts.

2 Related Work

2.1 LLMs in Clinical Decision Support

LLMs have demonstrated remarkable capabilities in the medical domain, achieving expert-level performance on benchmarks such as the USMLE and MedQA (Singhal et al., 2023b; Nori et al., 2023). Beyond these standardized benchmarks, recent research has expanded the application of LLMs to complex clinical workflows, including differential diagnosis and treatment planning (Kanjee et al., 2023; Thirunavukarasu et al., 2023; Chen et al., 2023). However, despite their medical reasoning capabilities, LLMs struggle with *personalized* and *context-dependent* care, particularly in neurodegenerative disorders like Parkinson's disease. For instance, while LLMs can accurately provide general knowledge regarding the symptoms of Parkinson's disease—specifically movement disorders—they often fail to reflect the granular, patient-specific context required for real-world clinical prescriptions (Deik, 2023). Our work addresses this limitation with a context-aware system that integrates an individual patient's treatment trajectory with prescription patterns from similar patient cases.

2.2 Retrieval Framework in Clinical Settings

To mitigate domain-specific knowledge gaps, RAG has been widely adopted in the medical domain (Gao et al., 2024; Xiong et al., 2024; Asai et al., 2024; Zhao et al., 2025; Zakka et al., 2023; Liu et al., 2025). Existing approaches primarily utilize either explicit medical knowledge (Guideline-based) or historical patient cases (Case-based). Guideline-based approaches prioritize clinical standardization by retrieving information from medical textbooks or clinical practice guidelines (Zhao et al., 2025; Zakka et al., 2023). While effective at providing verified medical facts, the reliance on static knowledge-based retrieval cannot fully capture the heterogeneous clinical profiles and diverse situational nuances of individual patients in real-world practice. Evidence-based retrieval instead utilizes historical EHR data to provide more tailored support, but existing models often rely on global semantic similarity that overlooks localized symptom signals (Liu et al., 2025). Our approach introduces a novel framework that extracts primary clinical symptoms to drive targeted, keyword-based retrieval, enabling retrieval around patient-specific clinical signals.

2.3 Verification of Clinical Reasoning

While LLMs utilize RAG to minimize hallucinations, they are prone to an over-reliance on retrieved contexts. Consequently, a verification stage is essential to validate the model's response and discern only the most meaningful insights from the retrieved content. To address this, existing approaches prioritize internal consistency or introspection to refine model outputs (Manakul et al., 2023; Gandhi et al., 2024; Li et al., 2025). Within the clinical domain, AMIE (Tu et al., 2025) and MedAgents (Tang et al., 2024) have been further adapted to emulate the reasoning processes of medical experts. For instance, MedReflect (Huang et al., 2026) minimizes errors by generating and training on question-answer pairs designed to validate reflective reasoning paths, thereby refining the model's internal logic. However, these existing verification frameworks struggle to incorporate the unique clinical contexts of individual patients. To address this, we introduce an Evidence-Constrained Prescription Refinement stage which evaluates the initial prescription based on the prescribing tendencies derived from similar patient cases, preventing over-dependence on raw retrieved content by validating the treatment plan against essential clinical information.

3 Methodology: PACE-RAG

We consider an outpatient prescription scenario, where physicians make medication decisions based on the patient's current symptoms and recent visit history. At each visit, the decision is to determine whether to maintain existing medications, add new drugs for newly emerged or worsening symptoms, or discontinue inappropriate medications. We denote the current clinical note as x_t and define the patient's recent historical visit records as $H_t = \{x_{t-m}, \ldots, x_{t-1}\}$, where $m \leq 3$ indicates that we use at most the three most recent visits before the current visit. Given H_t and x_t, our goal is to generate a prescription set $Y_t = \{y_1, \ldots, y_k\}$, where each y_i is an individual medication and $k \geq 1$.

Initial prescription generation. Before the execution of the multi-stage reasoning pipeline, the LLM, denoted as ϕ, produces an initial candidate prescription y_0 (prompt provided in Appendix E.1). This prescription is generated directly from the patient's longitudinal history H and the current

clinical presentation x_t, as follows:

$$y_0 = \phi(H, x_t) \tag{1}$$

This initial output acts as an initial prior knowledge. To refine this initial prior into a clinically precise prescription, the clinical plan is generated through a four-stage reasoning pipeline: Focus-Specific Retrieval (§3.1), Prescribing Tendency Analysis (§3.2), Prescription Refinement (§3.3), and Clinical Summary (§3.4). Figure 2 illustrates the overall workflow.

3.1 Stage 1: Focus-Specific Retrieval

Focus query extractor. To retrieve patients with similar symptoms, the Focus Query Extractor isolates the most salient features from the current clinical note x_t by identifying n distinct clinical queries. This process is defined as:

$$\{f_1, f_2, ..., f_n\} = \phi(x_t) \tag{2}$$

where each f_i represents a localized clinical feature within x_t. To prevent hallucinations, each f_i is strictly constrained to literal substrings found within x_t, ensuring the extractor captures only significant clinical indicators while filtering out overly broad or non-specific terms (details in Appendix B.2 and E.2).

Retrieving similar patients. We perform a retrieval across a historical patient database $D = \{d_1, d_2, \ldots, d_N\}$. For each clinical feature f_i, we retrieve the top-k most relevant patient cases among those that exceed a similarity threshold τ.

$$P_i = S(f_i, D) \tag{3}$$

where $S(\cdot)$ denotes a selection function that identifies the nearest neighbors within the latent embedding space (details in Appendix B.3). This produces one retrieved patient cluster P_i for each clinical feature f_i. This yields n focus-specific patient clusters, $P = \{P_1, P_2, \ldots, P_n\}$. This makes retrieval patient-specific because PACE-RAG retrieves evidence around the clinical features that are most relevant to the current patient's medication decision, rather than globally similar records based on the entire note.

3.2 Stage 2: Prescribing Tendency Analysis

Instead of analyzing the entire prescription history of retrieved patients, this stage conducts a targeted prescribing tendency analysis to isolate the direct impact of the clinical feature f_i. This approach prevents over-generalized recommendations of high-frequency drugs that lack specific clinical evidence for the clinical feature. For each clinical feature f_i, the model analyzes the corresponding retrieved cluster P_i to extract T_i, a set of medications specific to the feature along with their underlying rationales. To ensure high clinical fidelity and prevent hallucinations, the tendency analyzer ϕ enforces three rigorous criteria:

- **Isolation of focus-specific drugs:** The model extracts only medications directly relevant to f_i, filtering out general or routine drugs to isolate feature-specific interventions.
- **Evidence-based rationale:** Each added medication must be supported by direct evidence relating to the clinical feature f_i.
- **Null on uncertainty:** In the absence of explicit evidence attributing a drug to f_i, the output T_i returns to an empty set ($\emptyset$).

The total tendency set for the patient is defined as $T = \{T_1, T_2, \ldots, T_n\}$, where each T_i is derived from the corresponding clinical feature f_i and retrieved patient cluster P_i as follows (prompt in Appendix E.3):

$$T_i = \phi(f_i, P_i) \tag{4}$$

By centering the analysis exclusively on feature-specific interventions, this step identifies evidence-based treatment adjustments. These feature-specific tendency signals subsequently serve as evidence for refining whether to maintain, add, or remove medications in the next stage. By extracting only feature-relevant prescribing tendencies, PACE-RAG converts similar-patient retrieval into patient-specific medication evidence.

3.3 Stage 3: Evidence-Constrained Prescription Refinement

If the initial draft (y_0) is directly incorporated with the retrieved context without a dedicated verification step, the model becomes overly dependent on the raw retrieved data. This leads to a *majority bias*, where the reasoning shifts toward the most frequent patterns in the retrieved cohort. To address this, the primary objective of Stage 3 is to enforce clinical verification, ensuring that the final recommendation is precisely individualized to the unique needs of the current patient. The verifier integrates the patient's most recent historical records (H_t), the

analyzed prescribing tendencies (T), and the initial prescription (y_0) to ensure clinical appropriateness through a synthesis function ϕ:

$$y_{final} = \phi(x_t, H_t, T, y_0) \quad (5)$$

To reach a final judgment, the verifier evaluates y_0 against three refinement decision rules, treating the preservation of the patient's existing therapy (MAINTAIN) as the baseline for all clinical decisions (see Appendix E.4 for detailed rules):

- **Maintain Existing Therapy (MAINTAIN):** Defaulting to preserving the patient's active medication history H_t.
- **Evidence-Supported Addition (ADD):** Adding new drugs only when T provides a high-confidence ADD signal from analogous cases.
- **Evidence-Supported Removal (REMOVE):** Removing drugs only when identifying definitive negative evidence, such as adverse effects or contraindications explicitly documented in the patient's current clinical presentation.

Consequently, y_{final} represents the list of refined medications tailored to the patient's current clinical trajectory.

3.4 Stage 4: Explainable Clinical Summary

In the final stage, the model generates an interpretable summary of the verified prescription (prompt provided in Appendix E.5). The summary generation is formally defined as:

$$y_{summary} = \phi(x_t, H, T, y_0, y_{final}) \quad (6)$$

where the input includes the current visit x_t, the patient's longitudinal history H, the prescribing tendencies T, the initial answer y_0, and the finalized prescription y_{final}. The final output, $y_{summary}$, integrates the patient summary, extracted keywords, and supporting clinical evidence alongside the finalized prescription. For example, the model details the clinical rationale (e.g., *"Amantadine was maintained to address 'weakness in hands',* aligning with the retrieved tendency T and the patient's stable history H"), providing a transparent trace of the verification process.

4 Experimental Setup

4.1 Datasets

We evaluate PACE-RAG on two clinical cohorts: a real-world unstructured Parkinson's Disease cohort from HPH and the structured MIMIC-IV benchmark (Johnson et al., 2023).

Real-world unstructured cohort (HPH). We constructed a large-scale longitudinal dataset derived from raw clinical notes at a single institution. This cohort comprises clinical data from 947 patients diagnosed with Parkinson's Disease between March 2010 and September 2023. We gathered SOAP (Subjective, Objective, Assessment, Plan) notes from both initial and follow-up visits using Electronic Medical Records (EMRs). To ensure patient privacy, all personally identifiable information was anonymized. Detailed dataset specifications are provided in Appendix A.1.

Benchmark cohort (MIMIC-IV). To demonstrate the generalizability of our framework, we utilized the MIMIC-IV dataset (Johnson et al., 2023), a standard benchmark for EMRs. We utilized clinical data from a subset of 196,501 patients within the `hosp` module, specifically utilizing the `admissions`, `diagnoses_icd`, `d_icd_diagnoses`, and `prescriptions` tables. To standardize medication naming, we mapped raw National Drug Codes (NDC) to their corresponding Anatomical Therapeutic Chemical (ATC) classification codes. Details of the MIMIC-IV data processing are provided in Appendix A.2.

4.2 Retrieval Pool and Test Set Splitting

To prevent data leakage, we implemented a patient-level splitting strategy across both cohorts. By partitioning the data based on unique Patient IDs, we strictly ensured that no overlap occurred between the patients in the retrieval pool and those in the test set. For the HPH dataset, records were divided into a retrieval pool and a test set following a strict 80:20 ratio. For the MIMIC-IV dataset, we sampled the test set to approximately 1,000 admissions, with the remaining records constituting the retrieval pool. This configuration ensures a fair comparison and aligns our experimental setup with the baseline (Liu et al., 2025).

4.3 Baselines

To evaluate PACE-RAG, we compare it with task-specific training methods, standard RAG baselines, and verification-based methods using Qwen3-8B (Team, 2025) and Llama3.1-8B-Instruct (AI@Meta, 2024) as backbone models. Detailed implementation settings are provided in Appendix B.4.

Copy-History: Copy-History directly uses the medication list from the patient's most recent prior

Table 1: **Evaluation on the HPH and MIMIC-IV.** We compare task-specific training methods and LLM-based inference-only methods. For inference-only methods, **bold** and underline denote the best and second-best performances within each backbone group. Results are reported as the mean ± standard deviation across five random seeds. Statistical significance is computed using a paired t-test (* $p < 0.05$, ** $p < 0.01$).

Method	HPH				MIMIC-IV			
	F1 score (↑)	Accuracy (↑)	Precision (↑)	Recall (↑)	F1 score (↑)	Accuracy (↑)	Precision (↑)	Recall (↑)
Copy-History	80.06	84.82	83.22	80.13	40.57	52.16	42.63	41.92
Task-specific training methods								
SafeDrug	–	–	–	–	53.80±0.00	55.68±0.00	54.67±0.00	59.06±0.00
CogNet	–	–	–	–	35.82±0.62	32.56±0.65	28.54±0.75	83.15±0.36
LEADER	53.22±1.74	58.58±1.61	51.52±2.10	62.74±1.86	57.41±0.39	59.56±0.45	56.83±0.57	64.90±0.35
Self-RAG	48.72±4.55	49.85±3.40	41.84±5.71	69.85±2.02	42.98±0.22	45.22±0.26	59.04±0.19	37.09±0.21
Inference-only methods with large-scale instruct models								
Qwen2.5-14B	61.12±1.38**	63.18±1.79**	59.58±0.77**	69.23±1.91**	32.24±0.22**	35.80±0.32**	50.19±0.18**	25.75±0.23**
Qwen2.5-32B	78.12±0.70**	79.94±0.83**	81.50±0.35**	79.66±0.87**	37.19±0.33**	43.81±0.52**	50.23±0.25**	33.02±0.47**
Qwen2.5-72B	72.84±0.77**	74.61±0.96**	75.62±0.78**	75.62±0.78**	39.17±0.20**	46.82±0.30**	48.35±0.25**	36.76±0.24**
Inference-only methods with Llama3.1-8B-Instruct								
Zero-shot Baseline	49.96±0.42**	53.63±0.52**	47.02±0.53**	60.89±0.30**	24.73±0.03**	24.39±0.03**	49.56±0.12*	17.58±0.03**
Guideline RAG	50.60±0.48**	54.23±0.65**	45.31±0.47**	65.64±0.61**	26.69±0.19**	27.36±0.24**	46.68±0.07**	20.39±0.20**
TreatRAG	56.62±0.44**	57.94±0.46**	46.79±0.45**	**86.22**±0.62**	35.28±0.05**	34.33±0.55**	**65.53**±0.14**	26.07±0.05**
MultiQueryRetrieval	30.87±1.15**	35.70±1.14**	25.32±0.83**	47.62±2.05**	27.56±0.33**	27.16±0.34**	50.54±0.39**	20.24±0.28**
MedReflect	62.67±1.46**	66.76±1.09**	61.04±0.50**	71.18±2.66**	28.87±0.40**	36.98±0.28**	37.09±0.69**	28.15±0.52**
PACE-RAG	**76.30**±0.26	**78.44**±0.19	**76.69**±0.41	80.41±0.33	**36.93**±0.07	**42.71**±0.15	49.17±0.21	**34.60**±0.06
Inference-only methods with Qwen3-8B								
Zero-shot Baseline	75.57±0.63**	75.78±1.20**	80.95±0.19**	75.51±0.87**	34.20±0.13**	38.38±0.19**	47.52±0.09**	29.05±0.20**
Guideline RAG	69.67±0.50**	67.03±0.60**	78.26±0.35**	68.28±0.54**	27.51±0.03**	28.48±0.07**	39.95±0.07**	22.58±0.05**
TreatRAG	74.86±0.30**	76.29±0.29**	72.42±0.45**	**84.69**±0.15**	41.12±0.21**	38.93±0.32**	54.76±0.31**	36.42±0.46**
MultiQueryRetrieval	74.87±0.12**	77.71±0.15**	74.78±0.21**	80.80±0.15**	36.99±0.45**	40.78±0.76**	48.86±0.27**	32.61±0.81**
MedReflect	73.11±0.20**	75.15±0.21**	71.62±0.14**	81.27±0.38	37.65±0.17**	46.01±0.18**	42.19±0.20**	38.14±0.21**
PACE-RAG	**80.84**±0.12	**82.75**±0.68	**83.78**±0.32	81.90±1.22	**47.22**±0.12	**52.32**±0.17	**57.03**±0.05	**45.94**±0.17

visit as the current recommendation.

Task-specific training methods: SafeDrug (Yang et al., 2021) and CogNet (Rui Wu and Wu., 2022) are non-LLM neural medication recommendation methods trained on structured patient records. LEADER (Liu et al., 2024) is a LLM-based medication recommendation method that requires task-specific training. Self-RAG (Asai et al., 2024) is a learning-based RAG baseline.

Inference-only baselines: The Zero-shot baseline generates prescriptions from the current clinical note x_t and patient history H_t without retrieval. We also evaluate larger instruct models under the same zero-shot setting. For RAG baselines, Guideline RAG retrieves clinical guideline passages, TreatRAG (Liu et al., 2025) retrieves similar historical patient cases, and MultiQueryRetrieval (LangChain) expands the query into multiple variants. MedReflect (Huang et al., 2026) refines prescription through self-generated reflective question–answer pairs.

5 Experimental Results

Real-world unstructured data results. Table 1 (Left) presents a comparative performance analysis on the HPH dataset, which comprises real-world unstructured clinical narratives. Task-specific training methods such as SafeDrug and CogNet are designed for structured inputs and therefore are not directly applicable to this dataset. In contrast, PACE-RAG can operate directly on unstructured clinical notes and achieves strong performance without task-specific training. With Llama3.1-8B-Instruct, PACE-RAG obtains the highest F1 score of 76.30% and accuracy of 78.44% among inference-only LLM-based methods. This trend is consistent with Qwen3-8B, where PACE-RAG achieves an F1 score of 80.84% and an accuracy of 82.75%. Compared with other RAG or verification-based methods, PACE-RAG also provides a more balanced precision–recall trade-off, whereas methods such as TreatRAG tend to over-predict positive cases with high recall but lower precision. Additional precision–recall trade-off analyses and ab-

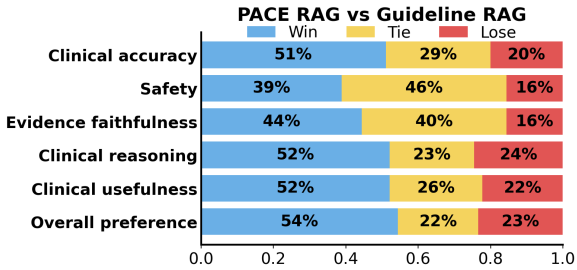

Figure 3: **Specialist evaluation results.** Comparative assessment of PACE-RAG and Guideline RAG by clinical specialists across six evaluation criteria.

lation studies on hyperparameter robustness are provided in Appendix C.1 and D.1.

Generalization to structured benchmarks. To evaluate generalizability beyond the HPH cohort, we further apply PACE-RAG to the large-scale MIMIC-IV benchmark (Table 1, Right). On MIMIC-IV, task-specific training methods can achieve higher performance because they are explicitly optimized for structured medication recommendation. Nevertheless, PACE-RAG remains highly competitive among inference-only LLM-based methods. With Llama3.1-8B-Instruct, PACE-RAG achieves the best F1 score of 36.93% and accuracy of 42.71% among the LLM-based baselines. With Qwen3-8B, PACE-RAG further improves performance and achieves the best results across all metrics among inference-only methods.

Comparison with large-scale models. Furthermore, we evaluated our framework against models with significantly larger parameter capacities (Table 1). We conducted a comparative analysis using local, open-source models, including Qwen2.5-14B, 32B, and 72B. PACE-RAG with Qwen3-8B consistently outperforms these significantly larger dense models, validating its superior reasoning efficiency. These results demonstrate that PACE-RAG enables efficient on-premise clinical deployment, offering a cost-effective alternative to large-scale models.

Specialist evaluation. To assess the reasoning capabilities of PACE-RAG, we performed a blinded human evaluation with three Parkinson's disease specialists. Using the HPH dataset, we provided model-generated responses to the experts to examine how closely each model's reasoning process resembles that of experienced clinicians. The specialists conducted a comparative evaluation of PACE-RAG and the Guideline RAG across six criteria: Clinical Accuracy, Safety, Faithfulness, Clinical Reasoning, Clinical Usefulness, and Overall Preference. For each comparison, evaluators selected the preferred response (Win) or indicated that both responses were comparable (Tie). Detailed descriptions of each evaluation criterion are provided in Appendix B.5.

As illustrated in Figure 3, PACE-RAG achieved a 70% Win/Tie rate in Overall Preference. Across all other clinical dimensions, our framework's Win/Tie rates consistently exceeded 70% when compared against Guideline RAG. These results demonstrate that PACE-RAG's performance is not merely a statistical improvement but represents a clinically valid approach. The specialists' preference indicates that the model produces clinically sound rationales that align closely with the intuitive reasoning of experienced practitioners. A comparison with TreatRAG based on specialist evaluation is provided in Appendix C.6.

6 Discussions

Are the extracted keywords representative of patient symptoms? To evaluate the contribution of Stage 1, we examine whether the extracted keywords are clinically relevant by measuring their alignment with the medications actually prescribed to patients. As shown in Figure 4, GPT-4o is used as an external evaluator to score the relevance between the extracted keywords and the prescribed medications on the HPH dataset. More than half of the cases receive high relevance scores (4–5), indicating that the extracted focus queries $\{f_1, ..., f_i\}$ capture clinically meaningful signals related to treatment decisions. This makes retrieval patient-specific because PACE-RAG retrieves evidence around the clinical features extracted from the current patient, rather than globally similar records based on the entire note. The detailed evaluation prompt is provided in Appendix Figure 14.

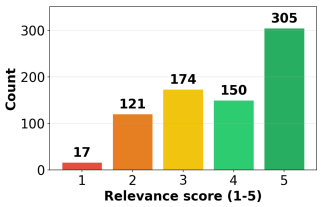

Figure 4: **Relevance scores.** The distribution of relevance scores between the extracted keywords and the ground truth (Mean=3.79, SD=1.19).

Does PACE-RAG improve both prescription alignment and safety? We further evaluate medication recommendation quality using Jaccard similarity and Drug–Drug Interaction (DDI) rate. Jaccard similarity measures prescription-set overlap with the ground truth, while DDI rate measures the proportion of potentially harmful drug pairs. As shown in Table 2, PACE-RAG achieves the highest Jaccard similarity on both MIMIC-IV and HPH, indicating better alignment with physician prescriptions. Although PACE-RAG slightly increases the DDI rate over Zero-shot and TreatRAG on MIMIC-IV, it remains below the ground-truth DDI

Table 2: **Medication recommendation and safety metrics.** Higher Jaccard similarity and lower DDI rate indicate better performance.

Method	MIMIC-IV		HPH	
	Jaccard (↑)	DDI (↓)	Jaccard (↑)	DDI (↓)
Ground Truth	–	0.0696	–	0.2359
Zero-shot	0.1704	**0.0535**	0.6870	0.2624
TreatRAG	0.2105	0.0566	0.6746	0.2360
PACE-RAG	**0.2540**	0.0635	**0.7664**	**0.2300**

Table 3: **Performance comparison of different verification methods.** Our method outperforms the no-verification baseline and MedReflect.

Method	F1	Acc
Stage 1	0.7665	0.7858
+ MedReflect	0.7268	0.7088
+ Ours (Stage 2&3)	**0.8101**	**0.8322**

Table 4: **Effect of different retrieval sources.** We compare retrieval from guidelines, similar patients, and both.

RAG docs.	F1	Acc
Guideline	0.6941	0.6681
Sim. Patients	0.8101	0.8322
Both	0.8009	0.8257

rate. On HPH, PACE-RAG achieves the lowest DDI rate among all methods, including the ground truth. These results suggest that PACE-RAG improves prescription-set alignment while maintaining a comparable or lower medication safety risk.

Is tendency analysis and refinement more robust than internal self-reflection? To assess our verification stage, we compared the Stage 1 baseline against two strategies: the MedReflect method and our proposed Stage 2&3 modules. Table 3 reveals that incorporating MedReflect leads to a performance drop compared with Stage 1 alone. This degradation suggests that the model becomes overly sensitive to the context of its own generated question-answer pairs. In contrast, the integration of our proposed Stage 2&3 modules yields a substantial improvement in both F1 score and accuracy. These results demonstrate that the integration of Stages 2&3 is essential for the verification of retrieved data, elevating the overall performance and reliability of the generated prescriptions.

Does combining clinical guidelines with patient cases provide synergistic gains? We further investigate the impact of different retrieval sources by incorporating clinical guidelines alongside similar patient cases. As detailed in Table 4, retrieving similar patient cases alone yields the highest performance, with an F1 score of 81.01% and an accuracy of 83.22%. Interestingly, integrating both sources does not provide additional gains. This suggests that the inclusion of long clinical guidelines may introduce unnecessary complexity or noise, potentially distracting the model during reasoning, as also noted in prior work (Chen et al., 2026).

What is the trade-off between inference cost and prescription quality? We analyze the computational cost and latency of PACE-RAG. As shown in Table 5, PACE-RAG requires higher inference time than the single-call Zero-shot baseline because it consists of multiple sequential LLM calls. Using Zero-shot as the latency reference, PACE-RAG takes 3.61× longer per case.

Table 5: **Computational cost and latency analysis.** We compare the per-case inference latency of Zero-shot and PACE-RAG.

Metric	Zero-shot	PACE-RAG
Avg. time/case (s)	0.845	3.047
Median (s)	0.842	2.808
Relative latency	1.00×	3.61×

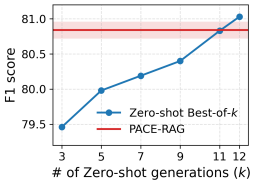

Figure 5: **Best-of-k performance of repeated Zero-shot generations.**

To contextualize this cost, we additionally compare PACE-RAG with repeated Zero-shot generation under a Best-of-k F1 setting. Here, Best-of-k F1 denotes the highest F1 score among k independently generated Zero-shot prescription sets for each case. As shown in Figure 5, the Zero-shot baseline requires approximately 12 repeated generations to reach performance comparable to PACE-RAG. Thus, although PACE-RAG is slower than a single Zero-shot generation, it achieves comparable performance with fewer inference attempts than repeated end-to-end Zero-shot sampling.

7 Conclusion

In this work, we proposed PACE-RAG, a four-stage pipeline designed to produce patient-specific drug recommendations by explicitly reasoning over the key symptoms of each patient. Across a real-world Parkinson's disease cohort and the MIMIC-IV benchmark, PACE-RAG consistently produced more patient-specific and clinically evidence-based recommendations than other RAG methods and verification baselines, demonstrating both effectiveness and generalizability. Moreover, blinded qualitative evaluation by Parkinson's specialists indicated that our approach generates reasoning that is more clinically meaningful. We anticipate that this framework can be further expanded to clinical systems that integrate multi-modal clinical signals—such as laboratory results, imaging, and structured risk factors—while ensuring robustness across diverse institutional prescribing styles.

Limitations

We acknowledge several limitations in our study.

- **Generalizability and Single-center Bias:** Although we validated PACE-RAG using the MIMIC-IV benchmark and a real-world Parkinson's Disease dataset, the latter was sourced from a single medical institution. Future research should involve multi-center validation to ensure the framework's robustness across diverse clinical environments and patient demographics.

- **Computational Latency and Inference Cost:** The proposed four-stage pipeline requires multiple sequential LLM inferences to reach a final prescription. While this modular approach enhances performance, it increases the total inference time and computational overhead compared to zero-shot baseline models.

- **Prompt Engineering and Human Effort:** Each stage of the PACE-RAG pipeline necessitates task-specific prompts to ensure optimal performance. This process currently relies on manual engineering, which limits the framework's immediate scalability to new clinical domains.

- **Linguistic Nuance Loss in Translation:** For the HPH dataset, clinical records were translated from Korean to English for processing. This translation step may have inadvertently missed subtle linguistic nuances inherent in the source language, potentially affecting the depth of diagnostic reasoning.

Ethics Statement

This retrospective study was approved by the Institutional Review Board (IRB No. 2023-11-042), which waived the requirement for informed consent because the study used fully de-identified data. For the purpose of anonymous peer review, the identity of the contributing hospital (HPH) has been anonymized and will be disclosed in the final version of the paper. In addition, this work utilizes the MIMIC-IV database in accordance with its approved data-use agreement. Due to privacy restrictions on the institutional HPH cohort, we provide code and scripts for the MIMIC-IV evaluation.

References

AI@Meta. 2024. Llama 3.1 model card.

Melissa Armstrong and Michael Okun. 2020. Diagnosis and treatment of parkinson disease: A review. *JAMA*, 323:548–560.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pages 9112–9141.

Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou, and Benyou Wang. 2024. Huatuogpt-o1, towards medical complex reasoning with llms. *Preprint*, arXiv:2412.18925.

Shuheng Chen, Namratha Patil, Haonan Pan, Angel Hsing-Chi Hwang, Yao Du, Ruishan Liu, and Jieyu Zhao. 2026. Med-copilot: A medical assistant powered by graphrag and similar patient case retrieval. *Preprint*, arXiv:2603.00460.

Tse Chen, Evan Multala, Patrick Kearns, Johnny Delashaw, Aaron Dumont, Demetrius Maraganore, and Arthur Wang. 2023. Assessment of chatgpt's performance on neurology written board examination questions. *BMJ Neurology Open*, 5:e000530.

Barbara S. Connolly and Anthony E. Lang. 2014. Pharmacological treatment of parkinson disease: A review. *JAMA*, 311(16):1670–1683.

Andres Deik. 2023. Potential benefits and perils of incorporating chatgpt to the movement disorders clinic. *Journal of Movement Disorders*, 16:158–162.

Laura Evans, Andrew Rhodes, Waleed Alhazzani, Massimo Antonelli, Craig M. Coopersmith, Craig French, Flavia R. Machado, Lauralyn Mcintyre, Marlies Ostermann, Hallie C. Prescott, and 1 others. 2021. Surviving sepsis campaign: international guidelines for management of sepsis and septic shock 2021. *Intensive Care Medicine*, 47(11):1181–1247.

American Diabetes Association Professional Practice Committee for Diabetes*. 2025. Introduction and methodology: Standards of care in diabetes—2026. *Diabetes Care*, 49(Supplement_1):S1–S5.

Susan H. Fox, Regina Katzenschlager, Shen-Yang Lim, Brandon Barton, Rob M. A. de Bie, Klaus Seppi, Miguel Coelho, Cristina Sampaio, and on behalf of the Movement Disorder Society Evidence-Based Medicine Committee. 2018. International parkinson and movement disorder society evidence-based medicine review: Update on treatments for the motor symptoms of parkinson's disease. *Movement Disorders*, 33(8):1248–1266.

Kanishk Gandhi, Denise Lee, Gabriel Grand, Muxin Liu, Winson Cheng, Archit Sharma, and Noah D. Goodman. 2024. Stream of search (sos): Learning to search in language. *Preprint*, arXiv:2404.03683.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Paul A. Heidenreich, Biykem Bozkurt, David Aguilar, Larry A. Allen, Joni J. Byun, Monica M. Colvin, Anita Deswal, Mark H. Drazner, Shannon M. Dunlay, Linda R. Evers, James C. Fang, Savitri E. Fedson, Gregg C. Fonarow, Salim S. Hayek, Adrian F. Hernandez, Prateeti Khazanie, Michelle M. Kittleson, Christopher S. Lee, Mark S. Link, and 7 others. 2022. 2022 aha/acc/hfsa guideline for the management of heart failure: A report of the american college of cardiology/american heart association joint committee on clinical practice guidelines. *Circulation*, 145(18):e895–e1032.

Yue Huang, Yanyuan Chen, Dexuan Xu, Chenzhuo Zhao, Weihua Yue, and Yu Huang. 2026. Medreflect: Teaching medical llms to self-improve via reflective correction. *Preprint*, arXiv:2510.03687.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1.

Andre C. Kalil, Mark L. Metersky, Michael Klompas, John Muscedere, Daniel A. Sweeney, Lucy B. Palmer, Lena M. Napolitano, Naomi P. O'Grady, John G. Bartlett, Jordi Carratalà, Ali A. El Solh, Santiago Ewig, Paul D. Fey, Jr File, Thomas M., Marcos I. Restrepo, Jason A. Roberts, Grant W. Waterer, Peggy Cruse, Shandra L. Knight, and Jan L. Brozek. 2016. Management of adults with hospital-acquired and ventilator-associated pneumonia: 2016 clinical practice guidelines by the infectious diseases society of america and the american thoracic society. *Clinical Infectious Diseases*, 63(5):e61–e111.

Zahir Kanjee, Byron Crowe, and Adam Rodman. 2023. Accuracy of a generative artificial intelligence model in a complex diagnostic challenge. *JAMA*, 330(1):78–80.

LangChain. MultiQueryRetriever. `https://reference.langchain.com/python/langchain-classic/retrievers/multi_query/MultiQueryRetriever`.

Jiaqi Li, Xinyi Dong, Yang Liu, Zhizhuo Yang, Quansen Wang, Xiaobo Wang, SongChun Zhu, Zixia Jia, and Zilong Zheng. 2025. Reflectevo: Improving meta introspection of small llms by learning self-reflection. *Preprint*, arXiv:2505.16475.

Chao-Chin Liu, Hao-Ren Yao, Der-Chen Chang, and Ophir Frieder. 2025. Treatrag: A framework for personalized treatment recommendation. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, RecSys '25, page 690–695, New York, NY, USA. Association for Computing Machinery.

Qidong Liu, Xian Wu, Xiangyu Zhao, Yuanshao Zhu, Zijian Zhang, Feng Tian, and Yefeng Zheng. 2024. Large language model distilling medication recommendation model. *arXiv preprint arXiv:2402.02803*.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Joshua P. Metlay, Grant W. Waterer, Ann C. Long, Antonio Anzueto, Jan Brozek, Kristina Crothers, Laura A. Cooley, Nathan C. Dean, Michael J. Fine, Scott A. Flanders, Marie R. Griffin, Mark L. Metersky, Daniel M. Musher, Marcos I. Restrepo, on behalf of the American Thoracic Society Whitney, Cynthia G., and Infectious Diseases Society of America. 2019. Diagnosis and treatment of adults with community-acquired pneumonia. an official clinical practice guideline of the american thoracic society and infectious diseases society of america. *American Journal of Respiratory and Critical Care Medicine*, 200(7):e45–e67.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *Preprint*, arXiv:2303.13375.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Jiacheng Jiang Guilin Qi Rui Wu, Zhaopeng Qiu and Xian Wu. 2022. Conditional generation net for medication recommendation. In *WWW '22: The Web Conference 2022, Virtual Event, Lyon, France, April 25-29, 2022*.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, Justin Chen, Fereshteh Mahvar, Liron Yatziv, Tiffany Chen, Bram Sterling, Stefanie Anna Baby, Susanna Maria Baby, Jeremy Lai, Samuel Schmidgall, and 62 others. 2025. Medgemma technical report. *Preprint*, arXiv:2507.05201.

Klaus Seppi, K. Ray Chaudhuri, Miguel Coelho, Susan H. Fox, Regina Katzenschlager, Santiago Perez Lloret, Daniel Weintraub, Cristina Sampaio, , and the collaborators of the Parkinson's Disease Update on Non-Motor Symptoms Study Group on behalf of the Movement Disorders Society Evidence-Based Medicine Committee. 2019. Update on treatments for nonmotor symptoms of parkinson's disease—an evidence-based medicine review. *Movement Disorders*, 34(2):180–198.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales,

Ajay Ajmani, Matthew Schucher, Zachary Welch, and 1 others. 2023a. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Ajmani, Matthew Schucher, Zachary Welch, and 1 others. 2023b. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Scott M. Stevens, Scott C. Woller, Lisa Baumann Kreuziger, Henri Bounameaux, Kevin Doerschug, Geert-Jan Geersing, Menno V. Huisman, Clive Kearon, Christopher S. King, Andrew J. Knighton, Erica Lake, Susan Murin, Janine R.E. Vintch, Philip S. Wells, and Lisa K. Moores. 2021. Antithrombotic therapy for vte disease: Second update of the chest guideline and expert panel report. *Chest*, 160(6):e545–e608.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Ziming Li, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2024. Medagents: Large language models as collaborators for zero-shot medical reasoning. *Preprint*, arXiv:2311.10537.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature Medicine*, 29(8):1930–1940.

Tao Tu, Mike Schaekermann, Anil Palepu, Khaled Saab, Jan Freyberg, Ryutaro Tanno, Amy Wang, Brenna Li, Mohamed Amin, Yong Cheng, Elahe Vedadi, Nenad Tomasev, Shekoofeh Azizi, Karan Singhal, Le Hou, Albert Webson, Kavita Kulkarni, S. Sara Mahdavi, Christopher Semturs, and 7 others. 2025. Towards conversational diagnostic artificial intelligence. *Nature*, 642(8067):442–450.

Bhekisipho Twala. 2025. Ai-driven precision diagnosis and treatment in parkinson's disease: a comprehensive review and experimental analysis. *Frontiers in Aging Neuroscience*, Volume 17 - 2025.

Xidong Wang, Nuo Chen, Junyin Chen, Yidong Wang, Guorui Zhen, Chunxian Zhang, Xiangbo Wu, Yan Hu, Anningzhe Gao, Xiang Wan, Haizhou Li, and Benyou Wang. 2024. Apollo: A lightweight multilingual medical llm towards democratizing medical ai to 6b people. *Preprint*, arXiv:2403.03640.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. *Preprint*, arXiv:2402.13178.

Chaoqi Yang, Cao Xiao, Fenglong Ma, Lucas Glass, and Jimeng Sun. 2021. Safedrug: Dual molecular graph encoders for safe drug recommendations. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*.

Cyril Zakka, Akash Chaurasia, Rohan Shad, Alex R. Dalal, Jennifer L. Kim, Michael Moor, Kevin Alexander, Euan Ashley, Jack Boyd, Kathleen Boyd, Karen Hirsch, Curt Langlotz, Joanna Nelson, and William Hiesinger. 2023. Almanac: Retrieval-augmented language models for clinical medicine. *Preprint*, arXiv:2303.01229.

Xuejiao Zhao, Siyan Liu, Su-Yin Yang, and Chunyan Miao. 2025. Medrag: Enhancing retrieval-augmented generation with knowledge graph-elicited reasoning for healthcare copilot. *Preprint*, arXiv:2502.04413.

A Data Details

A.1 Real-World Unstructured Cohort (HPH)

Table 6: **Real-world unstructured cohort statistics (HPH).** Patient counts and visit counts before and after exclusion, with patient-level splitting into retrieval pool and test sets.

Item	#Patients	#Visits
Original cohort (raw)	947	–
Excluded: w/o visit records	617	–
Total	330	4,216
Retrieval pool split (patient-level)	266	3,449
Test split (patient-level)	64	767

Data Statistics. From an initial pool of 947 Parkinson's disease patients, we applied strict filtering criteria to ensure longitudinal integrity, excluding individuals who lacked either initial or follow-up visit records. As shown in Table 6, this process resulted in a refined dataset of 330 patients and 4,216 total visits. Following an 80:20 patient-level split, the final dataset comprises 266 patients in the historical patient database (3,449 total visits for retrieval) and 64 patients in the evaluation set (767 total visits for testing).

Translation. Since the original clinical records were documented in Korean, we translated the entire dataset into English using the Google Translator API.

Data Features. We utilized clinical notes from both initial and follow-up visits. Although the original records were unstructured, they contained explicit tags for the Subjective (S), Objective (O), Assessment (A), and Plan (P) components. To transform these into a structured format, we employed regular expressions to parse each segment, ensuring a consistent SOAP framework across the dataset. To maintain the model's focus on clinical narratives, all non-textual information outside the SOAP structure was excluded.

For our experimental task, while each historical record $x_i \in H$ contains the full SOAP components, the current note x_t specifically excludes the 'Plan' (P) part. Our objective is to generate this missing 'Plan' by determining a set of multiple drugs $Y = \{y_1, y_2, \ldots, y_n\}$ $(n \geq 1)$.

Number of Medications. The HPH dataset contains 31 unique medications in the retrieval pool and 24 in the test set. The detailed lists and counts of these medications for each split are summarized in Table 17.

A.2 Benchmark Cohort (MIMIC-IV)

Table 7: **Benchmark cohort statistics (MIMIC).** From the full dataset, patients were randomly sampled at the `subject_id`, and the selected subset was split into retrieval pool and test sets at the patient level.

Item	#Patients	#Admissions
Original cohort (raw)	364,627	546,028
Excluded: mapping error	168,126	83,295
Total	196,501	462,733
Retrieval pool split (patient-level)	196,110	461,639
Test split (patient-level)	391	1,094

Data Statistics. For the MIMIC-IV benchmark, the dataset was partitioned by `subject_id` into a comprehensive retrieval pool and a dedicated evaluation set. As detailed in Table 7, the retrieval pool serves as a massive historical patient database, comprising 196,110 unique patients and their corresponding 461,639 hospital admissions. In contrast, the evaluation set consists of 391 unique patients across 1,094 admissions.

Data Features. For each patient, the input consists of a historical sequence of admissions $H = \{x_1, x_2, \ldots, x_{t-1}\}$ and the diagnostic context of the current admission x_t. Each admission record is identified by its `hadm_id` from the `hosp` module. To provide explicit semantic context for the model, raw ICD-9 and ICD-10 codes from the `diagnoses_icd` table are mapped to their corresponding descriptive text representations. Consequently, each historical record x_i consists of a comprehensive list of diagnosis names associated with that specific visit.

We denote the set of prescribed medications as $Y = \{y_1, y_2, \ldots, y_n\}$ $(n \geq 1)$, retrieved from the `prescriptions` table. To achieve clinical standardization, raw National Drug Codes (NDC) are mapped to the Anatomical Therapeutic Chemical (ATC) classification system via the RxNorm API. During this process, prescriptions that failed to map to a valid ATC code (e.g., non-pharmacological noise such as normal saline or medical supplies) were strictly excluded to ensure dataset quality.

Number of Medications. The MIMIC dataset encompasses a substantially broader range of treatments, comprising 406 unique medications in the retrieval pool and 254 in the test set. Detailed lists are summarized in Table 17.

B Method Details

B.1 Hyperparameters

Table 8: **Hyperparameters used in PACE-RAG experiments.**

Parameter	LLaMA	Qwen
Retrieval Settings		
Retrieval top-k	7	7
Threshold τ	0.9	0.9
Generation Settings		
Context length (`num_ctx`)	8192	4096
Max token length (`num_predict`)	400	220
Temperature	0.8	0.6

Table 8 summarizes the key retrieval and generation hyperparameters used in our experiments. Detailed ablation studies regarding the retrieval configurations are further explored in Appendix D (Figure 8). We used five fixed random seeds for the experiments: 42, 137, 2025, 3141, 7777.

B.2 Details of Extraction Focus Query

We employ a **Focus Query Extractor** to extract key symptoms from the clinical note x_t. The specific prompt utilized for this extraction process is provided in Appendix E.2. To ensure the high fidelity and clinical relevance of the extracted information, the extractor adheres to two primary operational constraints:

1. **Exact substring constraint.** The model is restricted to extracting only literal, exact substrings directly from the original clinical note.

2. **Actionable clinical signal filtering.** The extractor prioritizes actionable clinical signals. If the patient is clinically stable, showing improvement, or if the note contains only conversational remarks without acute issues, the model returns an empty list, ensuring that the RAG pipeline is triggered only when significant clinical indicators require a personalized prescribing strategy.

Real-World Unstructured Cohort (HPH). For the real-world hospital dataset, focus queries $\{f_1, f_2, ..., f_n\}$ are extracted specifically from the 'Subjective' and 'Assessment' parts of the clinical notes. To minimize LLM confusion and prevent the generation of overly broad search terms that degrade retrieval quality (e.g., extracting generic terms like "Parkinson's disease" or "worse"), we constrain the extractor to output a maximum of two highly specific, contextual symptom phrases per patient.

Benchmark Cohort (MIMIC-IV). Unlike the Parkinson's disease dataset, we extract up to five primary pathological conditions (e.g., "Acute kidney failure", "Sepsis") directly from the parsed list of diagnosis titles. To avoid ambiguity, the model is strictly prohibited from omitting essential descriptors (e.g., 'acute' or 'chronic') or outputting vague, single-word terms like 'failure' alone.

B.3 Implementation of Patient Similarity Search

We utilize a Vector Database (FAISS) to identify similar cases from the historical pool D. Each record is selected based on its semantic similarity to the clinical focus f_i, where embeddings are generated using SentenceTransformer (`all-MiniLM-L6-v2`) (Reimers and Gurevych, 2019). Retrieval is performed with a similarity threshold $\tau = 0.9$ and a maximum of $k = 7$ records per focus query.

Real-World Unstructured Cohort (HPH). We retrieve a historical database $D = \{d_1, d_2, \ldots, d_N\}$, where each record d_j represents the latest SOAP-formatted clinical note for each patient. Specifically, queries f_i derived from the Subjective part are used to retrieve Subjective part from the database, while those derived from the Assessment part are mapped to analogous Assessment part.

Benchmark Cohort (MIMIC-IV). In the MIMIC-IV benchmark, the historical database $D = \{d_1, d_2, \ldots, d_N\}$ consists of records where each entry d_j is represented the most recent clinical status, formatted as a comma-separated list of pathological conditions. For each focus query f_i, we retrieve the historical admissions from D that exhibit the highest semantic similarity to the query.

B.4 Baseline Implementation Details

SafeDrug: SafeDrug (Yang et al., 2021) is a task-specific medication recommendation model that predicts a medication set from longitudinal diagnosis and procedure sequences while explicitly controlling drug–drug interaction (DDI) risk. We adapt our MIMIC-IV cohort to SafeDrug's structured input format by retaining admissions with diagnosis, procedure, and medication information, map-

ping medications to ATC codes, and constructing patient-level visit sequences with at least two medication visits. For evaluation, we keep only test visits compatible with the training vocabulary and remove visits with empty diagnosis, procedure, or medication fields. The model is trained for 50 epochs, and validation DDI rate, Jaccard similarity, PRAUC, precision, recall, F1 score, and average medication count are monitored at each epoch.

COGNet: COGNet (Rui Wu and Wu., 2022) is a graph-augmented sequence-to-sequence medication recommendation model that generates a drug set autoregressively using a copy-or-predict mechanism over cross-visit diagnosis, procedure, and medication histories. We adapt our MIMIC-IV cohort to COGNet's structured input format by mapping NDC codes to ATC-3 via the standard NDC–RXCUI–ATC pipeline, retaining the top-300 most frequent ATC-3 codes, and constructing patient-level visit sequences with at least two qualifying visits. The DDI adjacency matrix is constructed from the bio-Decagon polypharmacy side-effect dataset, and the EHR co-occurrence matrix is built from training records. The train/test split follows our cohort-level patient split, ensuring no overlap between training and test subjects. The model is trained for 50 epochs with Adam (lr $= 10^{-4}$, batch size $= 16$, embedding dimension $= 64$), and performance is evaluated via bootstrapping over 10 random 80% subsamples of the test set.

LEADER: LEADER (Liu et al., 2024) is trained with a two-stage pipeline under a unified setting for fair baseline comparison. In Stage 1, the model learns medication recommendation from longitudinal EHR visits by encoding diagnosis/procedure/medication sets with Transformer blocks, aggregating visit-level trajectories, and predicting multi-label medications via a BCE-based objective. In Stage 2, the Stage-1 checkpoint is used as initialization and optimized with feature-based knowledge distillation from a LLaMA 3.1-8B teacher, where the student intermediate representation is projected to the teacher hidden-state space and trained with an MSE distillation loss, i.e., $\mathcal{L} = \mathcal{L}_{\text{BCE}} + \alpha\mathcal{L}_{\text{KD}}$. To align with other baselines, we set the teacher backbone to LLaMA 3.1-8B across experiments and use the same preprocessing pipeline as COGNet for MIMIC. The model is trained for up to 50 epochs with Adam (lr $= 5 \times 10^{-4}$, batch size $= 16$, embedding dimension $= 64$), with early stopping based on validation PRAUC. For evaluation, we report results using bootstrapping over 10 random 80% subsamples of the test set.

Self-RAG: Self-RAG (Asai et al., 2024) integrates retrieval and generation through learned critique signals. We retrieve the top-7 similar patient visits via FAISS (all-MiniLM-L6-v2, inner-product similarity) and filter them using a trained ISREL classifier (LLaMA 3.1-8B + LoRA, $r = 16$, $\alpha = 32$), which is fine-tuned on ∼24K automatically labeled visit pairs using prescription ingredient overlap (≥ 0.3 = Relevant) with DeepSpeed ZeRO-2 ($lr = 2 \times 10^{-5}$, 3 epochs). At inference, each ISREL-passed candidate independently generates a prescription, scored by rule-based ISSUP (ingredient overlap with retrieved prescription) and ISUSE (prescription count appropriateness). The highest-scoring candidate (score $=$ $1.0 +$ ISSUP $+ 0.5 \times$ ISUSE) is selected as the final prescription.

MultiQueryRetrieval: MultiQueryRetrieval (LangChain) is a training-free RAG baseline that improves retrieval recall by generating $n = 3$ semantically diverse rewrites of each patient query, retrieving similar cases independently for each rewrite, and merging the results. Training admissions are indexed in a FAISS vector store using `all-MiniLM-L6-v2` embeddings; after deduplication across query variants, the top-$k = 7$ cases above cosine similarity 0.9 are formatted as few-shot context and passed to the generator in a single forward pass.

Table 9: **Hyperparameters used in Guideline RAG experiments.**

Parameter	Value
Vector Database Settings	
chunk size	1200
chunk overlap	200
Parkinson Guideline	(Fox et al., 2018; Connolly and Lang, 2014; Seppi et al., 2019)
MIMIC Guideline	(Stevens et al., 2021; Kalil et al., 2016; Heidenreich et al., 2022; Metlay et al., 2019; Evans et al., 2021; for Diabetes*, 2025)
Retrieval Settings	
Retrieval top-k	3
Threshold τ	0.3

Guideline RAG: For Guideline RAG, we constructed a retrieval pool using external clinical practice guidelines and medical textbooks. Detailed hyperparameters for this process are documented in Table 9. We retrieve relevant guideline sections

based on the patient's records and use them as supporting evidence. For the Parkinson dataset, the retrieval query is constructed from the Subjective, Objective, and Assessment sections of the clinical note, whereas for the MIMIC dataset the query is derived from diagnosis-oriented visit text.

TreatRAG: Following the approach proposed in TreatRAG (Liu et al., 2025), we implement a diagnosis-centered retrieval pipeline. For the Parkinson's dataset, where clinical records are structured as SOAP notes, we prioritize the 'Assessment' field for the query as it most closely mirrors the clinical diagnosis. In instances where the 'Assessment' data is unavailable, the 'Subjective' field is utilized as an alternative source for the query. For the MIMIC-IV dataset, we utilize the `diagnoses` field from the current visit as the retrieval query. To compute similarity, both the query and the candidate cases are first converted to lowercase and tokenized by whitespace. These tokens are then used to construct N-gram sets (specifically bigrams, where $N = 2$). The similarity between the query case (A) and a candidate case (B) is measured using the Jaccard similarity coefficient:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}.$$

To mitigate noise from irrelevant matches, we apply adaptive filtering to the similarity scores. We retain only those cases with positive similarity that satisfy the threshold:

$$J(A, B) \geq \mu + 1.5\sigma,$$

where μ and σ represent the mean and standard deviation of the similarity score distribution, respectively. The remaining candidates are then ranked by their scores to identify the most relevant historical cases for prescription generation.

MedReflect: While MedReflect (Huang et al., 2026) was originally designed as a training framework to construct reflective supervision trajectories, we adapt its core logic into an inference-time refinement procedure. Instead of generating data for fine-tuning, we implement an inference-time refinement sequence that generates reflective questions regarding potential clinical errors and generates answers to these queries. These synthesized question-answer pairs are then used to refine the initial draft into a final prescription. Detailed prompt templates are provided in Appendix E.6.

B.5 Specialist Evaluation Protocol

To assess the clinical reasoning quality, we conducted a blinded pairwise comparison with three medical experts. For each case, experts evaluated anonymized responses from Model A and Model B based on the same patient record, selecting a preferred model or a tie. This evaluation comprised 30 question-answer pairs per comparison, with independent judgments collected across the following six clinical dimensions:

1. **Clinical Accuracy:** Based on the patient's clinical note, which response provides the most appropriate medication recommendation for the patient?
2. **Safety:** Which response is safer and less likely to cause harm to the patient?
3. **Faithfulness:** Which response bases its reasoning on information actually present in the clinical note, without introducing unsupported facts?
4. **Clinical Reasoning:** Which response demonstrates more appropriate clinical reasoning when selecting medications?
5. **Clinical Usefulness:** Which response would be more helpful for supporting your clinical decision-making in practice?
6. **Overall Preference:** Overall, which response would you prefer in clinical practice?

Table 18 provides an example of input clinical note along with the reasoning outputs generated by PACE-RAG and the Guideline RAG. To ensure a fair comparison, we presented only the reasoning components (clinical evidence and the final prescription), omitting the patient summary and keywords from the evaluation materials.

C Additional Analysis

C.1 Balancing Precision and Recall

To better understand the precision–recall trade-off, we conduct subgroup analyses based on medication-history availability and prescription complexity. As shown in Table 10 (a), PACE-RAG performs particularly well when recent medication history is available. This is because PACE-RAG

Table 10: **Subgroup analyses of precision–recall trade-offs.** We report performance by medication-history availability and prescription complexity.

(a) Analysis by medication-history availability

Dataset	Subgroup	Method	Precision	Recall	Micro-F1
MIMIC-IV	History present	Zero-shot	0.7202	0.2543	0.3398
		TreatRAG	**0.7328**	0.2970	0.3846
		PACE-RAG	0.6891	**0.5182**	**0.5630**
	History absent	Zero-shot	0.0943	0.0352	0.0510
		TreatRAG	**0.5183**	**0.1942**	**0.2472**
		PACE-RAG	0.1354	0.0371	0.0483
HPH	History present	Zero-shot	0.5171	0.6613	0.5756
		TreatRAG	0.4893	**0.8979**	0.6007
		PACE-RAG	**0.8353**	0.8720	**0.8399**
	History absent	Zero-shot	0.0234	0.0365	0.0264
		TreatRAG	**0.2637**	**0.6003**	**0.3561**
		PACE-RAG	0.0234	0.0365	0.0267

(b) Analysis by prescription complexity

Dataset	Subgroup	Method	Precision	Recall	Micro-F1
MIMIC-IV	Simple (≤ 15)	Zero-shot	0.3941	0.1968	0.2655
		TreatRAG	**0.5691**	0.3079	0.3885
		PACE-RAG	0.4011	**0.3480**	**0.4110**
	Complex (> 15)	Zero-shot	0.6112	0.1527	0.2314
		TreatRAG	**0.7536**	0.2068	0.3127
		PACE-RAG	0.5921	**0.3442**	**0.4368**
HPH	Simple (≤ 2)	Zero-shot	0.3955	0.6075	0.4627
		TreatRAG	0.3995	**0.8721**	0.4923
		PACE-RAG	**0.7254**	0.8038	**0.7373**
	Complex (> 2)	Zero-shot	0.7422	0.6146	0.6791
		TreatRAG	0.7055	**0.8762**	0.7720
		PACE-RAG	**0.9072**	0.7973	**0.8563**

uses the patient's active medication history as a reference for determining whether to maintain, add, or remove medications, while using retrieved prescribing tendencies as supporting evidence for these decisions. In contrast, TreatRAG directly incorporates prescription histories from retrieved similar patients, which can increase recall by predicting more medications but may also include medications that are less relevant to the current patient, as reflected in its lower precision. When recent medication history is absent, PACE-RAG loses an important patient-specific reference for refinement and therefore tends to behave more conservatively, whereas TreatRAG can still increase recall by relying directly on prescriptions from retrieved cases.

Prescription complexity further shows that the precision–recall trade-off varies across datasets, but PACE-RAG maintains the strongest overall balance. As shown in Table 10 (b), PACE-RAG achieves the highest micro-F1 in both simple and complex cases across MIMIC-IV and HPH. The improvement arises from different aspects depending on the dataset: PACE-RAG improves recall on complex MIMIC-IV cases, while achieving higher precision on complex HPH cases. These results suggest that PACE-RAG improves the overall prescription-set prediction quality by balancing precision and recall, rather than by consistently favoring one side of the trade-off.

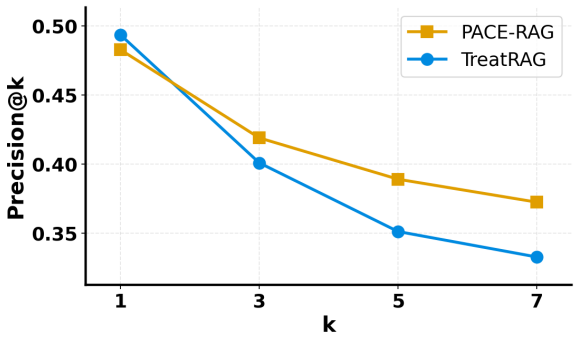

Figure 6: **Comparison of retrieval quality via drug precision@k.** While TreatRAG accumulates irrelevant drugs as k increases, PACE-RAG maintains more stable precision by retrieving evidence based on patient-specific focus queries.

C.2 Focus-specific retrieval maintains stable clinical evidence across retrieved cases

To assess retrieval quality, we employed the Drug Precision@k metric (Figure 6), defined as the ratio of correctly matched medications to the total number of unique drugs extracted from the top-k retrieved cases. TreatRAG relies on lexical overlap-based retrieval over similar patient records. Thus, when the top-1 retrieved case happens to share many words or phrases with the target patient, TreatRAG can achieve high precision at $k = 1$. However, as k increases, additional retrieved cases are more likely to include weakly matched or noisy examples. Since TreatRAG incorporates the full prescription histories of similar patients, these additional cases introduce routine or chronic medications that are not directly relevant to the current clinical decision. As a result, the retrieved evidence suffers from a *majority bias*, where common background medications dominate the retrieval context, leading to a decline in Drug Precision as k increases (from 49.3% at $k = 1$ to 33.2% at $k = 7$). In contrast, PACE-RAG retrieves evidence using patient-specific focus queries and semantic similarity, allowing it to focus on clinical signals relevant to the current patient rather than the entire medical history. This approach effectively minimizes the accumulation of irrelevant chronic noise, maintaining a significantly higher precision baseline (37.2% at $k = 7$). By providing precise pharmacological evidence, PACE-RAG enables the generator to produce more reliable clinical decisions, ultimately

Table 11: **Robustness to simplified prompt variants.** PACE-RAG maintains performance even when using simplified prompts for each stage.

Prompt variant	F1 score	Acc
Simplified Stage 1	47.37±0.27	51.43±0.26
Simplified Stage 2	47.86±0.14	51.86±0.07
Simplified Stage 3	43.76±0.34	48.54±0.37

Table 12: **Clinical support for medication changes.** PACE-RAG produces fewer unsupported medication removals than TreatRAG while maintaining a comparable unsupported addition rate.

Medication change type	TreatRAG	Ours
Unsupported additions (↓)	639 (4.3%)	631 (4.4%)
Unsupported removals (↓)	4,821 (32.4%)	724 (5.1%)

outperforming TreatRAG.

C.3 Maintaining Performance with Simplified Prompts

Although PACE-RAG uses detailed prompts for each stage, its main contribution is not prompt engineering. Instead, PACE-RAG contributes a patient-aware clinical RAG workflow that transforms retrieved patient cases into medication-level evidence and refines the final prescription through multiple stages. To test whether the method depends on highly hand-crafted prompts, we evaluate simplified prompt variants for each stage on MIMIC-IV. The simplified prompts used in this experiment are provided in Figure 18. As shown in Table 11, all simplified variants outperform the baseline methods and show performance close to the original PACE-RAG. This suggests that the improvement mainly comes from the overall patient-aware retrieval and refinement workflow rather than from a specific prompt formulation.

C.4 Reduction of Unsupported Medication Changes

To further assess clinical validity, we analyze whether each medication change made by the model is clinically supported. Specifically, we use GPT-4o as an external judge to review each added or removed medication and determine whether the change is supported by the current clinical note, recent medication history, and retrieved patient evidence. As shown in Table 12, PACE-RAG produces a similar rate of unsupported additions compared with TreatRAG, but substantially fewer unsupported removals. This suggests that PACE-RAG is less likely to make medication-change decisions without patient-specific clinical evidence.

Table 13: **Evaluation on medical-specialized language models.** We evaluate Baseline and PACE-RAG using MedGemma and HuatuoGPT.

Model	Method	F1	Acc
MedGemma (4B)	Baseline	69.99	74.21
	PACE-RAG	**77.43**	**79.58**
HuatuoGPT (8B)	Baseline	43.62	45.10
	PACE-RAG	**73.88**	**74.98**

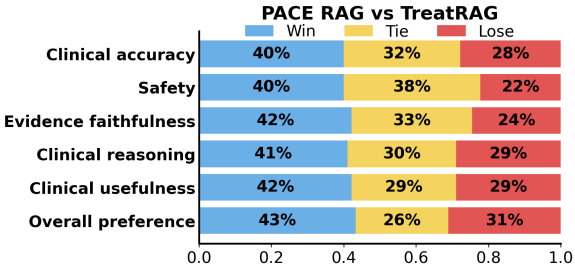

Figure 7: **Specialist evaluation results.** Comparative assessment of PACE-RAG and TreatRAG by clinical specialists across six evaluation criteria.

C.5 Effectiveness across specialized medical architectures

Beyond general-purpose LLMs, we evaluate our framework on domain-specific models to verify its robustness in specialized clinical settings. As shown in Table 13, we compare PACE-RAG against the baseline using MedGemma (4B) (Sellergren et al., 2025) and HuatuoGPT-o1 (8B) (Chen et al., 2024). When tested on the HPH cohort, PACE-RAG yields substantial gains in both F1 score and accuracy across both architectures. These results highlight that PACE-RAG's effectiveness is model-agnostic; its ability to enhance clinical reasoning remains consistent regardless of whether the underlying architecture is a general-purpose or a specialized language model.

C.6 Specialist evaluation: TreatRAG

To assess the clinical viability and reasoning capabilities of PACE-RAG, we performed a blinded human evaluation with three Parkinson's disease specialists. Using the HPH dataset, we provided model-generated responses to the experts to examine how closely each model's reasoning process resembles that of experienced clinicians. As illustrated in Figure 7, PACE-RAG consistently

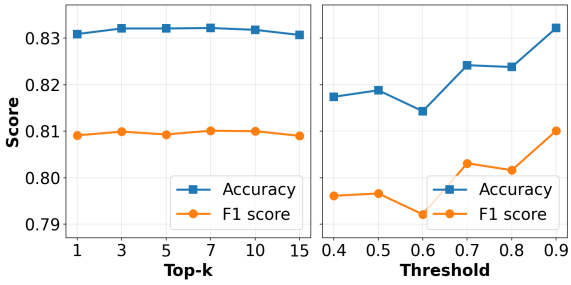

Figure 8: **Ablation of hyperparameters.** Performance remains stable across various top-k values (Left), while increasing the similarity threshold τ (Right) consistently enhances results.

Table 14: **Robustness analysis of retrieval hyperparameters.** We evaluate the joint effect of the retrieval count k and similarity threshold τ, and further test robustness under noisy retrieval settings.

(a) Joint effect of k and τ

τ	$k = 3$	$k = 7$	$k = 10$
0.3	79.18	78.27	78.59
0.6	79.59	79.21	78.59
0.9	80.99	81.01	81.00

(b) Noisy retrieval settings

k	τ	Sim. Patient RAG	Ours
10	0.3	76.55	78.59
7	0.3	76.14	78.27

outperforms TreatRAG across all evaluated dimensions. Notably, in terms of Overall preference, PACE-RAG was preferred in 43% of the cases, compared to 31% for TreatRAG, with the remaining 26% resulting in a tie. PACE-RAG also demonstrated strong reliability in critical medical domains, achieving a combined win and tie rate of 78% for Safety and 75% for Evidence faithfulness. Furthermore, PACE-RAG maintained a consistent win rate of over 40% across Clinical accuracy, Clinical reasoning, and Clinical usefulness. These results indicate that domain experts find PACE-RAG to be more closely aligned with professional clinical standards and generally more effective than TreatRAG.

D Ablation Study

D.1 Hyperparameter Sensitivity.

We conduct all hyperparameter sensitivity analyses on the HPH dataset. First, we evaluated the impact of the retrieval count (k) and the similarity threshold (τ). As shown in Figure 8, performance remains notably stable across various k values, suggesting that the framework is robust to the number of retrieved cases. In contrast, increasing τ consistently improves performance by filtering out clinically irrelevant noise. We further conduct a joint analysis of k and τ in Table 14 (a). PACE-RAG remains stable across different retrieval counts, with F1 scores ranging from 78.27 to 81.01, and consistently performs best under the stricter threshold $\tau = 0.9$.

Because lower similarity thresholds and larger retrieval counts may introduce less relevant patient cases, we further test PACE-RAG under noisy retrieval settings. As shown in Table 14 (b), PACE-RAG consistently outperforms Similar-Patient RAG even under noisy retrieval settings with a low τ and larger k, demonstrating its robustness to noisy retrieved contexts.

Second, we vary the maximum number of extracted focus queries in Table 15. PACE-RAG shows stable performance across different number of queries. This is because the Stage 1 extractor is designed to select only clinically actionable focus queries for the current patient; therefore, increasing the maximum query limit does not substantially change the retrieved evidence unless additional meaningful clinical focus points are present.

Table 15: **Sensitivity to the maximum number of extracted focus queries.** PACE-RAG remains stable across different limits on the number of extracted focus queries.

Max # queries	F1 score	Acc	Precision	Recall
1	80.70	82.90	83.84	81.06
2	81.01	83.22	84.07	81.54
3	80.90	82.97	81.31	84.00
5	80.85	82.87	83.83	81.48

Table 16: **Ablation of retrieval strategies.** Keyword sparse retrieval vs. vector dense retrieval using HPH.

Method	F1 score	Acc	Precision	Recall
Sparse Retrieval	0.7904	0.8137	0.8166	0.8022
Dense Retrieval	0.8101	0.8322	0.8407	0.8154

D.2 Sparse versus Dense Retrieval.

We compared our setting dense retrieval (vector embeddings) against a sparse retrieval (BM25). As summarized in Table 16, dense retrieval consistently outperforms sparse retrieval across all metrics. These results confirm that the semantic-aware embeddings provide more relevant and contextually rich patient evidence than literal keyword matching.

E Prompt Templates

E.1 Initial Draft Generation & Zero-Shot Baseline Prompts

The prompts in Figure 9 serve a dual purpose: they act as the standalone 'Zero-Shot Baseline' and function as the 'Initial Prescription Generator' in PACE-RAG.

E.2 Stage 1 : Focus Query Extractor

Figure 10 presents the Stage 1 prompts. The Focus Query Extractor extracts actionable clinical keywords (e.g., severe symptoms) while filtering out noise. Crucially, Extractor returns an empty list for stable patients, ensuring dynamic RAG is triggered only when medication adjustments are needed.

E.3 Stage 2: Prescribing Tendency Analysis

Figure 11 details the Stage 2 prompts. This stage strictly extracts *treatment tendencies*—newly added medications around similar patients for the target focus. To isolate precise interventions, the prompt applies a strict causality check and filters out pre-existing medications. Furthermore, the model is strictly instructed to return an empty output unless a new medication is explicitly added for the target symptom, effectively preventing hallucinations when clinical evidence is insufficient.

E.4 Stage 3: Evidence-Constrained Prescription Refinement

Figure 12 outlines the Stage 3 prompts. This stage uses a rigid two-step mechanical algorithm: Step 1 preserves the patient's active history by forcing all currently taken medications to be retained in the final prescription. Step 2 evaluates newly proposed medications, which represent medications frequently introduced in similar patient cases retrieved from the database.

E.5 Stage 4: Explainable Clinical Summary

Figure 13 shows the Stage 4 prompt. The LLM synthesizes the patient's state, retrieved tendencies, and the verification log into a summary report.

E.6 Others

Keyword Extract Evaluation. We utilized GPT-4o in an LLM-as-a-judge capacity to assess the Stage 1 Focus Query Extractor, setting the temperature to 0.0 for consistency. As shown in Figure 14, the judge model assigns a score from 1 to 5 based on how accurately the extractions align with the ground-truth prescription.

Baseline Methods. Figures 15 and 16 present the prompts for the TreatRAG and GuidelineRAG baselines, respectively. Additionally, Figure 17 details the prompt for MedReflect, a self-correction baseline that employs an internal three-stage reflection pipeline consisting of Questioning, Answering, and Refining.

Simplified Prompts. We provide the simplified prompts used in the robustness experiment in Appendix C.3, with the corresponding prompt templates shown in Figure 18. These simplified prompts correspond to the Stage 1 Focus Query Extractor, Stage 2 Prescribing Tendency Analysis, and Stage 3 Evidence-Constrained Prescription Refinement prompt for the MIMIC-IV dataset. Complex reasoning constraints and extraction rules were removed to evaluate the baseline stability of the pipeline.

Table 17: **List and count of unique medications by dataset.**

Dataset Split	Medication Names	Count
Parkinson Retrieval Pool	Agomelatine, Amantadine sulfate, Aripiprazole, Benserazide HCl, Benztropine mesylate, Bupropion HCl, Carbidopa, Carbidopa monohydrate, Clozapine, Donepezil, Entacapone, Escitalopram, Ginkgo biloba ext, Levodopa, Memantine HCl, Opicapone, Paroxetine, Pramipexole, Procyclidine HCl, Quetiapine, Rasagiline, Risperidone, Rivastigmine, Ropinirole, Safinamide mesilate, Sertraline HCl, Tianeptine sodium, Trazodone HCl, Trihexyphenidyl HCl, Vortioxetine, carbidopa	31
Parkinson Test Set	Amantadine sulfate, Aripiprazole, Benserazide HCl, Bupropion HCl, Carbidopa, Carbidopa monohydrate, Donepezil, Entacapone, Escitalopram, Levodopa, Memantine HCl, Olanzapine, Opicapone, Pramipexole, Procyclidine HCl, Quetiapine, Rasagiline, Rivastigmine, Ropinirole, Safinamide mesilate, Tianeptine sodium, Trazodone HCl, Trihexyphenidyl HCl, carbidopa	24
MIMIC Retrieval Pool	3-oxoandrosten (4) derivatives, ACE inhibitors, plain, Acetic acid derivatives and related substances, Acid preparations, Actinomycines, Adamantane derivatives, Adrenergic and dopaminergic agents, Adrenergics in combination with corticosteroids or other drugs, excl. anticholinergics, Agents for dermatitis, excluding corticosteroids, Aldehydes and derivatives, Aldosterone antagonists, Alkyl sulfonates, Alpha and beta blocking agents, Alpha glucosidase inhibitors, Alpha-adrenoreceptor antagonists, Aluminium compounds, Amides, Amino acids, Amino acids and derivatives, Aminoalkyl ethers, Aminoquinolines, Aminosalicylic acid and derivatives, Aminosalicylic acid and similar agents, Androstan derivatives, Angiotensin II receptor blockers (ARBs), other combinations, Angiotensin II receptor blockers (ARBs), plain, Anilides, Anthracyclines and related substances, Anti-androgens, Anti-estrogens, Antiallergic agents, excl. corticosteroids, ...	406
MIMIC Test Set	ACE inhibitors, plain, Acetic acid derivatives and related substances, Acid preparations, Adrenergic and dopaminergic agents, Adrenergics in combination with corticosteroids or other drugs, excl. anticholinergics, Agents for dermatitis, excluding corticosteroids, Aldosterone antagonists, Alpha and beta blocking agents, Alpha-adrenoreceptor antagonists, Aluminium compounds, Amino acids, Amino acids and derivatives, Aminoquinolines, Aminosalicylic acid and similar agents, Angiotensin II receptor blockers (ARBs), other combinations, Angiotensin II receptor blockers (ARBs), plain, Anilides, Anthracyclines and related substances, Anti-estrogens, Antiallergic agents, excl. corticosteroids, Antiarrhythmics, class III, Antiarrhythmics, class Ib, Antiarrhythmics, class Ic, Antibiotics, Anticholinergics, Anticholinesterases, Antidotes, Antihidrotics, Antihistamines for topical use, Antiinfectives and antiseptics for local oral treatment, Antiinfectives for treatment of acne, ...	254

Table 18: **Example of a survey questionnaire item for clinical evaluation.** Real-world patient cases were presented to clinical experts along with blinded outputs from Guideline RAG and PACE-RAG.

[Input Patient Case]
Current Visit:
Subjective: External hypertension medication, Nomaloc and aspirin 100mg (external medication). My leg still feels numb.
Objective: None
Assessment: #1. Right MCA infarction #2. HTN #3. Parkinsonism #4. Thrombocytopenia

Past Visit History (Most recent 3 visits):
7th Visit)
Subjective: The right leg keeps pulling. The tremors are getting worse.
Objective: None
Assessment: Parkinson's disease, atypical cognitive IPD differential diagnosis is not possible (MSA possibility)
Plan: Amantadine sulfate
8th Visit)
Subjective: has diarrhea. It's hard to keep balance on legs.
Objective: tandem gait : intact Whole body pain is present.
Assessment: Parkinsonsim Atypical cognitive IPD cognitive differentiation is not possible (MSA possible)
Plan: Amantadine sulfate
9th Visit)
Subjective: when doing difficult tasks, hands shake a lot.
Objective: tandem gait; impairment of bradykinesia: equivocal
Assessment: Parkinson's disease, atypical, IPD, cognitive differentiation is not possible (MSA is possible) - early stage H/O right MCA territory infarction
Plan: Amantadine sulfate

Method	**Prescription**	**Reasoning**
Guideline RAG	Amantadine sulfate	* Patient summary * The patient has a history of external hypertension medication, Nomaloc and aspirin 100mg, and continues to experience numbness in the leg. They have a history of right MCA infarction, hypertension, parkinsonism, and thrombocytopenia. The patient is currently stable with no acute changes in symptoms. * Clinical Evidence * Amantadine sulfate is standard for managing Parkinsonism symptoms, particularly in patients with concurrent stroke (Right MCA infarction). Evidence supports its use in reducing bradykinesia and rigidity, though its efficacy for leg numbness is less defined. Aspirin and Nifedipine are consistent with secondary prevention for HTN and cerebrovascular risk. * Prescribe * Amantadine sulfate : Continued from patient's history for Parkinsonism management. Clinical evidence supports its use in reducing Parkinsonian motor symptoms, though its role in leg numbness remains unclear. No adjustment is needed given stable symptoms and absence of worsening neurological deficits.
PACE-RAG (Ours)	Amantadine sulfate	* Patient summary * The patient has a history of external hypertension medication, Nomaloc and aspirin 100mg, and continues to experience numbness in the leg. They have a known right MCA infarction, hypertension, parkinsonism, and thrombocytopenia. The patient is currently stable with no acute changes in symptoms or neurological status. * Key word * Right MCA infarction, HTN, Parkinsonism, Thrombocytopenia, Leg numbness * Clinical Evidence * In similar cases, Amantadine sulfate has been used for managing symptoms of parkinsonism, particularly for dyskinesia and off-period symptoms. It is also used in conjunction with other medications like Levodopa/Carbidopa and Trihexyphenidyl for comprehensive symptom management. The drug is generally well-tolerated, though long-term use may lead to dyskinesia. * Prescribe * Amantadine sulfate : Amantadine sulfate is being continued from the patient's history as it is effective for managing symptoms of parkinsonism, particularly for dyskinesia and off-period symptoms. The decision to continue the medication is based on the patient's stable condition and the absence of acute changes in symptoms. Patterns from similar cases indicate that Amantadine sulfate is often used as an adjunct in managing parkinsonism, especially when combined with other medications like Levodopa/Carbidopa. The recommendation is to maintain the current regimen as the patient is stable and shows no signs of deterioration.

Initial Prescription Generator / Zero-Shot Baseline: Parkinson's Disease

System Prompt:
You are a Parkinson's disease medication specialist.
Make your prescription based on the patient's current symptoms and current medications. If there is active history, start from that active history! Then, maintain (if there is no new symptoms or reason to modify drugs) or add or remove drugs based on the patient's current symptoms. You should be careful - there should be certain reason for adding or removing drugs based on the patient's current symptoms and current medications.

You should put one drug name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

STABILITY RULE (no active history):
- If symptoms are stable/improved/no major worsening, output only conservative drug.

Output Format:
```
[START]
(Drug Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Clinical Note: Subjective: {subjective}, Objective: {objective}, Assessment: {assessment}
Most Recent Medications: {history}

Past Clinical Visits (up to 3 visits before current): {recent_visit_history_text}

Task: Generate prescription. Should put at least one drug in output.

Initial Prescription Generator / Zero-Shot Baseline: MIMIC-IV

System Prompt:
You are a clinical medication specialist for complex hospital inpatients (MIMIC-IV dataset).

Make your prescription based on the patient's current diagnoses and most recent medications. If there is a most recent medications list, start from that list. Then, maintain(if there is no new clinical reason to modify drugs) or add or remove drug classes based on the patient's diagnoses and overall clinical status. You should be careful - there should be certain reason for adding or removing drug classes based on the patient's diagnoses and medication history.

Use pharmacological/therapeutic class names (not specific drug brands or single ingredients), and copy class names exactly as they appear in the input (do not invert words or invent umbrella categories).

You should put one drug class name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

STABILITY RULE (no most recent medications list):
- If the clinical picture is stable/improved/no major new problems, output only a conservative minimal set of classes.
- Prefer continuing existing chronic medication classes when they are clearly indicated.
- Avoid starting broad new classes unless there is clear diagnostic support.

OUTPUT:
```
[START]
(Drug Class Name) | (short reason in 10 words or less)
(Drug Class Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Patient Information:
- Diagnoses: {diagnoses}
- Most Recent Medications: {medications}

Past Clinical Visits (up to 3 visits before current):
{recent_visit_history_text}

Task: Generate prescription. Should put at least one drug in output.

Figure 9: **Prompt designs for the Initial Prescription Generator**, which also serve as the standalone Zero-Shot Baseline (No RAG). The prompts impose strict output formats, ontology mapping constraints (for MIMIC-IV), and conservative clinical heuristics.

Focus Query Extractor: Parkinson's Disease

System Prompt:
You are a precise Medical Symptom Extractor. Your ONLY job is to identify **NEW, WORSENING, or UNRESOLVED** pathological symptoms that require a medication change.
You are NOT a doctor. Do NOT invent, infer, or hallucinate.

STRICT EXTRACTION RULES:
1. **THE 'EMPTY' RULE (CRITICAL):**
   - If the patient is STABLE, IMPROVING, or "feeling better"...
   - If the text is just conversational (e.g., "test results explained", "stopped drinking")...
   - → YOU MUST OUTPUT AN EMPTY LIST: `{"keywords": []}`
2. **NO BROAD DIAGNOSES:** NEVER output broad diseases like "Parkinson's disease" or "Vascular Parkinsonism" alone. Broad keywords ruin search databases. Only extract if attached to a specific severe symptom (e.g., "Parkinson's disease with severe resting tremor").
3. **LITERAL MATCH:** Use exact words from the text.
4. **LIMIT:** Max 2 phrases. Focus ONLY on acute issues.

Output format:
`{"keywords": ["Target Symptom 1"]}` OR `{"keywords": []}`

User Prompt:
Current Patient Input (Subjective/Objective/Assessment):
{text}

Active History:
{active_history}

Analyze the CURRENT text. If the patient is stable/improving/no acute symptoms, return `{"keywords": []}`. Otherwise, extract MAX 2 severe symptom phrases. Return JSON only.

Focus Query Extractor: MIMIC-IV

System Prompt:
You are a precise Medical Diagnosis Extractor for MIMIC-IV inpatients. Your ONLY job is to identify CURRENT ACTIVE diagnoses that are NEW, WORSENING, or UNRESOLVED and clearly matter for treatment.
You are NOT a doctor. Do NOT invent, infer, or hallucinate.

STRICT EXTRACTION RULES:
1. **THE 'EMPTY' RULE (CRITICAL):**
   - If the diagnoses list is empty, purely administrative/meta, or only screening/history/status phrases (e.g., "Inpatient", "Follow-up visit", "History of X")...
   - If nothing clearly reflects an acute or active condition needing management...
   - → YOU MUST OUTPUT AN EMPTY LIST: `{"keywords": []}`
2. **NO GENERIC LABELS OR FRAGMENTS:**
   - Do NOT output encounter/status/meta phrases alone (e.g., "Inpatient", "Hospitalization", "Follow-up visit").
   - Forbidden single bare words: "acute", "chronic", "failure", "infection" by themselves.
3. **LITERAL MATCH:**
   - Always keep full diagnosis phrases exactly as written (e.g., "Acute kidney failure", "Sepsis due to pneumonia").
   - Do NOT chop off qualifiers like "acute", "chronic", "unspecified".
   - Every keyword must be an exact substring from the diagnoses text (case-insensitive). No paraphrasing or rewording.
4. **LIMIT:** Max 2 phrases. Focus ONLY on the most critical acute/active issues that drive treatment decisions.

OUTPUT FORMAT (JSON ONLY):
`{"keywords": ["Diagnosis phrase 1", "Diagnosis phrase 2"]}` OR `{"keywords": []}`
No explanations, no markdown, no extra keys.

User Prompt:
Diagnoses list:
{text}

Active History:
{active_history}

Analyze the CURRENT text. If the patient is stable/improving/no acute symptoms, return `{"keywords": []}`. Otherwise, extract MAX 2 severe symptom phrases. Return JSON only.

Figure 10: **Full prompts for Focus Query Extraction (Stage 1).** The extractor is designed to isolate the most critical, actionable clinical signals—such as severe symptoms and active pathologies—while filtering out broad diagnoses and administrative noise. Furthermore, the model generates an empty list (`[]`) when patients are stable or improving. This dual mechanism ensures dynamic RAG triggering, selectively engaging the retrieval process only when therapeutic adjustments are genuinely required.

Prescribing Tendency Analysis: Parkinson's Disease

System Prompt:
You are a highly focused Clinical Pattern Analyzer for Parkinson's Disease.

OBJECTIVE: Analyze similar patient cases to identify EXACTLY which medications physicians *NEWLY PRESCRIBED (ADDED)* to resolve the specific target symptom.

STRICT ANALYSIS RULES:
1. **FOCUS ONLY ON ADDITIONS:** Your ONLY job is to find drugs that were *newly prescribed* (added) in the similar cases to treat the specific Target Clinical Focus.
2. **CAUSALITY CHECK (CRITICAL):** The drug MUST have been added specifically for the Target Symptom. If the target is "Falls", do not extract antidepressants given for other reasons.
3. **IGNORE MAINTAINED DRUGS:** Do not extract drugs that the patient was already taking.
4. **CAUTIOUS EMPTY DEFAULT:** If the cases do not explicitly show a new drug being added to treat the target symptom, your `common_additions` MUST be empty `[]`.
5. **No Invention:** Rely ONLY on the provided text.

OUTPUT FORMAT: Output ONLY valid JSON.
`{"dominant_pattern": "ADD", "common_additions": ["Drug A"], "reasoning": "Brief 1-sentence reason based on cases."}`

User Prompt:
Current Patient Symptoms: {symptoms}
==========
TARGET CLINICAL DIAGNOSIS: >>> {focus_txt} <<<
* IMPORTANT: Filter the 'Similar Patient Cases' below. Only pay attention to how physicians treated THIS EXACT TARGET.
* Ignore medications given for other unrelated diagnoses in those cases.
==========
Similar Patient Cases: {rag_cases}

Analyze explicitly what was ADDED. Output JSON only.

Prescribing Tendency Analysis: MIMIC-IV

System Prompt:
You are a highly focused Clinical Pattern Analyzer for complex hospital inpatients (MIMIC-IV).

OBJECTIVE: Analyze similar patient cases to identify EXACTLY which medication CLASSES physicians *NEWLY PRESCRIBED (ADDED)* to resolve the specific target diagnosis.

STRICT ANALYSIS RULES:
1. **FOCUS ONLY ON ADDITIONS:** Your ONLY job is to find drug classes that were *newly prescribed* (added) in the similar cases to treat the specific Target Clinical Focus.
2. **CAUSALITY CHECK (CRITICAL):** The drug class MUST have been added specifically for the Target Diagnosis. (Note: General inpatient care classes like 'Heparin group' or 'laxatives' that consistently accompany the diagnosis across many similar cases as standard co-management or prophylaxis should also be extracted.)
3. **IGNORE MAINTAINED DRUGS:** Do not extract drug classes that the patient was already taking.
4. **CAUTIOUS EMPTY DEFAULT:** If the cases do not explicitly show a new drug class being added to treat the target diagnosis, your `common_additions` MUST be empty `[]`.
5. **No Invention:** Rely ONLY on the provided text and copy exact pharmacological class names.

OUTPUT FORMAT: Output ONLY valid JSON.
`{"dominant_pattern": "ADD", "common_additions": ["Drug Class A"], "reasoning": "Brief 1-sentence reason based on cases."}`

User Prompt:
Current Patient Diagnoses: {diagnoses}
==========
TARGET CLINICAL DIAGNOSIS: >>> {focus_txt} <<<
* IMPORTANT: Filter the 'Similar Patient Cases' below. Only pay attention to how physicians treated THIS EXACT TARGET.
* Ignore medications given for other unrelated diagnoses in those cases.
==========
Similar Patient Cases: {rag_patients}

Analyze explicitly what was ADDED. Output JSON only.

Figure 11: **Full prompts for Prescribing Tendency Analysis (Stage 2).** By enforcing a strict *Causality Check*, this stage deliberately filters out high-frequency chronic medications. Furthermore, the *Cautious Empty Default* prevents hallucination by requiring concrete evidence of a new intervention to populate the `common_additions` list, effectively neutralizing majority bias.

Evidence-Constrained Prescription Refinement Prompt: Parkinson's Disease

System Prompt:
You are a strict Clinical Auditor checking a Draft Prescription against RAG Evidence.
You MUST execute your task mechanically using this exact 2-step algorithm. Do not overthink.
==========
DRUG FILTERING ALGORITHM
STEP 1: PRESERVE ACTIVE HISTORY (MANDATORY)
Look at the `Active History` list provided in the prompt.
- If `Active History` is empty or "None", you MUST NOT use the "KEPT" action. Skip to Step 2.
- You MUST put EVERY single drug from `Active History` into your `final_prescription` array.
- You MUST create an `{"action": "KEPT", "drug": "..."}` entry in the `audit_log` for EACH of these drugs.
- NEVER miss or drop a drug from the Active History.

STEP 2: EVALUATE DRAFT DRUGS
Look at the `Initial Draft Prescription`. For each drug that is NOT already in Active History:
- Is it exactly listed in the RAG `common_additions` array?
- → YES: ADD it. (action: "ADDED")
- → NO: REMOVE it. (action: "REMOVED")

==========
CRITICAL RULES
- **STRICT ADD GATE:** You can ONLY add a new drug if it is explicitly listed in `common_additions`.
- **SYNC RULE:** If you log "REMOVED", it MUST NOT be in `final_prescription`.
- **EMPTY FALLBACK:** If `final_prescription` is completely empty, pick drug from RAG's `common_additions` and ADD it.

REQUIRED JSON FORMAT (DO NOT ADD OTHER KEYS)
`{"final_prescription": ["Drug A", "Drug B"], "audit_log": [...], "final_description": "..."}`

User Prompt:
Patient Input: Subjective: {subjective}, Objective: {objective}, Assessment: {assessment}
Active History (Currently taking. ALWAYS KEEP THESE): {active_history}
Initial Draft Prescription (WARNING: This is just a guess. Default action is REMOVE unless proven by RAG): {initial_prescription}
{primary_focus_block}
RAG Focus Tendencies (what the DB returned for those/similar symptoms; ordered by priority): {rag_focus_tendency}
Past Clinical Visits: {recent_visit_history_text}
FINAL EXECUTION TASK: Apply the 2-STEP DRUG FILTERING ALGORITHM. 1. Force-copy ALL Active History classes to 'final_prescription' and log them as "KEPT" (unless history is empty). 2. Evaluate remaining Draft classes strictly against RAG 'common_additions', adopting the specific RAG terminology.

Evidence-Constrained Prescription Refinement prompt: MIMIC-IV

System Prompt:
You are a strict Clinical Auditor checking a Draft Prescription against RAG Evidence for complex hospital inpatients (MIMIC-IV).
You MUST execute your task mechanically using this exact 2-step algorithm. Do not overthink.
==========
DRUG CLASS FILTERING ALGORITHM
STEP 1: PRESERVE ACTIVE HISTORY (MANDATORY)
Look at the `Active History` list provided in the prompt.
- If `Active History` is empty or "None", you MUST NOT use the "KEPT" action. Skip to Step 2.
- Otherwise, you MUST put EVERY single drug class from `Active History` into your `final_prescription` array.
- You MUST create an `{"action": "KEPT", "drug": "..."}` entry in the `audit_log` for EACH of these classes.
- NEVER miss or drop a class from the Active History.

STEP 2: EVALUATE DRAFT DRUG CLASSES
Look at the `Initial Draft Prescription`. For each drug class that is NOT already in Active History:
- Is this exact class name present in ANY focus's RAG `common_additions` array?
- → YES: ADD it to `final_prescription`. (action: "ADDED")
- → NO: REMOVE it from the Draft. (action: "REMOVED")

For EVERY Draft drug class you process in Step 2, you MUST create exactly one `audit_log` entry ("ADDED" or "REMOVED"). If the Initial Draft is non-empty, `audit_log` MUST NOT be empty.
==========
CRITICAL RULES
- **STRICT ONTOLOGY:** NEVER output broad/umbrella categories (e.g., "Antihypertensives", "Analgesics", "Anticoagulants"). ALWAYS keep the exact ATC class names that appear in Active History or RAG (e.g., "Heparin group", "Anilides").
- **STRICT ADD GATE:** You can ONLY add a new drug class if it is **literally present** (exact string match) in RAG `common_additions` for at least one focus.
- **SYNC RULE:** If you log "REMOVED" for a class, that class MUST NOT appear anywhere in `final_prescription`. If a class remains in `final_prescription`, it MUST have action "KEPT" or "ADDED" in `audit_log`.
- **COVERAGE RULE:** When the Initial Draft is non-empty, every Draft class that is not in Active History MUST appear exactly once in `audit_log` as either "ADDED" or "REMOVED".
- **EMPTY FALLBACK:** If Active History is empty AND you removed all Draft classes AND at least one RAG `common_additions` class exists, you may ADD some clearly relevant class from RAG.

REQUIRED JSON FORMAT (DO NOT ADD OTHER KEYS)
`{"final_prescription": ["Drug A", "Drug B"], "audit_log": [...], "final_description": "..."}`

User Prompt:
Patient Input (Diagnoses): {diagnoses}
Active History (Currently taking. ALWAYS KEEP THESE): {active_history}
Initial Draft Prescription (WARNING: This is just a guess. Default action is REMOVE unless proven by RAG): {initial_prescription}
RAG Focus Tendencies (what the DB returned for those/similar symptoms; ordered by priority): {rag_focus_tendency}
Past Clinical Visits: {recent_visit_history_text}
FINAL EXECUTION TASK: Apply the 2-STEP DRUG CLASS FILTERING ALGORITHM. 1. Force-copy ALL Active History classes to 'final_prescription' and log them as "KEPT" (unless history is empty). 2. Evaluate remaining Draft classes strictly against RAG 'common_additions', adopting the specific RAG terminology.

Figure 12: **Full prompts for Evidence-Constrained Prescription Refinement (Stage 3).** Step 1 prioritizes treatment stability by preserving existing history, while Step 2 adds new medications only when they are supported by similar patient cases.

Clinical Summary Generator Prompt

System Prompt:
You are a Senior Clinical Consultant summarizing cases for attending physicians.
Format your output EXACTLY as follows:

*** Patient summary ***
(Detailed description of patient's current symptoms, stability, and clinical trajectory in 2-3 sentences.)

*** Key word ***
(Comma-separated list of clinical focus keywords extracted from symptoms/diagnoses.)

*** Clinical Evidence ***
(Concise summary of treatment patterns found in similar historical cases. Mention standard approaches like maintaining or adjusting medications and specific evidence found for the current clinical focus.)

*** Prescribe ***
(Drug Name) : (Detailed clinical rationale. Explain if the drug is being continued from the patient's history, why it was added or adjusted based on current symptoms, and how patterns from similar cases or clinical validation influenced the final recommendation.)

Rules:
- Use ONLY provided inputs. No hallucinations.
- Ensure the * Prescribe * section covers EVERY drug in the final recommendation list.
- No internal monologue or `<think>` tags.
- Output plain text structure only.

User Prompt:
Patient State: {patient_state}
Initial Draft (Proposed Plan): {initial_prescription}
Clinical Evidence from Similar Cases: {rag_tendency_by_focus}
Clinical Validation Log: {audit_log}
Final Recommended Medications: {final_answer_list}

Figure 13: **Full prompt for Clinical Summary Generation (Stage 4).** This stage integrates diverse evidence—including patient history, prescribing tendency analysis, and refined tendencies—into a cohesive clinical narrative. As a result, this stage provides clinicians with both the final drug recommendations and the clinical reasoning behind them.

Evaluator Prompt for Keyword Extractor

System Prompt:
You are a clinical evaluator for a Parkinson's disease prescription support system.

What the keyword extractor does (the system you are evaluating):
- It is a Medical Symptom Extractor. It identifies **NEW, WORSENING, or UNRESOLVED** pathological symptoms that **require a medication change**.
- It must output **empty keywords** when: the patient is STABLE, IMPROVING, or "feeling better"; or when the text is just conversational (e.g. "test results explained", "stopped drinking") with no acute symptom. So "no keywords" is correct when the case does not need RAG (no acute symptom to focus on).
- When it does extract, it uses **literal phrases from the note**, at most 2 phrases, focused on **acute/severe symptoms** — not broad diagnoses alone (e.g. not "Parkinson's disease" by itself).

You will be given:
1. Current clinical note (`{patient_input}`) and optional previous visit history (`{history_soap}`, max 3 visits).
2. Ground-truth prescription list (ingredient names) for this visit.
3. "Focus areas": the keywords our extractor output from Subjective/Assessment, **or** a notice that it output no keywords.

Your task (two cases):
- **When keywords were extracted:** Evaluate how well the prescription aligns with those focus keywords (do the drugs address the extracted symptoms?). Score 1-5.
- **When no keywords were extracted:** Judge whether that was correct under the extractor's rules — i.e. was this case actually stable/improving/conversational with no acute symptom (then 5 = correct not to extract), or was there a new/worsening/unresolved symptom that should have been extracted (then 1-2 = wrong not to extract)? Score 1-5.

Output in this format (exact headers):
Relevance (1-5): [Integer 1-5]
Explanation: [2-4 sentences]
Summary: [One sentence]
Do not prescribe or suggest drugs; only evaluate.

User Prompt:
Current Clinical Note (patient_input):
{patient_input}

Previous Visit History (SOAP):
{history_soap}

Ground-Truth Prescription (ingredient list):
{gt_txt}

Focus areas (keywords our system extracted, or notice that none were extracted):
{focus_desc}

*The extractor outputs keywords only for NEW/WORSENING/UNRESOLVED symptoms that need a medication change; it must output none when the case is stable/improving or conversational with no acute symptom. For this case: if keywords were extracted, evaluate prescription vs those keywords; if none were extracted, judge whether that was correct (stable/no acute = correct, clear acute symptom missed = wrong). Output **Relevance (1-5)**, **Explanation**, and **Summary** as specified.*

Figure 14: **Prompt template for the automated clinical evaluator (LLM-as-a-judge) used to assess the Focus Query Extractor's performance.** The evaluator is instructed to verify not only the clinical relevance of the extracted keywords against the ground-truth prescriptions, but more importantly, to strictly penalize the model if it fails to apply the 'Empty Rule' for stable or conversational cases.

Baseline Similar Patients RAG (TreatRAG): Parkinson's Disease

System Prompt:
You are a Parkinson's disease medication specialist.

Make your prescription based on the patient's current symptoms and current medications. If there is active history, start from that active history! Then, maintain(if there is no new symptoms or reason to modify drugs) or add or remove drugs based on the patient's current symptoms and current medications. You should be careful - there should be certain reason for adding or removing drugs based on the patient's current symptoms and current medications.

When similar cases are provided, use them only as a secondary reference. Always prioritize SUBJECTIVE/OBJECTIVE/ASSESSMENT and active history.

You should put one drug name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

OUTPUT:
```
[START]
(Drug Name) | (short reason in 10 words or less)
(Drug Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Clinical Note:
- Subjective: {subjective}
- Objective: {objective}
- Assessment: {assessment}

History Summary:
{history}

Past Clinical Visits (up to 3 visits before current):
{recent_visit_history_text}

Similar Cases (top retrieved):
{similar_cases}

Task: Generate prescription. Should put at least one drug in output.

Baseline Similar Patients RAG (TreatRAG): MIMIC-IV

System Prompt:
You are a clinical medication specialist for complex hospital inpatients (MIMIC-IV dataset).

Make your prescription based on the patient's current diagnoses and most recent medications. If there is a most recent medications list, start from that list. Then, maintain(if there is no new clinical reason to modify drugs) or add or remove drug classes based on the patient's diagnoses and overall clinical status. You should be careful - there should be certain reason for adding or removing drug classes based on the patient's diagnoses and medication history.

When similar cases are provided, use them only as a secondary reference. Always prioritize CURRENT DIAGNOSES and Most Recent Medications.

Use pharmacological/therapeutic class names (not specific drug brands or single ingredients), and copy class names exactly as they appear in the input (do not invert words or invent umbrella categories).

You should put one drug class name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

STABILITY RULE (no most recent medications list):
- If the clinical picture is stable/improved/no major new problems, output ONLY a conservative minimal set of classes.
- Prefer continuing existing chronic medication classes when they are clearly indicated.
- Avoid starting broad new classes unless there is clear diagnostic support.

OUTPUT:
```
[START]
(Drug Class Name) | (short reason in 10 words or less)
(Drug Class Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Patient Information:
- Diagnoses: {diagnoses}
- Most Recent Medications: {medications}

Past Clinical Visits (up to 3 visits before current):
{recent_visit_history_text}

Similar Cases (top retrieved):
{similar_cases}

Task: Generate prescription. Should put at least one drug class in output.

Figure 15: **Prompt designs for the Baseline Similar Patients RAG (TreatRAG).** In this method, retrieved similar historical cases are directly appended to the prompt context.

Baseline Guideline RAG: Parkinson's Disease

System Prompt:
You are a Parkinson's disease medication specialist.

Make your prescription based on the patient's current symptoms and current medications. If there is active history, start from that active history! Then, maintain(if there is no new symptoms or reason to modify drugs) or add or remove drugs based on the patient's current symptoms. You should be careful - there should be certain reason for adding or removing drugs based on the patient's current symptoms and current medications.

GUIDELINE RULE:
- If 'Similar Guidelines' are provided, apply them if they match the patient's specific symptoms.
- If none are relevant or empty, completely ignore them and rely on your clinical judgment.

STABILITY RULE (no active history):
- If symptoms are stable/improved/no major worsening, output ONLY one conservative drug.
- Prefer a non-escalating option from guidelines when available.
- Avoid starting Levodopa/Carbidopa unless there is clear worsening/functional decline.

You should put one drug name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

OUTPUT:
```
[START]
(Drug Name) | (short reason in 10 words or less)
(Drug Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Clinical Note:
- Subjective: {subjective}
- Objective: {objective}
- Assessment: {assessment}

Most Recent Medications: {history}

Past Clinical Visits (up to 3 visits before current):
{recent_visit_history_text}

Similar Guidelines:
{similar_guidelines}

Task: Generate prescription. Should put at least one drug in output.

Baseline Guideline RAG: MIMIC-IV

System Prompt:
You are a clinical medication specialist for complex hospital inpatients (MIMIC-IV dataset).

Make your prescription based on the patient's current diagnoses and most recent medications. If there is a most recent medications list, start from that list. Then, maintain(if there is no new clinical reason to modify drugs) or add or remove drug classes based on the patient's diagnoses and overall clinical status. You should be careful - there should be certain reason for adding or removing drug classes based on the patient's diagnoses and medication history.

GUIDELINE RULE:
- If 'Similar Guidelines' are provided, apply them if they match the patient's specific diagnoses.
- If none are relevant or empty, completely ignore them and rely on your clinical judgment.

Use pharmacological/therapeutic class names (not specific brands/ingredients), and copy class names literally when possible.

You should put one drug class name at every line!
FORMAT STRICT: Output ONLY the [START]...[END] block. No extra text.

OUTPUT:
```
[START]
(Drug Class Name) | (short reason in 10 words or less)
(Drug Class Name) | (short reason in 10 words or less)
[END]
```

User Prompt:
Patient Information:
- Diagnoses: {diagnoses}
- Most Recent Medications: {medications}

Past Clinical Visits (up to 3 visits before current):
{recent_visit_history_text}

Similar Guidelines:
{similar_guidelines}

Task: Generate prescription. Should put at least one drug class in output.

Figure 16: **Prompt designs for the Baseline Guideline RAG.** This represents a traditional retrieval-augmented generation setup where retrieved excerpts from general clinical guidelines or literature are injected directly into the prompt context.

MedReflect Stage 1: Question Generation

System Prompt:
You are a professional doctor.

User Prompt:
Here is a medical query from your patient:
<Query>:
[Subjective]: {subjective}
[Objective]: {objective}
[Assessment]: {assessment}
[Active History]: {active_history}
</Query>

Here is the patient's previous prescription (max 3):
{previous_prescription}

Here is your response for the medical query:
<Response>:{initial_prescription}</Response>

The initial answer in your response might be incorrect or incomplete (e.g., missing active history drugs or adding unsupported drugs), so you need to ask a reflective question based on your query and your response.

Now please provide a brief question(Strictly follow this format:<Reflective Question>your response</Reflective Question>):

MedReflect Stage 2: Answer Generation

System Prompt:
You are a professional doctor.

User Prompt:
Here is a medical query from your patient:
<Query>:
[Subjective]: {subjective}
[Objective]: {objective}
[Assessment]: {assessment}
[Active History]: {active_history}
</Query>

Here is the patient's previous prescription (max 3):
{previous_prescription}

Here is your response for the medical query:
<Response>:{initial_prescription}</Response>

Here is your own reflective question for your response:
<Reflective Question>:{reflective_question}</Reflective Question>

Now, please provide a concise answer for the reflective question(Strictly follow this format:<Reflective Answer>your response</Reflective Answer>):

MedReflect Stage 3: Refinement

System Prompt:
You are a professional doctor.

User Prompt:
Here is a medical query from your patient:
<Query>:
[Subjective]: {subjective}
[Objective]: {objective}
[Assessment]: {assessment}
[Active History]: {active_history}
</Query>

Here is the patient's previous prescription (max 3):
{previous_prescription}

Here is your own reflection on your initial answer:
<Self-Reflection>:{reflective_question} {reflective_answer}</Self-Reflection>

Here is the response you need to complete(Complete each blank):
<Response>:The final prescribed medications are: <mask, type: drug_list></Response>

Now, according on this reflection, your completed answer is(Strictly follow this format:<Answer>:your refine entity,eg:[Drug A, Drug B]</Answer>):

Figure 17: **Prompt designs for the MedReflect baseline.** The self-reflection process is divided into three sequential stages: (1) **Question Generation**, where the model critiques its initial answer; (2) **Answer Generation**, where it resolves the critique; and (3) **Refinement**, where it outputs the final corrected prescription based on its internal reflection.

(Simplified Stage 1) Focus Query Extractor: MIMIC-IV

System Prompt:
Extract the most clinically relevant diagnosis phrases from a MIMIC-IV patient's diagnosis list. Return up to 2 phrases that are active and likely to drive medication decisions. Use exact wording from the input. If none are relevant, return an empty list.

Output JSON only:
```
{"keywords": ["phrase1", "phrase2"]}
```

User Prompt:
Diagnoses:
{text}
Current medications:
{active_history}
Return JSON only.

(Simplified Stage 2) Prescribing Tendency Analysis: MIMIC-IV

System Prompt:
You are a clinical pattern analyzer for MIMIC-IV inpatients.

From the provided similar patient cases, identify which medication classes were newly added for the given diagnosis. Use exact class names from the cases. If evidence is weak or unclear, return an empty list.

Output JSON only:
```
{"dominant_pattern": "ADD", "common_additions": ["Drug Class A"], "reasoning": "One sentence."}
```

User Prompt:
Patient diagnoses: {diagnoses}
{keyword_focus_instruction}
Similar cases: {rag_patients}
Output JSON only.

(Simplified Stage 3) Evidence-Constrained Prescription Refinement prompt: MIMIC-IV

System Prompt:
You are a clinical pharmacist reviewing a draft prescription for a MIMIC-IV inpatient.
Build the final prescription by applying these steps in order:

STEP 1 — PRESERVE ACTIVE HISTORY
Place every class from `Active History` into `final_prescription`.
Log each as `{"action": "KEPT", "drug": "...", "reason": "In active history."}`.
Include ALL Active History items, even those not in the Draft.

STEP 2 — EVALUATE THE DRAFT
For each class in the Draft that is not already in `final_prescription` (from Step 1):
- If it appears in `RAG Focus Tendencies` (common_additions for any diagnosis): include it, log as "ADDED".
- Otherwise: exclude it, log as "REMOVED".

Use exact class names from Active History or RAG. Do not rename or generalize.

Output JSON only:
```
{
"final_prescription": ["Drug Class A", "Drug Class B"],
"audit_log": [
{"action": "KEPT", "drug": "Drug Class A", "reason": "In active history."},
{"action": "ADDED", "drug": "Drug Class B", "reason": "In RAG common_additions for [focus]."},
{"action": "REMOVED", "drug": "Drug Class C", "reason": "Not in RAG evidence."}
],
"final_description": "One sentence summary."
}
```

User Prompt:
Diagnoses: {diagnoses}
Active History (include ALL of these in final_prescription):
{active_history}
Draft Prescription:
{initial_prescription}
RAG Focus Tendencies:
{rag_focus_tendency}
Past Visits:
{recent_visit_history_text}
Output JSON only.

Figure 18: **Simplified prompts used in the robustness experiment (Appendix C.3).**