RxDiff: Discrete Diffusion for Medication Recommendation with Inference-Time Safety Control

Jinke Feng[1,2] **Wenjie Du**[1,2,†] **Xuchang Zhong**[3] **Yang Wang**[1,2,†]

[1]University of Science and Technology of China, China
[2]Suzhou Institute for Advanced Research, USTC, China
[3]Beijing Institute of Technology, China

fengjinke@mail.ustc.edu.cn
{duwenjie,angyan}@ustc.edu.cn
xuchang.zhong@bit.edu.cn

Abstract

Medication recommendation requires generating drug combinations that are therapeutically effective while minimizing harmful drug-drug interactions (DDIs). Both objectives are rooted in the combinatorial nature of prescriptions. Existing discriminative methods predict drugs independently, neglecting inter-drug dependencies; autoregressive methods introduce sequential dependencies but impose arbitrary generation orders and accumulate errors. Both paradigms confine DDI mitigation to training-time penalties, offering limited capacity to regulate interactions during inference for individual prescriptions. We reformulate medication recommendation as a discrete diffusion process and propose RxDiff, which leverages iterative denoising for bidirectional and revisable generation and provides a natural interface for dynamic DDI regulation at each step. A clinically grounded dual-guided forward process and an inference-time interaction-aware modulation mechanism are designed to jointly enable RxDiff to achieve consistent accuracy improvements, lower and more individually controlled DDI rates, and robustness to evolving DDI knowledge across MIMIC-III, MIMIC-IV, and a real-world inpatient dataset. Code is available at https://github.com/f1sher0/RxDiff.

1 Introduction

Managing patients with multiple concurrent conditions requires carefully selecting combinations of co-administered medications, a task that grows increasingly complex with the rising prevalence of multimorbidity [30, 34, 45, 37]. The widespread adoption of Electronic Health Records (EHR) has enabled data-driven approaches to support such decisions [1, 36], giving rise to learning-based medication recommendation systems. Early methods leveraged longitudinal patient history for personalized recommendation [35, 51, 46]; subsequent work incorporated molecular pharmacology and drug interaction constraints to jointly improve accuracy and safety [47, 49]; recent efforts refined recommendations through further set-level prediction, fine-grained interaction severity and patient-specific substructure modeling [39, 5, 11]. Together, these advances have achieved consistent progress in both recommendation accuracy and interaction safety on real-world EHR benchmarks.

While accuracy and safety are typically treated as two separate optimization objectives, they both stem from the inherent combinatorial nature of prescriptions. Accuracy reflects therapeutic efficacy, which depends not only on individual drug effects but critically on inter-drug synergy: Amoxicillin combined with Clavulanic Acid extends antibacterial coverage against β-lactamase-producing pathogens [6],

[†]Corresponding authors

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

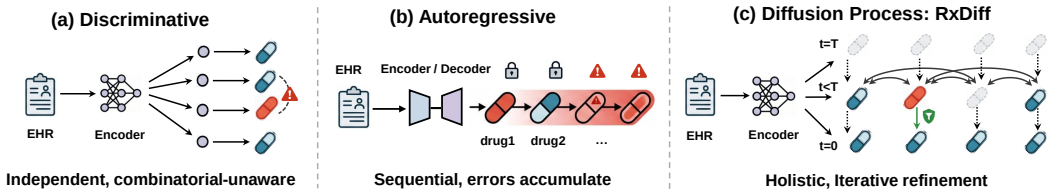

Figure 1: Paradigm comparison. (a) Discriminative methods predict drugs independently, missing combinatorial dependencies. (b) Autoregressive methods impose sequential order, accumulating errors. (c) RxDiff captures holistic dependencies and refines prescriptions via iterative denoising.

and tuberculosis treatment requires concurrent multi-drug regimens to prevent resistance [3]. In both cases, single-agent therapy is clinically insufficient. Safety is combinatorial in turn, as drug-drug interactions (DDIs) are properties of drug pairs rather than of individuals, e.g., Warfarin and Aspirin are each safe in isolation, yet their combination substantially elevates bleeding risk [13]. Better combinatorial modeling is therefore essential to more accurate and safer medication recommendation.

Existing methods, however, exhibit fundamental limitations in modeling this combinatorial property. Discriminative approaches [35, 47, 49, 11] formulate medication recommendation as multi-label classification with independent per-drug output heads, leaving combinatorial property largely outside the modeling target. While 4SDrug [39] further adopts a set-to-set formulation, it reduces prediction to per-drug scoring, with combinatorial structure captured only indirectly. Autoregressive approaches [51, 46] introduce partial inter-drug dependencies through sequential generation, but the sequential paradigm itself imposes an arbitrary generation order as an inductive bias [43, 38], restricts each prediction to a unidirectional prefix without access to the full prescription state, and prevents revision of sampled drugs, allowing errors to accumulate along the generation chain [4, 32, 23, 20].

In addition, existing methods show limited capacity for inference-time safety control. Current DDI-mitigation mechanisms—whether DDI-related loss penalties [35, 49, 5, 11], DDI-aware representations [46], or reinforcement learning rewards that penalize interaction-laden generations [51]—primarily act at training time, instilling a parametric prior that statistically minimizes DDI in expectation. While broadly effective, such a prior provides no per-prescription regulation, leaving high-DDI cases possible and potentially compromising individual patients [14]. Besides, residual interactions, e.g., rare in dataset or newly identified [40, 31, 21, 22, 50], can slip through the trained prior with limited constraint. An explicit mechanism to intervene in the inference process itself is therefore needed to actively regulate DDI risk during the generation of any individual prescription.

To address these limitations, in this paper we reformulate medication recommendation as a discrete diffusion process and instantiate it as a novel framework, RxDiff. Under this formulation, a prescription is progressively corrupted into noise through a forward process, and a model learns to reverse this corruption via iterative denoising, reconstructing the prescription step by step. This reformulation addresses both limitations in a principled manner. For combinatorial modeling, iterative denoising allows the model to perceive the current state of the entire prescription at every generation step, with each medication predicted under bidirectional awareness of all others; later steps further revise earlier decisions based on the holistic prescription state, avoiding the chain-accumulated errors inherent to autoregressive generation. For inference-time safety control, the step-wise generation naturally admits dynamic DDI constraints conditioned on the current prescription state at each step, yielding safety regulation throughout the entire inference trajectory.

Building on this formulation, RxDiff further introduces two principled designs that tailor discrete diffusion to the clinical requirements of medication recommendation. First, we propose a background-frequency- and DDI-guided forward process that encodes prescription specificity and interaction risk directly into the diffusion dynamics, driving the model to capture patient-specific drug combinations while steering it away from hazardous ones. Second, we introduce interaction-aware modulation during iterative denoising, which imposes controllable DDI constraints on each single prescription at inference time. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to reformulate medication recommendation as a discrete diffusion process, establishing a unified generative framework that provides a principled foundation for holistic combinatorial modeling and inference-time safety regulation.

- We propose a clinically grounded dual-guided forward process while preserving closed-form marginals for efficient training, together with an inference-time interaction-aware modulation mechanism that imposes tunable, per-prescription DDI constraints at each denoising step.

- Extensive experiments on two public benchmark datasets and an additional real-world general inpatient dataset demonstrate that RxDiff achieves consistent accuracy improvements, lower average DDI rates, and more favorable individual-level DDI distributions.

2 Related Work

Medication recommendation. By architectural design, existing methods can be broadly categorized into discriminative and generative approaches. Discriminative methods formulate the task as multi-label classification over the drug vocabulary. GAMENet [35] combines EHR co-occurrence and DDI graphs via a memory network; SafeDrug [47] incorporates molecular substructures with DDI-aware regularization; MoleRec [49] and SubRec [11] refine this direction with substructure-level modeling; REFINE [5] further introduces a DDI severity graph to weight interactions by clinical risk. Generative methods instead model inter-drug dependencies through sequential generation: LEAP [51] generates prescriptions autoregressively with reinforcement fine-tuning for DDI reduction, and COGNet [46] extends this with a Transformer copy mechanism that leverages longitudinal history. Among existing work, 4SDrug [39] is conceptually closest to ours in recognizing the importance of combinatorial modeling and reformulating the task as set-to-set prediction; however, constrained by the discriminative framework, its prediction still reduces to per-drug scoring followed by aggregation to avoid combinatorial explosion. In contrast, RxDiff achieves holistic combinatorial modeling through bidirectional iterative denoising, while further incorporating an inference-time safety regulation mechanism absent in prior work.

Discrete diffusion models. Denoising diffusion models [15] have emerged as a powerful generative paradigm through iterative corruption and reconstruction [48, 8]. Extending this framework to discrete state spaces, D3PM [2] introduces a general formulation with categorical transition matrices, with subsequent works [17, 33, 26] further improving expressiveness, training stability, and sampling efficiency. [10, 16] explore inference-time control through classifier and classifier-free guidance by steering generation toward desired global attributes through a position-independent conditioning objective. These frameworks have been successfully applied to discrete structured generation across several domains, including text generation [25, 33, 26, 29], molecular generation [42, 24] and protein design [44, 12], demonstrating the paradigm's capacity to model combinatorial dependencies. Despite these advances, discrete diffusion has not, to our knowledge, been explored for medication recommendation, whose pharmacological priors and combinatorial safety constraints pose unique challenges for encoding domain-specific knowledge into the forward process and enforcing dynamic interaction regulations that vary across positions and evolve with the generation state.

3 Problem Formulation

Electronic Health Record (EHR). A patient's medical history is represented as a sequence of clinical visits $\mathcal{V} = [v^{(1)}, v^{(2)}, \ldots, v^{(N)}]$, where N is the total number of visits. Each visit $v^{(i)} = [\mathbf{v}_d^{(i)}, \mathbf{v}_p^{(i)}]$ comprises a multi-hot diagnosis vector $\mathbf{v}_d^{(i)} \in \{0,1\}^{|D|}$ and a multi-hot procedure vector $\mathbf{v}_p^{(i)} \in \{0,1\}^{|P|}$ over diagnosis set D and procedure set P respectively.

Prescription. Let $\mathcal{M} = \{m_1, m_2, \ldots, m_{|\mathcal{M}|}\}$ denote the medication vocabulary. The ground-truth prescription for visit $v^{(N)}$ is represented as a set $X_0 \subseteq \mathcal{M}$ of co-administered medications, where each medication serves as a discrete atomic token.

DDI Matrix. Drug-drug interactions are represented as a symmetric matrix $A \in \{0,1\}^{|\mathcal{M}|\times|\mathcal{M}|}$, where $A_{ij} = 1$ indicates a known harmful interaction between medications m_i and m_j.

Medication Recommendation. Given a patient's longitudinal EHR $\mathcal{V} = [v^{(1)}, \ldots, v^{(N)}]$ and the DDI matrix A, the goal is to recommend an optimal prescription $\hat{X}_0 \approx X_0$ for the current visit $v^{(N)}$.

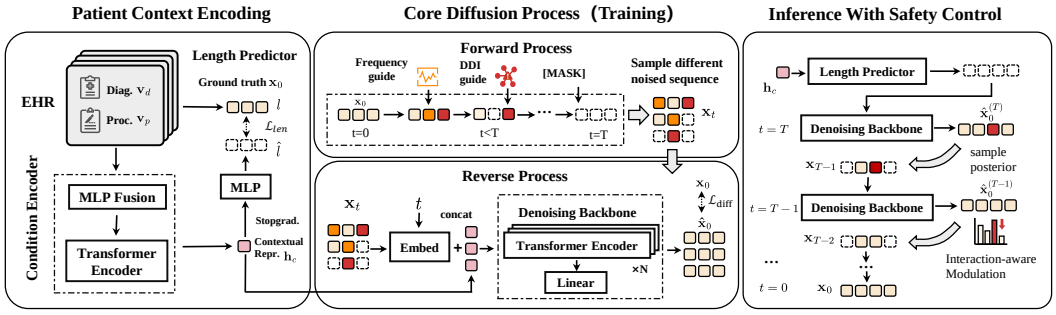

Figure 2: Overview of RxDiff. The condition encoder maps longitudinal EHR to a contextual vector $\mathbf{h}_c$, from which a length predictor estimates prescription size. The forward process corrupts ground-truth prescriptions via frequency- and DDI-guided noise; the denoising backbone learns to recover clean prescriptions from corrupted sequences. At inference, a fully noised sequence is iteratively denoised with interaction-aware modulation applied at each step.

4 Method

As illustrated in Figure 2, RxDiff is organized into three stages: 1) **Patient Context Encoding** (Section 4.2) encodes longitudinal EHR into a contextual vector and estimates prescription length, providing the clinical grounding for all subsequent generation; 2) the **Core Diffusion Process** (Section 4.3), in which a dual-guided forward process injects drug frequency and DDI structure into the corruption dynamics, and a denoising backbone recovers clean prescriptions through bidirectional dependency modeling; 3) **Inference-Time Safety Control** (Section 4.4) applies interaction-aware modulation at each iterative denoising step to impose controllable, per-prescription DDI constraints.

4.1 Prescription Serialization

A prescription X_0 is an unordered set, yet per-token operations require an explicit positional structure. We therefore convert X_0 into a fixed-length sequence $\mathbf{x}_0 \in (\mathcal{M} \cup \{\texttt{PAD}\})^L$ by sorting drug tokens according to their vocabulary indices, establishing a bijective mapping between prescription sets and sequences. This serialization is invertible and preserves the set semantics of the prescription since the original set X_0 is fully recoverable by collecting all non-padding tokens. A binary indicator vector $\mathbf{b} \in \{0,1\}^L$ marks drug positions ($b_i = 1$) and excludes `[PAD]` positions from all computations.

4.2 Patient Context Encoding

Condition Encoder. The condition encoder maps the patient's longitudinal EHR data $\mathcal{V} = [v^{(1)}, \ldots, v^{(N)}]$ to a single contextual vector $\mathbf{h}_c \in \mathbb{R}^d$. For each visit $v^{(i)}$, diagnosis and procedure information are encoded separately and fused:

$$\mathbf{h}_d^{(i)} = \text{MLP}_d(\mathbf{v}_d^{(i)}), \quad \mathbf{h}_p^{(i)} = \text{MLP}_p(\mathbf{v}_p^{(i)}), \quad \mathbf{h}^{(i)} = \text{MLP}_f([\mathbf{h}_d^{(i)}; \mathbf{h}_p^{(i)}]) \tag{1}$$

The sequence of visit representations $[\mathbf{h}^{(1)}, \ldots, \mathbf{h}^{(N)}]$ is then processed by a Transformer encoder [41] to capture longitudinal dependencies across visits:

$$[\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(N)}] = \text{TransformerEncoder}_{\text{visit}}([\mathbf{h}^{(1)}, \ldots, \mathbf{h}^{(N)}]) \tag{2}$$

The output at the final visit is taken as the patient contextual vector $\mathbf{h}_c = \mathbf{u}^{(N)}$.

Length Predictor. At inference time, the number of medications to prescribe, i.e., the number of drug positions in $\mathbf{x}_0$, is unknown. Since the prescription length reflects the complexity of the patient's health context, a length predictor estimates it from $\mathbf{h}_c$:

$$\hat{l} = \arg\max \ \text{MLP}_\phi(\text{stopgrad}(\mathbf{h}_c)) \in \{1, \ldots, L\} \tag{3}$$

The stop-gradient ensures that $\mathbf{h}_c$ is shaped exclusively by the diffusion objective described later.

4.3 Core Diffusion Process

The diffusion process defines a corruption trajectory $\mathbf{x}_0 \to \mathbf{x}_1 \to \cdots \to \mathbf{x}_T$ over the serialized prescription and a learned reverse process that recovers $\mathbf{x}_0$ from noise.

4.3.1 Dual-Guided Forward Process

The forward process should degrade prescriptions toward two targets aligned with the core objectives of medication recommendation. First, the reverse process should recover patient-specific prescriptions from population-level statistics; the forward process should therefore corrupt toward statistically common drug combinations. Second, the reverse process should learn to sense and avoid DDI risk; the forward process should therefore corrupt toward high-interaction combinations. We formalize both through a dual-guided substitution distribution over each drug token:

$$r(m \mid x_i^0,\, \{x_j^0\}_{j\neq i}) \propto \text{freq}(m) \cdot \exp\left(\tau \cdot \text{conf}(m, \{x_j^0\}_{j\neq i})\right), \quad m \in \mathcal{M} \setminus \{x_i^0\} \tag{4}$$

where x_i^0 denotes the token at position i of $\mathbf{x}_0$, $\text{freq}(m)$ denotes the empirical frequency of medication m in the training corpus, and $\text{conf}(m, \{x_j^0\}_{j\neq i}) = \sum_{j\neq i} A_{m,x_j^0}$ counts the DDI conflicts that drug m would introduce against the remaining clean prescription. The temperature $\tau \geq 0$ controls the strength of the DDI prior; at $\tau = 0$ the distribution reduces to frequency-only substitution.

However, substitution alone is insufficient: substituted tokens retain concrete drug identities, preventing the forward process from reaching complete information loss. Since $\mathbf{x}_0$ is unavailable at inference time, generation requires a prior $q(\mathbf{x}_T \mid \mathbf{x}_0)$ independent of $\mathbf{x}_0$. We therefore introduce a `[MASK]` token as an absorbing state, extending the vocabulary to $\mathcal{M}^+ = \mathcal{M} \cup \{\texttt{MASK}, \texttt{PAD}\}$, and combine substitution and masking noise with $\gamma \in [0, 1)$ controlling their balance.

When $\tau > 0$, the substitution distribution r depends on the clean context $\{x_j^0\}_{j\neq i}$. Since this context is fully observed during training, the forward transitions remain conditionally independent across positions given $\mathbf{x}_0$. However, directly computing the cumulative marginal under a general context-dependent substitution kernel would require repeated multiplication of dense, position-specific transition matrices, which is computationally expensive. We therefore impose a semi-absorbing constraint: each token may be substituted at most once, after which it can only be retained or transition to `[MASK]`. This sparse transition structure admits an analytic cumulative marginal while preserving the context-dependent frequency and DDI priors. The resulting per-token transition is:

$$q(x_i^t \mid x_i^{t-1}, \{x_j^0\}_{j\neq i}) = \begin{cases} \alpha_t \mathbf{1}[x_i^t = x_i^0] + \beta_t(1-\gamma)\mathbf{1}[x_i^t = \texttt{MASK}] & \\ \quad + \beta_t\gamma \sum_{m\in\mathcal{M}} r(m \mid x_i^0,\, \{x_j^0\}_{j\neq i})\mathbf{1}[x_i^t = m], & x_i^{t-1} = x_i^0 \\ (\alpha_t + \beta_t\gamma)\mathbf{1}[x_i^t = x_i^{t-1}] + \beta_t(1-\gamma)\mathbf{1}[x_i^t = \texttt{MASK}], & x_i^{t-1} \in \mathcal{M} \setminus \{x_i^0\} \\ \mathbf{1}[x_i^t = \texttt{MASK}], & x_i^{t-1} = \texttt{MASK} \end{cases} \tag{5}$$

where $\alpha_t = 1 - \beta_t$ with β_t following a cosine schedule. The three cases correspond to transitions from the original, substituted, and masked states respectively. The enlarged retention probability $\alpha_t + \beta_t\gamma$ in the second case absorbs the would-be second substitution to enforce the semi-absorbing constraint. Formal proofs of conditional independence, closed-form marginals, and terminal convergence are provided in Appendix A.

4.3.2 Reverse Process

The reverse process learns to recover $\mathbf{x}_0$ from a noised sequence $\mathbf{x}_t$ by iteratively sampling from the posterior $q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0)$. By Bayes' rule and the conditional independence established in Appendix A, the posterior factorizes per-token as:

$$q(x_i^{t-1} \mid x_i^t, x_i^0, \{x_j^0\}_{j\neq i}) \propto q(x_i^t \mid x_i^{t-1}, \{x_j^0\}_{j\neq i}) \cdot q(x_i^{t-1} \mid x_i^0, \{x_j^0\}_{j\neq i}) \tag{6}$$

Since $\mathbf{x}_0$ is unavailable at inference time, we adopt the x_0-parameterization [2], in which the denoising network f_θ directly predicts the distribution over clean tokens at each position:

$$p_\theta(x_i^0 \mid \mathbf{x}_t, t, \mathbf{h}_c) = \text{softmax}(f_\theta(\mathbf{x}_t, t, \mathbf{h}_c))_i \tag{7}$$

The posterior is then approximated by substituting the model prediction $\hat{\mathbf{x}}_0 = \arg\max\, p_\theta(x_i^0 \mid \mathbf{x}_t, t, \mathbf{h}_c)$ into Eq. (6). At the final step $t = 1$, $\hat{\mathbf{x}}_0$ is returned directly. All operations are restricted to positions with $b_i = 1$; `[PAD]` positions are carried through unchanged.

4.3.3 Denoising Backbone

The denoising network f_θ instantiates Eq. (7). It takes the noised sequence $\mathbf{x}_t$, timestep t, and contextual vector $\mathbf{h}_c$ as input, and outputs logits over $\mathcal{M}$ at each drug position. Each position is represented by sum of a token embedding, a learned positional embedding, and a projected sinusoidal timestep embedding:

$$\mathbf{z}_i = \text{Emb}(x_i^t) + \text{PosEmb}_i + \text{MLP}_t(\text{SinEmb}(t)), \quad i = 1, \ldots, L \tag{8}$$

Positional embeddings only encode position identity here, enabling the model to distinguish drug positions for per-token denoising. To inject patient context, $\mathbf{h}_c$ is prepended as a virtual condition token at position zero, forming an extended sequence $\tilde{\mathbf{z}} = [\mathbf{h}_c; \mathbf{z}_1, \ldots, \mathbf{z}_L] \in \mathbb{R}^{(L+1)\times d}$. This sequence is processed by an N_{layer}-layer pre-norm Transformer encoder:

$$\tilde{\mathbf{z}}^{(\ell+1)} = \text{TransformerLayer}\left(\tilde{\mathbf{z}}^{(\ell)}\right), \quad \ell = 0, \ldots, N_{\text{layer}} - 1 \tag{9}$$

After the final layer, the condition token is discarded and the drug-position representations are projected to logits over $\mathcal{M}$. Specifically, for position i with $b_i = 1$, the logit for drug m is:

$$z_{i,m} = \mathbf{W}_{\text{out}}\, \tilde{\mathbf{z}}_i^{(N_{\text{layer}})} + \mathbf{b}_{\text{out}} \tag{10}$$

where $\tilde{\mathbf{z}}_i^{(N_{\text{layer}})} \in \mathbb{R}^d$ is the final-layer representation at position i and $\mathbf{W}_{\text{out}} \in \mathbb{R}^{|\mathcal{M}|\times d}$, $\mathbf{b}_{\text{out}} \in \mathbb{R}^{|\mathcal{M}|}$ are learnable parameters. `[MASK]` and `[PAD]` are excluded from the output space.

4.3.4 Training Objective

Diffusion loss. Given a training pair $(X_0, \mathcal{V})$, a timestep $t \sim \mathcal{U}\{1, T\}$ is sampled and $\mathbf{x}_t$ is drawn from the closed-form cumulative marginal $q(\mathbf{x}_t \mid \mathbf{x}_0)$ derived in Appendix A. The denoising backbone is trained to predict $\mathbf{x}_0$ directly via cross-entropy over drug positions:

$$\mathcal{L}_{\text{diff}} = -\sum_{i:\, b_i=1} \log p_\theta(x_i^0 \mid \mathbf{x}_t, t, \mathbf{h}_c) = -\sum_{i:\, b_i=1} \left[z_{i,\, x_i^0} - \log \sum_{m\in\mathcal{M}} \exp(z_{i,m}) \right] \tag{11}$$

Length loss. The length predictor is trained via cross-entropy against the ground-truth length $|X_0|$:

$$\mathcal{L}_{\text{len}} = -\log \text{MLP}_\phi(\text{stopgrad}(\mathbf{h}_c))[|X_0|] \tag{12}$$

The full training loss is $\mathcal{L} = \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{len}}$. All parameters are updated jointly in a single training loop.

4.4 Inference with Safety Control

Initialization. Given a patient's longitudinal EHR $\mathcal{V}$, the condition encoder first computes $\mathbf{h}_c$. The length predictor then estimates the prescription length $\hat{l} = \arg\max \text{MLP}_\phi(\mathbf{h}_c)$, and the reverse process is initialized with $\hat{l}$ `[MASK]` tokens padded to length L, forming $\mathbf{x}_T$.

Reverse sampling. The reverse process iterates from $t = T$ to $t = 1$. At each step, the denoising backbone predicts $\hat{\mathbf{x}}_0$ via Eq. (7), and $\mathbf{x}_{t-1}$ is sampled per-token from the posterior in Eq. (6) with $\hat{\mathbf{x}}_0$ substituted for $\mathbf{x}_0$. At t=1, $\hat{\mathbf{x}}_0$ is returned directly.

Interaction-aware modulation. While the forward process encodes DDI structure into the model's learned distribution, training-time constraints alone cannot guarantee interaction safety for individual inference instances. We therefore apply interaction-aware modulation to the denoiser logits at every reverse step t. Let $z_{i,m}^{(t)}$ denote the logit for drug m at position i at step t. The adjusted logit is:

$$\tilde{z}_{i,m}^{(t)} = z_{i,m}^{(t)} - \mu \sum_{j\neq i,\, b_j=1} A_{m,\, \hat{x}_j^{0,(t+1)}} \tag{13}$$

where $\mu \geq 0$ controls the constraint strength, and $\hat{\mathbf{x}}_0^{(t+1)}$ is the clean prescription predicted at the previous step $t+1$, with no modulation applied at $t = T$. The penalty $\sum_j A_{m,\hat{x}_j^{0,(t+1)}}$ accumulates the number of known interactions between candidate drug m and all other predicted drugs, suppressing high-DDI candidates proportionally to their conflict count. Applied at each step, this modulation cumulatively steers away from high-DDI regions in an instance-level and controllable manner, complementing the internalized DDI awareness from training.

Table 1: Prescription generation results on MIMIC-III and MIMIC-IV. All metrics except #Meds are reported in %. Best results are in **bold** (statistically significant with $p < 0.05$ under paired t-test).

	MIMIC-III					MIMIC-IV				
Model	Jaccard↑	F1↑	PRAUC↑	DDI↓	#Meds	Jaccard↑	F1↑	PRAUC↑	DDI↓	#Meds
LR	43.16 (0.21)	57.24 (0.18)	64.01 (0.25)	8.10 (0.09)	19.89 (0.23)	35.99 (0.22)	49.01 (0.25)	56.87 (0.31)	8.42 (0.10)	11.12 (0.17)
LEAP	44.58 (0.26)	58.41 (0.23)	65.07 (0.28)	6.72 (0.12)	19.12 (0.24)	37.18 (0.24)	49.86 (0.27)	57.73 (0.29)	6.85 (0.14)	10.78 (0.20)
GAMENet	46.06 (0.19)	59.83 (0.22)	66.12 (0.31)	7.84 (0.11)	20.53 (0.19)	38.27 (0.18)	50.74 (0.20)	58.94 (0.26)	8.01 (0.13)	11.43 (0.21)
SafeDrug	46.72 (0.24)	60.59 (0.28)	67.24 (0.27)	5.91 (0.08)	19.94 (0.22)	39.04 (0.25)	51.63 (0.23)	60.35 (0.28)	5.57 (0.09)	11.28 (0.19)
4SDrug	47.03 (0.22)	60.94 (0.25)	67.82 (0.26)	6.05 (0.10)	18.73 (0.19)	39.25 (0.23)	52.07 (0.21)	61.24 (0.24)	5.67 (0.11)	10.56 (0.18)
MoleRec	47.28 (0.16)	61.37 (0.21)	68.31 (0.29)	6.47 (0.10)	20.15 (0.24)	39.46 (0.20)	52.41 (0.18)	61.78 (0.22)	6.23 (0.11)	11.61 (0.20)
COGNet	46.95 (0.20)	60.91 (0.24)	67.73 (0.28)	6.38 (0.09)	20.08 (0.21)	39.21 (0.19)	51.98 (0.22)	61.12 (0.25)	6.39 (0.10)	11.35 (0.20)
REFINE	47.42 (0.18)	61.68 (0.22)	68.74 (0.26)	5.88 (0.11)	20.24 (0.22)	39.63 (0.21)	52.79 (0.19)	62.41 (0.23)	5.41 (0.09)	11.74 (0.21)
SubRec	47.64 (0.23)	61.84 (0.19)	69.12 (0.24)	6.70 (0.12)	20.34 (0.20)	39.72 (0.17)	53.08 (0.21)	63.04 (0.19)	5.98 (0.08)	11.89 (0.22)
RxDiff	**48.49** (0.17)	**62.57** (0.24)	**70.59** (0.20)	**4.19** (0.07)	19.74 (0.21)	**40.39** (0.12)	**54.12** (0.16)	**64.35** (0.20)	**4.02** (0.04)	11.65 (0.18)

5 Experiment

5.1 Experimental Setup

Datasets. We evaluate on three datasets spanning different clinical settings and patient populations. MIMIC-III [19] and MIMIC-IV [18] are large-scale de-identified inpatient EHR databases. The third is a real-world general inpatient dataset collected from a tertiary hospital in China with a different clinical profile from the ICU-focused MIMIC benchmarks. The real-world dataset is fully de-identified and used under institutional data governance. We follow the preprocessing pipeline of GAMENet [35] and SafeDrug [47], while retaining medication granularity at ATC level-4 to preserve finer-grained therapeutic distinctions [27]. All datasets are split at the patient level into train/validation/test sets at a ratio of 4:1:1. Detailed dataset statistics are provided in Appendix B.1.

Baselines. We compare RxDiff against state-of-the-art medication recommendation methods: LEAP [51], GAMENet [35], SafeDrug [47], MoleRec [49], 4SDrug [39], COGNet [46], REFINE [5], and SubRec [11]. We additionally include logistic regression (LR) as a simple reference. Detailed descriptions of all baselines are provided in Appendix B.2.

Evaluation Metrics. We report Jaccard similarity, F1 score, and PRAUC to measure recommendation accuracy, DDI rate to measure population-level interaction safety, and average number of prescribed medications (Avg. #Meds) to assess prescription length calibration. Formal definitions of all metrics are provided in Appendix B.3.

Implementation Details. All baseline models are trained with their originally reported hyperparameters. For RxDiff, the diffusion process uses $T = 100$ steps with a cosine noise schedule, substitution ratio $\gamma = 0.2$, DDI noise temperature $\tau = 1.5$, and interaction-aware modulation strength μ selected as the largest value incurring less than 1% absolute degradation in Jaccard similarity on the validation set. Full hyperparameter settings and model implementation details are provided in Appendix B.4.

5.2 Main Results

Table 1 reports results on MIMIC-III and MIMIC-IV. RxDiff consistently achieves the highest recommendation accuracy across both datasets. By modeling drug combination dependencies through iterative denoising and capturing patient-specific prescription patterns through clinically-guided noise design, RxDiff produces more precise recommendations than existing methods. In terms of interaction safety, RxDiff achieves the lowest DDI rate on both datasets, reducing DDI by 28.7% on MIMIC-III and 25.7% on MIMIC-IV compared to REFINE, the strongest DDI-aware baseline. Notably, RxDiff achieves simultaneous improvements in both recommendation accuracy and interaction safety—an objective that proves challenging within existing frameworks, where accuracy-oriented methods such as SubRec

Table 2: Results on the real-world general inpatient dataset. Metrics in % except #Meds. Best in **bold**.

Model	Jaccard ↑	F1↑	PRAUC↑	DDI↓	#Meds
LR	47.12 (0.24)	61.98 (0.27)	69.23 (0.33)	6.71 (0.11)	20.92
LEAP	48.23 (0.26)	62.74 (0.24)	70.08 (0.29)	5.24 (0.13)	20.21
GAMENet	48.91 (0.20)	63.41 (0.23)	70.89 (0.28)	6.41 (0.14)	21.31
SafeDrug	49.74 (0.18)	64.52 (0.25)	72.31 (0.30)	4.33 (0.09)	21.05
4SDrug	50.08 (0.20)	64.93 (0.23)	72.87 (0.25)	4.47 (0.10)	19.86
COGNet	50.15 (0.21)	65.10 (0.22)	73.20 (0.27)	4.62 (0.10)	21.12
MoleRec	50.63 (0.22)	65.81 (0.19)	74.12 (0.26)	4.58 (0.10)	21.47
REFINE	50.79 (0.19)	66.18 (0.20)	75.31 (0.24)	4.21 (0.09)	21.52
SubRec	50.95 (0.21)	66.52 (0.22)	75.84 (0.23)	4.79 (0.11)	21.58
RxDiff	**51.38** (0.15)	**67.14** (0.21)	**76.78** (0.18)	**3.26** (0.05)	20.94

tend to exhibit suboptimal DDI control. The average number of prescribed medications predicted by RxDiff closely matches the ground-truth across both datasets, confirming that the length predictor accurately captures the relationship between clinical complexity and medication count.

Table 2 reports results on the real-world general inpatient dataset, which differs substantially from the MIMIC benchmarks in clinical profile with far fewer diagnoses per visit while maintaining a comparable average medication count, posing a non-trivial distribution shift. Despite this, RxDiff maintains consistent advantages in both recommendation accuracy and interaction safety, achieving the highest Jaccard, F1, and PRAUC scores alongside the lowest DDI rate. Compared to REFINE, the strongest DDI-aware baseline on this dataset, RxDiff further reduces DDI rate by 22.6%, demonstrating that the dual-stage interaction constraint generalizes effectively across clinically distinct settings.

5.3 Safety Analysis

Component ablation. We evaluate four configurations of the dual-stage interaction constraint in Figure 3(a): without any DDI constraint (w/o DDI), with inference-time modulation only (Inf. only), with DDI-guided forward process only (Train. only), and the full model combining both stages (Full). Removing all DDI constraints yields the highest accuracy but also the highest DDI rate, confirming that accurate recommendation does not inherently minimize interaction risk. Inference-time modulation alone reduces DDI but incurs a moderate accuracy cost due to the lack of internalized DDI awareness, while the DDI-guided forward process alone achieves greater DDI reduction with less accuracy degradation by internalizing DDI awareness during training. The full model attains the lowest DDI rate on both tasks while maintaining competitive accuracy, demonstrating that the two stages are effective and complementary.

Pareto analysis. Figure 3(b) traces the F1–DDI trade-off as μ increases from 0 to 1.0 (in steps of 0.2) under different values of τ. At $\tau = 0$, the model has no internalized DDI awareness and exhibits the steepest F1 decline per unit DDI reduction, as the entire safety burden falls on inference-time modulation. All $\tau > 0$ curves extend to lower DDI rates than $\tau = 0$, while maintaining higher F1 at comparable DDI levels. Among the $\tau > 0$ configurations, $\tau = 1.5$ achieves the most favorable trade-off: smaller values provide insufficient DDI guidance during training, while larger values over-constrain the noise distribution and depress the F1 starting point.

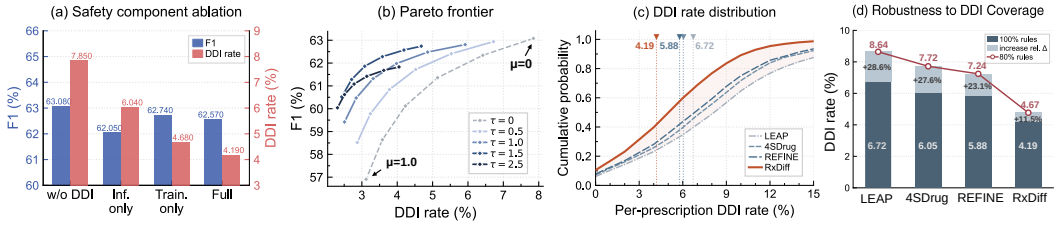

Figure 3: Safety analysis. (a) Ablation of safety components. (b) F1–DDI trade-off under varying τ and μ. (c) Per-prescription DDI distribution (CDF). (d) Robustness under reduced DDI coverage.

Individual-level DDI analysis. The aggregate DDI rate captures population-averaged safety but conceals the per-prescription distribution that ultimately matters for each patient. Figure 3(c) compares the empirical CDF of per-prescription DDI rates on MIMIC-III. RxDiff's CDF lies strictly above all baselines across the entire DDI range, raising the zero-DDI fraction to 10.5% while reducing the fraction of high-risk prescriptions (DDI > 0.1) from $20-30\%$ across baselines to 10.8%. This indicates that RxDiff not only generates a larger share of fully safe prescriptions, but more importantly, substantially reduces the occurrence of high-risk cases that pose greater clinical concern. The uniform distributional improvement, rather than a localized trade-off, suggests that inference-time modulation can effectively complement training-time mechanisms in providing per-prescription safety control beyond what aggregate optimization typically delivers.

Robustness to evolving DDI knowledge. In clinical deployment, the DDI knowledge base continuously expands as new interactions are identified. To evaluate robustness to such setting, we randomly exclude 20% of the DDI rules from training and evaluate all methods against the complete DDI rule set at inference, comparing against the fully-trained counterparts. As shown in Figure 3(d), baseline methods exhibit a substantial increase in DDI rate $(21-29\%)$ under reduced training coverage, as

their parametric priors cannot effectively suppress DDI patterns unseen during training. In contrast, RxDiff shows only a modest change (11.5%), as its inference-time modulation directly queries the current DDI rules rather than relying on parametric memory. This suggests that inference-time intervention offers a capability largely independent of training coverage, supporting deployment scenarios where the safety knowledge base evolves without requiring model retraining.

5.4 Ablation Study and Sensitivity Analysis

Forward process design. We evaluate the contribution of each component of the dual-guided substitution distribution through a 2×2 factorial ablation, varying whether the frequency prior and DDI guidance are included: no guidance (Uniform freq., $\tau = 0$), frequency-guided only (Freq. only, $\tau = 0$), DDI-guided only (DDI only, uniform freq., $\tau > 0$), and the full dual-guided design (Freq.+DDI). Results are reported in Figure 4(a).

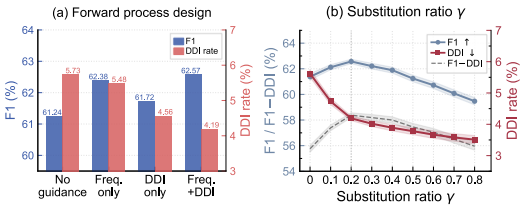

Figure 4: Ablation and sensitivity studies on MIMIC-III. (a) 2×2 factorial ablation of the forward process. (b) Effect of substitution ratio γ; the composite metric F1$-$DDI (dashed) is maximized at γ=0.2 (vertical line).

Removing both components (No guidance) yields the weakest performance on both recommendation accuracy and DDI rate, confirming that unstructured noise provides insufficient training signal for either objective. Adding the frequency prior (Freq. only) improves recommendation accuracy, as the model learns to recover patient-specific prescriptions from population-level statistics, but provides limited DDI reduction since no interaction structure is encoded in the forward process. Adding DDI guidance instead (DDI only) achieves stronger DDI reduction, as the forward process explicitly exposes the model to high-risk drug combinations during training, but yields lower recommendation accuracy compared to Freq. only with frequency prior absent. The full design achieves the best performance on both metrics, demonstrating that the frequency prior and DDI guidance address complementary aspects of the recommendation objective and are most effective in combination.

Substitution ratio γ. The parameter $\gamma \in [0, 1)$ controls the balance between substitution and masking noise in the forward process. Figure 4(b) reports F1 and DDI rate as γ varies across $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$. At $\gamma = 0$, the forward process reduces to pure masking noise. As γ increases, F1 first improves then degrades at large γ, while DDI rate decreases monotonically. The composite metric F1 - DDI is optimized at γ = 0.2, which is adopted as the default setting.

The non-monotonic behavior of F1 reveals two complementary roles of substitution-based noise. Unlike pure masking noise, which causes complete information loss at corrupted positions, substitution-based noise replaces them with concrete drug tokens, preventing the model from taking shortcuts by inferring missing medications from neighboring drug positions rather than reasoning from patient context. This forces the model to ground its predictions in patient-specific clinical information. Second, since substituted drugs consider population-level frequency statistics, the model must learn to distinguish between what is statistically common and what is optimal for the individual patient, directly strengthening patient-specific discriminative ability. At large γ, however, excessive substitution suppresses the masking signal that is essential for generation from pure noise, leading to performance degradation. The optimal $\gamma = 0.2$ balances these competing effects.

Controlled inference-time DDI comparison. To examine whether RxDiff's safety advantage merely stems from accessing DDI knowledge during inference, we equip the discriminative baselines REFINE [5] and SubRec [11] with DDI-based logit penalties and the autoregressive baseline COGNet [46] with DDI-aware greedy decoding. Both approaches adjust the original medication logits by $\widetilde{z}_m = z_m - \mu r_m$, but differ in their estimation of the interaction risk:

$$r_m^{\text{disc}} = \frac{\sum_{j \neq m} A_{m,j} p_j}{\sum_{j \neq m} p_j + \epsilon}, \qquad r_{m,t}^{\text{AR}} = \frac{\sum_{j \in S_{t-1}} A_{m,j}}{\max(1, |S_{t-1}|)}. \tag{14}$$

For discriminative models, $p_j = \sigma(z_j)$ denotes the predicted probability of medication j, and $\epsilon > 0$ is a small constant for numerical stability. For autoregressive decoding, S_{t-1} denotes the previously generated medication set, and the penalty is applied at each generation step before greedy selection.

All baselines access the same DDI matrix as RxDiff during inference, without additional training. The penalty strength μ is varied separately for each method, as its scale depends on the underlying prediction paradigm.

Table 3 compares the methods under varying inference-time DDI control strengths. At a matched DDI rate of approximately 4.3%, RxDiff achieves 48.52% Jaccard, outperforming the controlled REFINE (44.18%), SubRec (43.76%), and COGNet (44.91%) by 4.34, 4.76, and 3.61 percentage points, respectively. Increasing the modulation strength further reduces RxDiff's DDI rate to 3.27% while retaining 47.66% Jaccard. In contrast, stronger penalties substantially degrade the accuracy of discriminative baselines, accompanied by a pronounced reduction in predicted prescription lengths. This is because post-hoc logit suppression directly removes potentially interacting medications without reconsidering their therapeutic relevance. Autoregressive decoding exhibits a milder decline by incorporating DDI constraints progressively, but its sequential decisions cannot be revised once generated. In contrast, RxDiff evaluates interaction risks against the current full-prescription estimate and iteratively revises medication choices throughout reverse denoising. This revisable, prescription-level regulation enables a more favorable safety–accuracy trade-off, demonstrating that RxDiff's advantage extends beyond merely accessing DDI knowledge during inference.

Table 3: Inference-time DDI control on MIMIC-III. Metrics are reported in % except #Meds.

Model	μ	Jacc.↑	F1↑	DDI↓	#Meds
REFINE	0.0	47.42 (0.18)	61.68 (0.22)	5.88 (0.11)	20.24 (0.22)
	0.4	46.53 (0.22)	60.62 (0.25)	5.09 (0.10)	18.22 (0.24)
	0.8	44.18 (0.24)	57.93 (0.28)	4.34 (0.09)	16.31 (0.26)
SubRec	0.0	47.64 (0.23)	61.84 (0.19)	6.70 (0.12)	20.34 (0.20)
	0.5	46.21 (0.21)	60.15 (0.22)	5.21 (0.10)	18.39 (0.22)
	1.0	43.76 (0.25)	57.38 (0.27)	4.30 (0.09)	15.84 (0.25)
COGNet	0.0	46.95 (0.20)	60.91 (0.24)	6.38 (0.09)	20.08 (0.21)
	0.5	46.12 (0.22)	59.92 (0.23)	5.61 (0.08)	19.12 (0.20)
	2.0	44.91 (0.21)	58.51 (0.25)	4.43 (0.09)	17.76 (0.23)
RxDiff	0.2	**48.52** (0.17)	**62.68** (0.24)	4.29 (0.07)	19.82 (0.21)
	0.6	47.66 (0.18)	61.81 (0.22)	**3.27** (0.06)	18.95 (0.22)

6 Conclusion

We reformulate medication recommendation as a discrete diffusion process and propose RxDiff, a framework that addresses the inadequate combinatorial modeling and the absence of inference-time safety control in existing methods. By framing prescription generation as iterative denoising, RxDiff enables bidirectional dependency modeling across all drugs at every generation step, while naturally admitting dynamic constraints throughout the reverse trajectory. Two designs tailor the framework to clinical needs. A dual-guided forward process encodes drug frequency and DDI structure into the corruption dynamics for training-time interaction awareness. An inference-time modulation mechanism then regulates DDI risk at each denoising step. Experiments on MIMIC-III, MIMIC-IV, and a real-world general inpatient dataset demonstrate consistent accuracy improvements, lower average DDI rates, more favorable individual-level DDI distributions, and robustness under reduced DDI coverage. We hope this work encourages further exploration of generative paradigms that incorporate inference-time safety control into clinical decision support.

7 Acknowledgment

This paper is partially supported by the National Natural Science Foundation of China (No.12227901), Anhui Science Foundation for Distinguished Young Scholars (No.1908085J24) and Natural Science Foundation of China (No.62502491). The AI-driven experiments, simulations and model training were performed on the robotic AI-Scientist platform of Chinese Academy of Sciences.

References

[1] Adekunle Oyeyemi Adeniyi, Jeremiah Olawumi Arowoogun, Rawlings Chidi, Chioma Anthonia Okolo, and Oloruntoba Babawarun. The impact of electronic health records on patient care and outcomes: A comprehensive review. *World Journal of Advanced Research and Reviews*, 21(2):1446–1455, 2024.

[2] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

[3] Sanjeet Bagcchi. Who's global tuberculosis report 2022. *The Lancet Microbe*, 4(1):e20, 2023.

[4] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

[5] Suman Bhoi, Mong Li Lee, Wynne Hsu, and Ngiap Chuan Tan. Refine: A fine-grained medication recommendation system using deep learning and personalized drug interaction modeling. *Advances in Neural Information Processing Systems*, 36:24013–24024, 2023.

[6] Karen Bush and Patricia A Bradford. β-lactams and β-lactamase inhibitors: an overview. *Cold Spring Harbor perspectives in medicine*, 6(8):a025247, 2016.

[7] Deborah J Cook, Hugh D Fuller, Gordon H Guyatt, John C Marshall, David Leasa, Richard Hall, Timothy L Winton, Frank Rutledge, Thomas Todd, Peter Roy, et al. Risk factors for gastrointestinal bleeding in critically ill patients. *New England journal of medicine*, 330(6):377–381, 1994.

[8] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9):10850–10869, 2023.

[9] Ian H de Boer, Kamlesh Khunti, Tami Sadusky, Katherine R Tuttle, Joshua J Neumiller, Connie M Rhee, Sylvia E Rosas, Peter Rossing, and George Bakris. Diabetes management in chronic kidney disease: a consensus report by the american diabetes association (ada) and kidney disease: Improving global outcomes (kdigo). *Kidney international*, 102(5):974–989, 2022.

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[11] Wenjie Du, Xuqiang Li, Jinke Feng, Shuai Zhang, Wen Zhang, and Yang Wang. Integrating drug substructures and longitudinal electronic health records for personalized drug recommendation. *Advances in Neural Information Processing Systems*, 38:62950–62977, 2026.

[12] Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

[13] Morten L Hansen, Rikke Sørensen, Mette T Clausen, Marie Louise Fog-Petersen, Jakob Raunsø, Niels Gadsbøll, Gunnar H Gislason, Fredrik Folke, Søren S Andersen, Tina K Schramm, et al. Risk of bleeding with single, dual, or triple therapy with warfarin, aspirin, and clopidogrel in patients with atrial fibrillation. *Archives of internal medicine*, 170(16):1433–1441, 2010.

[14] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[17] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in neural information processing systems*, 34:12454–12465, 2021.

[18] Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

[19] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[20] Zhiqiang Kou, Jing Wang, Yuheng Jia, Biao Liu, and Xin Geng. Instance-dependent inaccurate label distribution learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(1):1425–1437, 2025.

[21] Zhiqiang Kou, Jing Wang, Jiawei Tang, Yuheng Jia, Boyu Shi, and Xin Geng. Exploiting multi-label correlation in label distribution learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 4326–4334, 8 2024.

[22] Zhiqiang Kou, Yucheng Xie, Hailin Wang, Jing Wang, Mingkun Xie, Shuo Chen, Yuheng Jia, Tongliang Liu, and Xin Geng. Rankmatch: A novel approach to semi-supervised label distribution learning leveraging rank correlation between labels. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS 2025)*, San Diego, California, USA, December 2025. Neural Information Processing Systems Foundation.

[23] Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182, 2018.

[24] Seul Lee, Karsten Kreis, Srimukh Prasad Veccham, Meng Liu, Danny Reidenbach, Yuxing Peng, Saee Gopal Paliwal, Weili Nie, and Arash Vahdat. Genmol: A drug discovery generalist with discrete diffusion. In *International Conference on Machine Learning*, pages 33205–33226. PMLR, 2025.

[25] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35:4328–4343, 2022.

[26] Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

[27] Gerhard Nahler. Anatomical therapeutic chemical classification system (atc). In *Dictionary of pharmaceutical medicine*, pages 8–8. Springer, 2009.

[28] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[29] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

[30] Jonathan Pearson-Stuttard, Majid Ezzati, and Edward W Gregg. Multimorbidity—a defining challenge for health systems. *The Lancet Public Health*, 4(12):e599–e600, 2019.

[31] Dima M Qato, Jocelyn Wilder, L Philip Schumm, Victoria Gillet, and G Caleb Alexander. Changes in prescription and over-the-counter medication and dietary supplement use among older adults in the united states, 2005 vs 2011. *JAMA internal medicine*, 176(4):473–482, 2016.

[32] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

[33] Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

[34] Juliana Schneider, Engi Abd Elhady Algharably, Andrea Budnick, Arlett Wenzel, Dagmar Draeger, and Reinhold Kreutz. High prevalence of multimorbidity and polypharmacy in elderly patients with chronic pain receiving home care are associated with multiple medication-related problems. *Frontiers in pharmacology*, 12:686990, 2021.

[35] Junyuan Shang, Cao Xiao, Tengfei Ma, Hongyan Li, and Jimeng Sun. Gamenet: Graph augmented memory networks for recommending medication combination. In *proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1126–1133, 2019.

[36] Yun Shen, Jiamin Yu, Jian Zhou, and Gang Hu. Twenty-five years of evolution and hurdles in electronic health records and interoperability in medical research: comprehensive review. *Journal of Medical Internet Research*, 27:e59024, 2025.

[37] Søren T Skou, Frances S Mair, Martin Fortin, Bruce Guthrie, Bruno P Nunes, J Jaime Miranda, Cynthia M Boyd, Sanghamitra Pati, Sally Mtenga, and Susan M Smith. Multimorbidity. *Nature reviews Disease primers*, 8(1):48, 2022.

[38] Mitchell Stern, William Chan, Jamie Kiros, and Jakob Uszkoreit. Insertion transformer: Flexible sequence generation via insertion operations. In *International Conference on Machine Learning*, pages 5976–5985. PMLR, 2019.

[39] Yanchao Tan, Chengjun Kong, Leisheng Yu, Pan Li, Chaochao Chen, Xiaolin Zheng, Vicki S Hertzberg, and Carl Yang. 4sdrug: Symptom-based set-to-set small and safe drug recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3970–3980, 2022.

[40] Nicholas P Tatonetti, Patrick P Ye, Roxana Daneshjou, and Russ B Altman. Data-driven prediction of drug effects and interactions. *Science translational medicine*, 4(125):125ra31–125ra31, 2012.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[42] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.

[43] Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*, 2015.

[44] Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

[45] Christopher JM Whitty, Carrie MacEwen, Andrew Goddard, Derek Alderson, Martin Marshall, Catherine Calderwood, Frank Atherton, Michael McBride, John Atherton, Helen Stokes-Lampard, et al. Rising to the challenge of multimorbidity, 2020.

[46] Rui Wu, Zhaopeng Qiu, Jiacheng Jiang, Guilin Qi, and Xian Wu. Conditional generation net for medication recommendation. In *Proceedings of the ACM web conference 2022*, pages 935–945, 2022.

[47] Chaoqi Yang, Cao Xiao, Fenglong Ma, Lucas Glass, and Jimeng Sun. Safedrug: Dual molecular graph encoders for recommending effective and safe drug combinations. *arXiv preprint arXiv:2105.02711*, 2021.

[48] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023.

[49] Nianzu Yang, Kaipeng Zeng, Qitian Wu, and Junchi Yan. Molerec: Combinatorial drug recommendation with substructure-aware molecular representation learning. In *Proceedings of the ACM Web Conference 2023*, pages 4075–4085, 2023.

[50] Shuai Zhang, Junfeng Fang, Xuqiang Li, Hongxin Xiang, Jun Xia, Ye Wei, Wenjie Du, and Yang Wang. Iterative substructure extraction for molecular relational learning with interactive graph information bottleneck. In *International Conference on Learning Representations*, volume 2025, pages 63678–63706, 2025.

[51] Yutao Zhang, Robert Chen, Jie Tang, Walter F Stewart, and Jimeng Sun. Leap: learning to prescribe effective and safe treatment combinations for multimorbidity. In *proceedings of the 23rd ACM SIGKDD international conference on knowledge Discovery and data Mining*, pages 1315–1324, 2017.

A Theoretical Analysis

Throughout, for clarity we use $r(\cdot \mid x_i^0, \mathbf{x}_0^{-i})$ to denote the substitution distribution over $\mathcal{M} \setminus \{x_i^0\}$, where $\mathbf{x}_0^{-i} \equiv \{x_j^0\}_{j \neq i}$ denotes the clean tokens at all positions other than i.

The dual-guided forward process introduced in Section 4.3 departs from standard discrete diffusion in two respects: the substitution distribution at each position is conditioned on the surrounding clean prescription, and substituted tokens evolve under semi-absorbing rather than fully Markovian dynamics. This appendix establishes three properties on which the validity of both training and inference-time safety control rests:

- **(P1)** The joint distribution $q(\mathbf{x}_t \mid \mathbf{x}_0)$ factorizes across positions, permitting direct sampling of $\mathbf{x}_t$ at arbitrary t without simulating intermediate steps.
- **(P2)** Both the cumulative marginal $q(x_i^t \mid x_i^0, \mathbf{x}_0^{-i})$ and the per-position reverse posterior admit closed-form expressions evaluable in $O(|\mathcal{M}|)$ per position.
- **(P3)** The terminal marginal $q(\mathbf{x}_T \mid \mathbf{x}_0)$ converges to a distribution independent of $\mathbf{x}_0$, ensuring that generation from the prior is well-posed.

Section A.1 establishes (P1) and (P2) through Proposition 1, Lemma 2, Theorem 3, and Proposition 4. Section A.2 establishes (P3) via Theorem 5 and reports the resulting computational savings. Throughout, $\alpha_t = 1 - \beta_t$, $\beta_t \in (0, 1)$, $\gamma \in [0, 1)$, and $r(\cdot \mid x_i^0, \mathbf{x}_0^{-i})$ denotes the substitution distribution over $\mathcal{M} \setminus \{x_i^0\}$.

A.1 Joint Sampling, Marginals, and Reverse Posterior

Proposition 1 (Conditional independence)**.** *Under Eq.* (5)*, the joint forward distribution conditioned on* $\mathbf{x}_0$ *factorizes across positions:*

$$q(\mathbf{x}_t \mid \mathbf{x}_0) \;=\; \prod_{i:\, b_i = 1} q(x_i^t \mid x_i^0, \mathbf{x}_0^{-i}). \tag{15}$$

Proof. Each per-step transition $q(x_i^t \mid x_i^{t-1}, \mathbf{x}_0)$ depends only on x_i^{t-1} and the deterministic context $\mathbf{x}_0^{-i}$, introducing no stochastic coupling across positions. Factorization is preserved under marginalization over $\mathbf{x}_{1:t-1}$. □

We characterize the cumulative dynamics in terms of three position-independent state probabilities. Define the state of token i at step t as O (original, $x_i^t = x_i^0$), S (substituted, $x_i^t \in \mathcal{M} \setminus \{x_i^0\}$), or M ($x_i^t = \texttt{MASK}$), with marginal probabilities $p_t^{\mathrm{O}}, p_t^{\mathrm{S}}, p_t^{\mathrm{M}}$.

Lemma 2 (State recurrences)**.** *With* $p_0^{\mathrm{O}} = 1$ *and* $p_0^{\mathrm{S}} = p_0^{\mathrm{M}} = 0$*,*

$$\begin{aligned} p_t^{\mathrm{O}} &= p_{t-1}^{\mathrm{O}}\, \alpha_t, \\ p_t^{\mathrm{S}} &= p_{t-1}^{\mathrm{S}}\, (\alpha_t + \beta_t \gamma) + p_{t-1}^{\mathrm{O}}\, \beta_t \gamma, \\ p_t^{\mathrm{M}} &= p_{t-1}^{\mathrm{M}} + (p_{t-1}^{\mathrm{O}} + p_{t-1}^{\mathrm{S}})\, \beta_t (1 - \gamma), \end{aligned} \tag{16}$$

and $p_t^{\mathrm{O}} + p_t^{\mathrm{S}} + p_t^{\mathrm{M}} = 1$ *for all* t*.*

Proof. The recurrences follow from Eq. (5) after collapsing the substituted state's self-retention α_t and would-be second substitution $\beta_t \gamma$ into combined retention $\alpha_t + \beta_t \gamma$, as dictated by the semi-absorbing rule. Conservation follows by summing (16) and applying $\alpha_t + \beta_t = 1$. □

Theorem 3 (Closed-form cumulative marginal)**.** *For all* $t \in \{1, \ldots, T\}$*,*

$$q(x_i^t = m \mid x_i^0, \mathbf{x}_0^{-i}) \;=\; \begin{cases} p_t^{\mathrm{O}}, & m = x_i^0, \\ p_t^{\mathrm{S}} \cdot r(m \mid x_i^0, \mathbf{x}_0^{-i}), & m \in \mathcal{M} \setminus \{x_i^0\}, \\ p_t^{\mathrm{M}}, & m = \texttt{MASK}. \end{cases} \tag{17}$$

Proof. Each value of m is consistent with exactly one of the three states. Conditional on S, the substituted token is drawn from $r(\cdot \mid x_i^0, \mathbf{x}_0^{-i})$ at the step of first substitution and preserved by self-retention thereafter, so its conditional distribution is r itself. Multiplying each conditional by its state probability yields (17), which is normalized since $p_t^{\mathrm{O}} + p_t^{\mathrm{S}} \sum_{m \neq x_i^0} r(m \mid \cdot) + p_t^{\mathrm{M}} = 1$. □

Proposition 4 (Reverse posterior)**.** *The per-position reverse posterior admits the closed form*

$$q(x_i^{t-1} \mid x_i^t, \mathbf{x}_0) \;\propto\; q(x_i^t \mid x_i^{t-1}, \mathbf{x}_0) \cdot q(x_i^{t-1} \mid x_i^0, \mathbf{x}_0^{-i}), \tag{18}$$

where the first factor is given by Eq. (5) *and the second by Theorem 3. Evaluation and normalization cost $O(|\mathcal{M}|)$ per position.*

Proof. Bayes' rule combined with Proposition 1 yields (18); both factors are distributions over $\mathcal{M}^+$ with $|\mathcal{M}| + 2$ atoms. □

A.2 Terminal Convergence and Computational Cost

Theorem 5 (Terminal convergence)**.** *If $\beta_t \geq \epsilon > 0$ for all t and $\gamma < 1$, then*

$$p_T^{\mathrm{O}} \;\leq\; (1-\epsilon)^T, \qquad p_T^{\mathrm{S}} \;\leq\; T\,\rho^T \quad \textit{with } \rho := 1 - \epsilon(1-\gamma) < 1, \tag{19}$$

and consequently $p_T^{\mathrm{M}} \to 1$ and $q(x_i^T \mid x_i^0, \mathbf{x}_0^{-i}) \to \delta_{\texttt{MASK}}$ as $T \to \infty$, independently of x_i^0 and $\mathbf{x}_0^{-i}$.

Proof. From (16), $p_t^{\mathrm{O}} = \prod_{s=1}^t \alpha_s \leq (1-\epsilon)^t$. The substituted state satisfies $p_t^{\mathrm{S}} \leq \rho\, p_{t-1}^{\mathrm{S}} + p_{t-1}^{\mathrm{O}} \leq \rho\, p_{t-1}^{\mathrm{S}} + (1-\epsilon)^{t-1}$, and unrolling gives $p_t^{\mathrm{S}} \leq t\,\rho^t$. Both bounds vanish as $T \to \infty$, so $p_T^{\mathrm{M}} = 1 - p_T^{\mathrm{O}} - p_T^{\mathrm{S}} \to 1$. Substituting into Theorem 3 yields the limiting point mass at `MASK`. □

The convergence is independent of the substitution structure: enriching the trajectory with pharmacologically guided corruption does not alter the asymptotic behavior of the forward process. At $T = 100$ with the cosine schedule used in our experiments, p_T^{O} and p_T^{S} are numerically negligible.

Computational cost. The closed-form expressions in Eqs. (16)–(18) reduce all training- and inference-time operations to $O(|\mathcal{M}|)$ per position, eliminating the per-sample matrix construction that would otherwise be required by a context-coupled forward process. Table 4 contrasts the two approaches.

Table 4: Computational cost under naive matrix construction versus the closed-form formulation. L: sequence length; $|\mathcal{M}|$: vocabulary size; T: diffusion steps.

Operation	Matrix formulation	Closed-form (ours)
State dynamics precomputation	$O(T\|\mathcal{M}\|^2)$	$O(T)$
Per-sample transition construction	$O(\|\mathcal{M}\|^2)$	—
Direct sampling of $\mathbf{x}_t$	$O(L\|\mathcal{M}\|)$	$O(L\|\mathcal{M}\|)$
Reverse posterior (per position)	$O(\|\mathcal{M}\|^2)$	$O(\|\mathcal{M}\|)$

B Experimental Setup Details

B.1 Dataset Statistics

Table 5 summarizes the statistics of all three datasets after preprocessing. The three datasets span distinct clinical settings that together provide a comprehensive evaluation.

MIMIC-III and MIMIC-IV are intensive care databases where patients are critically ill and frequently readmitted, resulting in rich longitudinal histories (averaging 4.92 and 7.28 visits per patient respectively). Each visit is densely coded with over 13 diagnoses and multiple procedures, reflecting the severity and complexity of ICU populations. The two versions differ notably in medication vocabulary size (112 vs. 212), providing a test of scalability across different formulary scales.

Table 5: Statistics of the three evaluation datasets after preprocessing.

Item	MIMIC-III	MIMIC-IV	Real-world
# patients	6,350	7,738	1,511
# visits	15032	20725	1,934
Diagnosis coding	ICD-9	ICD-9/10	ICD-10 BJ
Procedure coding	ICD-9	ICD-9/10	ICD-9 BJ
Medication coding	ATC	ATC	ATC
Diag. space size	1,959	3,973	1,241
Proc. space size	1,440	1,338	715
Med. space size	112	212	245
Avg. / max # visits	2.36 / 29	2.67 / 42	1.28 / 2
Avg. / max # diag. per visit	13.79 / 39	13.39 / 39	3.54 / 18
Avg. / max # proc. per visit	4.40 / 28	2.57 / 28	2.05 / 6
Avg. / max # med. per visit	19.78 / 63	11.87 / 70	20.43 / 51

The real-world general inpatient dataset, collected from a tertiary hospital in China, presents a fundamentally different clinical profile. As a general ward setting, patients are predominantly admitted for less severe conditions and typically undergo a single hospitalization, in contrast to the repeated ICU admissions observed in MIMIC. Each visit contains substantially fewer diagnoses (3.54 vs. $>$13) and procedures (2.05 vs. $>$2.57), reflecting both milder clinical complexity and differences in documentation practices across institutions and healthcare systems. Despite this sparse diagnostic and procedural input, the average prescription length remains comparable to MIMIC-III (20.43 vs. 19.78), suggesting that prescribing habits in general inpatient wards do not scale proportionally with diagnostic complexity—a pattern likely attributable to institutional formulary conventions, routine prophylactic regimens, and varying documentation granularity across hospitals. This combination of limited clinical evidence, absent longitudinal context, and high prescription complexity poses a non-trivial distribution shift, testing whether models trained to leverage rich multi-visit ICU histories can generalize to sparse single-visit settings.

Coding systems. The three datasets employ different clinical coding standards, introducing additional heterogeneity. MIMIC-III uses ICD-9 exclusively for both diagnoses and procedures. MIMIC-IV spans a transitional period during which the U.S. healthcare system migrated from ICD-9 to ICD-10, resulting in mixed coding within the same database; we retain both versions without cross-mapping to preserve the original clinical semantics and avoid information loss introduced by imperfect code translation. The real-world dataset adopts the Beijing Clinical Modification of ICD-10 (ICD-10 BJ) for diagnoses and ICD-9 BJ for procedures, which are localized adaptations of the international standards tailored to Chinese clinical practice with institution-specific extensions. Despite these differences in coding granularity and vocabulary, all three datasets use the Anatomical Therapeutic Chemical (ATC) classification for medications, providing a unified drug representation across settings. This shared medication coding ensures that the DDI matrix and drug frequency statistics are comparable across datasets, while the heterogeneous diagnosis and procedure vocabularies test the robustness of the condition encoder to varying input representations.

Data preprocessing. We follow the preprocessing pipeline established by GAMENet [35] and SafeDrug [47]. Medications are mapped to ATC level-4 codes to preserve finer-grained therapeutic distinctions than the level-3 grouping adopted in some prior work. All datasets are split at the patient level into training, validation, and test sets at a ratio of 4:1:1. Patient-level splitting ensures that all visits of the same patient appear exclusively in one partition, preventing information leakage from historical visits in the training set to future visits in the test set, which is a consideration particularly important for MIMIC-III and MIMIC-IV where patients have multiple admissions.

B.2 Baseline Descriptions

We compare RxDiff against eight representative medication recommendation methods spanning discriminative, generative, and simple reference baselines. Below we briefly describe each method and its approach to DDI mitigation.

LR (Logistic Regression). A simple multi-label classification baseline that independently predicts each medication from the current visit's diagnosis and procedure features. It serves as a non-neural reference to contextualize the performance of learning-based methods. No DDI mitigation mechanism is incorporated.

LEAP [51]. An autoregressive generative model that sequentially generates medications using an RNN decoder. LEAP introduces reinforcement learning fine-tuning with a reward signal that penalizes prescriptions containing DDI pairs, representing an early attempt to incorporate safety constraints into the generation process.

GAMENet [35]. A discriminative model that combines EHR-based drug co-occurrence memory with a global DDI graph through a graph augmented memory network. Patient history is encoded via a recurrent network, and the memory module retrieves relevant drug combinations from past prescriptions. DDI control is achieved through a DDI-based loss penalty during training.

SafeDrug [47]. A discriminative model that incorporates molecular-level drug representations via dual molecular graph encoders operating on the SMILES-derived molecular graph and the DDI interaction graph. The two encoders provide complementary views of drug properties, and a DDI-aware regularization term in the loss function penalizes predicted drug pairs with known interactions.

4SDrug [39]. A discriminative model that reformulates medication recommendation as set-to-set prediction, explicitly recognizing the set nature of both input symptoms and output prescriptions. Despite the set-level formulation, prediction is ultimately reduced to per-drug scoring with set-level aggregation to avoid combinatorial explosion. DDI mitigation is handled through a training-time loss.

COGNet [46]. An autoregressive generative model built on a Transformer decoder with a copy-or-generate mechanism. At each generation step, the model either copies a medication from the patient's historical prescriptions or generates a new one from the vocabulary. Longitudinal history is leveraged through the copy mechanism, and DDI control relies on DDI-aware input representations.

MoleRec [49]. A discriminative model that refines drug representations at the molecular substructure level, aligning substructure combinations with patient clinical conditions. By decomposing drugs into substructures and modeling their interactions with diagnoses, MoleRec captures finer-grained drug-condition relationships. DDI mitigation is achieved through a weight annealing strategy that adaptively adjusts the DDI loss weight based on the gap between the current DDI rate and a predefined target, enabling more stable and effective DDI control during training.

REFINE [5]. A discriminative model that introduces fine-grained DDI severity modeling by replacing the binary DDI matrix with a severity-weighted interaction graph. This allows the training loss to differentially penalize interactions based on their clinical risk level, prioritizing the suppression of severe interactions over mild ones.

SubRec [11]. A discriminative model that integrates drug substructure representations with longitudinal EHR encoding for personalized recommendation. It models substructure-level interactions between drugs and patient conditions, and incorporates a DDI-related loss penalty. SubRec achieves the highest accuracy among discriminative baselines but with less effective DDI control.

B.3 Evaluation Metrics

We evaluate medication recommendation performance along three dimensions: recommendation accuracy, interaction safety, and prescription length calibration. Let $\hat{X}_0$ denote the predicted prescription and X_0 the ground-truth prescription for a given visit, both represented as subsets of the medication vocabulary $\mathcal{M}$.

Jaccard Similarity. The Jaccard index measures the set-level overlap between the predicted and ground-truth prescriptions:

$$\text{Jaccard} = \frac{1}{|\mathcal{T}|} \sum_{k \in \mathcal{T}} \frac{|\hat{X}_0^{(k)} \cap X_0^{(k)}|}{|\hat{X}_0^{(k)} \cup X_0^{(k)}|} \tag{20}$$

where $\mathcal{T}$ denotes the set of test visits. Jaccard penalizes both missing and spurious medications equally, providing a balanced measure of prescription accuracy.

F1 Score. The F1 score is the harmonic mean of precision and recall, computed per visit and averaged:

$$\text{Precision}^{(k)} = \frac{|\hat{X}_0^{(k)} \cap X_0^{(k)}|}{|\hat{X}_0^{(k)}|}, \quad \text{Recall}^{(k)} = \frac{|\hat{X}_0^{(k)} \cap X_0^{(k)}|}{|X_0^{(k)}|} \tag{21}$$

$$\text{F1} = \frac{1}{|\mathcal{T}|} \sum_{k \in \mathcal{T}} \frac{2 \cdot \text{Precision}^{(k)} \cdot \text{Recall}^{(k)}}{\text{Precision}^{(k)} + \text{Recall}^{(k)}} \tag{22}$$

PRAUC (Area Under the Precision–Recall Curve). PRAUC evaluates the model's ranking quality across all medications. For each visit, the model produces a score for every medication in $\mathcal{M}$, and PRAUC measures the area under the precision–recall curve obtained by varying the decision threshold:

$$\text{PRAUC} = \frac{1}{|\mathcal{T}|} \sum_{k \in \mathcal{T}} \int_0^1 \text{Precision}^{(k)}(r)\, dr \tag{23}$$

where $\text{Precision}^{(k)}(r)$ is the precision at recall level r for visit k. Unlike Jaccard and F1, which evaluate binary predictions, PRAUC assesses the quality of the underlying score distribution before thresholding.

DDI Rate. The DDI rate measures the fraction of drug pairs in predicted prescriptions that have known harmful interactions:

$$\text{DDI} = \frac{\sum_{k \in \mathcal{T}} \sum_{(i,j):\, i<j,\, m_i, m_j \in \hat{X}_0^{(k)}} A_{m_i, m_j}}{\sum_{k \in \mathcal{T}} \binom{|\hat{X}_0^{(k)}|}{2}} \tag{24}$$

where A is the DDI matrix. The DDI rate reflects population-level safety by computing the ratio of interacting pairs to total drug pairs across all test prescriptions. A lower DDI rate indicates safer recommendations on average, though it does not capture the per-prescription distribution of interaction risk, which we analyze separately in Section 5.3.

Avg. #Meds. The average number of medications per prescription measures whether the model generates prescriptions of appropriate length:

$$\text{Avg. \#Meds} = \frac{1}{|\mathcal{T}|} \sum_{k \in \mathcal{T}} |\hat{X}_0^{(k)}| \tag{25}$$

This metric serves as a calibration check: a model that achieves high accuracy by over- or under-prescribing would exhibit a substantial deviation from the ground-truth average, even if set-overlap metrics appear reasonable.

B.4 Implementation Details

Table 6 summarizes the hyperparameter settings of RxDiff. All experiments are conducted on a single NVIDIA V100 GPU.

Model architecture. Both the condition encoder (visit-level Transformer) and the denoising backbone share the same architectural configuration: hidden dimension $d = 128$, 2 Transformer layers with 4 attention heads and pre-layer normalization. The condition encoder uses learned positional embeddings to capture visit ordering, while the denoising backbone uses separate learned positional embeddings for drug positions. The timestep t is encoded via a sinusoidal embedding projected through a two-layer MLP to match the hidden dimension. The patient contextual vector $\mathbf{h}_c$ is prepended to the drug sequence as a virtual condition token, as described in Section 4.3.3.

Training details. All models are trained with the AdamW optimizer at a learning rate of 1×10^{-4} and a batch size of 32 for a maximum of 100 epochs. Training is terminated early if the validation Jaccard score does not improve for 10 consecutive epochs, and the checkpoint with the highest validation Jaccard is used for evaluation. At each training step, a timestep t is sampled uniformly from $\{1, \ldots, T\}$, and the noised sequence $\mathbf{x}_t$ is drawn directly from the closed-form cumulative marginal (Theorem 3) without simulating intermediate steps. The training loss $\mathcal{L} = \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{len}}$ is computed as described in Section 4.3.4, with both components weighted equally.

Table 6: Hyperparameter settings of RxDiff.

Hyperparameter	**Value**
Model Architecture	
Hidden dimension d	128
Transformer layers N_{layer}	2
Attention heads	4
Positional embedding	Learned
Timestep embedding	Sinusoidal + MLP projection
Diffusion Process	
Total diffusion steps T	100
Noise schedule	Cosine
Substitution ratio γ	0.2
DDI noise temperature τ	1.5
Training	
Optimizer	AdamW
Learning rate	1×10^{-4}
Batch size	32
Maximum epochs	100
Early stopping	Patience 10 (validation Jaccard)
Inference	
Inference steps	$T = 100$
Modulation strength μ	Dataset-dependent (see below)

Selection of modulation strength μ. The inference-time modulation strength μ controls the trade-off between recommendation accuracy and DDI safety. As described in Section 5.1, μ is selected per dataset as the largest value that incurs less than 1% absolute degradation in Jaccard similarity on the validation set compared to $\mu = 0$. Specifically, we sweep μ over $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ and evaluate both Jaccard and DDI rate on the validation set for each value. The selected values are $\mu = 0.3$ for MIMIC-III, $\mu = 0.6$ for MIMIC-IV, and $\mu = 0.2$ for the real-world dataset. The lower μ on the real-world dataset reflects its smaller DDI matrix density, where aggressive modulation would over-suppress valid candidates.

Noise schedule. The noise level β_t follows a cosine schedule adapted from [28]:

$$\beta_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}, \quad \bar{\alpha}_t = \frac{f(t)}{f(0)}, \quad f(t) = \cos\left(\frac{t/T + s}{1+s} \cdot \frac{\pi}{2}\right)^2 \tag{26}$$

where $s = 0.008$ is a small offset to prevent β_t from being too small near $t = 0$. This schedule provides a gradual corruption at early steps and accelerates toward the end, which empirically yields more stable training and better reconstruction quality compared to a linear schedule. The resulting β_t is clipped to $[0, 0.999]$ to ensure numerical stability.

Baseline implementation. All baseline models are trained using their officially released code with the originally reported hyperparameters. For methods that specify a target DDI rate (e.g., SafeDrug, MoleRec), we retain the default target values from their respective publications. All baselines are trained on the same data splits and evaluated under identical metrics for fair comparison.

C Supplement Experiment

C.1 Iterative Denoising Analysis

We evaluate the effect of the total number of diffusion steps T and the inference starting step t_{inf} on recommendation performance. Figure 5 presents heatmaps of F1 score across $T \in \{10, 20, 40, 60, 80, 100, 120, 140\}$ and $t_{\text{inf}} \in \{1, 5, 10, 20, 40, 60, 80, 100, 120, 140\}$ (where $t_{\text{inf}} \leq T$) on all three datasets.

Three consistent patterns emerge across all datasets. First, for any fixed T, performance increases substantially as t_{inf} grows. The t_{inf}=1 row represents single-step prediction, which is conceptually

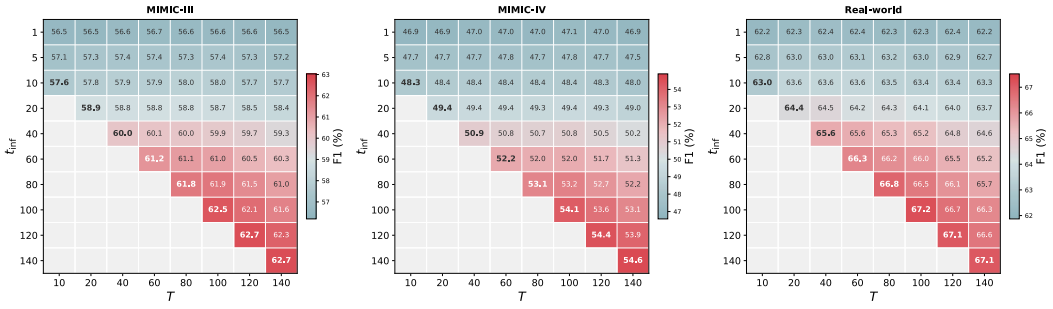

Figure 5: F1 score (%) across diffusion steps T and inference starting steps t_{inf} on three datasets. Gray cells denote invalid configurations ($t_{\text{inf}} > T$).

analogous to discriminative methods that predict all drugs in a single forward pass. The large gap between single-step and full-trajectory denoising confirms that iterative refinement captures combinatorial drug dependencies that single-pass prediction cannot resolve. Second, performance peaks near the diagonal where $t_{\text{inf}} \approx T$ and degrades when $t_{\text{inf}} \ll T$, indicating that the model benefits from utilizing the full denoising trajectory for which it was trained: partial trajectories introduce a train-inference mismatch that prevents the model from fully exploiting its learned corruption-reconstruction dynamics. Third, along the diagonal (t_{inf}=T), performance improves consistently as T increases but with diminishing returns, and T=100 represents a favorable operating point that balances accuracy with computational efficiency.

Beyond these shared trends, the three datasets exhibit informative differences that reflect their distinct clinical characteristics.

MIMIC-IV shows the largest benefit from iterative denoising. The gap between single-step prediction and full-trajectory denoising is 7.15 F1 points on MIMIC-IV, compared to 5.84 on MIMIC-III and 4.81 on the real-world dataset. This aligns with MIMIC-IV's larger medication vocabulary and more complex clinical profiles, which expands the combinatorial search space and makes multi-step refinement more critical for resolving inter-drug dependencies. The diagonal gains also decay more slowly on MIMIC-IV: T=100→120 still yields +0.34 F1, whereas MIMIC-III gains only +0.20 and the real-world dataset shows no improvement. This suggests that larger drug vocabularies require more denoising steps to fully capture the underlying combinatorial structure.

The real-world dataset saturates fastest. Starting from a higher single-step baseline (62.31 vs. 56.63 and 46.96), the real-world dataset reaches near-peak performance at fewer steps, with the diagonal plateauing around T=80–100. Beyond T=100, the diagonal F1 marginally decreases (67.12→67.05 at T=120), suggesting that excessive diffusion steps may introduce noise when longitudinal context is absent. This is consistent with the dataset's single-visit structure. Without historical prescriptions to inform the generation process, the combinatorial information extractable through iterative denoising is inherently more limited, and fewer steps suffice to capture the dominant prescribing patterns.

Based on these observations, T=100 is adopted as the default setting across all datasets, as it consistently achieves near-optimal performance while maintaining reasonable computational cost (Section C.2).

C.2 Computational Cost

Table 7 reports the training and inference time of all methods on MIMIC-III. All models are measured on a single NVIDIA V100 GPU under the same hardware environment.

Training efficiency. RxDiff trains in 3.83h, faster than all autoregressive baselines (8.56–9.21h) and comparable to discriminative methods (2.72–4.58h). This is because diffusion training does not require iterative generation. At each training step, a single timestep t is sampled and the noised sequence $\mathbf{x}_t$ is drawn directly from the closed-form cumulative marginal (Theorem 3), reducing each forward pass to a single denoising prediction. Discriminative baselines require forward passes through auxiliary molecular graph encoders (SafeDrug, MoleRec, SubRec) or DDI severity graphs (REFINE), introducing more per-step overhead. Autoregressive baselines are the slowest to train as

Table 7: Training and inference time comparison on MIMIC-III (single NVIDIA V100 GPU). Training time is wall-clock time until early stopping. Inference time is measured over the entire test set. Mean and standard deviation are reported over 5 runs.

Model	Paradigm	#Params (K)	Training Time (h)	Inference Time (s)
LR	Discriminative	301	0.08 (0.01)	10.31 (0.14)
GAMENet	Discriminative	454	3.91 (0.14)	23.47 (0.52)
SafeDrug	Discriminative	423	3.39 (0.17)	18.12 (0.47)
4SDrug	Discriminative	327	2.72 (0.20)	15.58 (0.38)
MoleRec	Discriminative	596	3.95 (0.19)	21.27 (0.56)
REFINE	Discriminative	478	3.41 (0.16)	18.89 (0.49)
SubRec	Discriminative	712	4.58 (0.18)	25.03 (0.63)
LEAP	Autoregressive	418	9.21 (0.23)	68.83 (1.42)
COGNet	Autoregressive	551	8.56 (0.25)	50.71 (1.08)
RxDiff	Diffusion	625	3.83 (0.12)	30.26 (0.74)

each training sample requires sequential token-by-token generation through the entire prescription; LEAP further incurs additional cost from reinforcement learning fine-tuning for DDI reduction.

Inference efficiency. A clear paradigm-dependent pattern emerges at inference time. Discriminative methods perform single-pass prediction and are the fastest (10.31–25.03s for the entire test set), with variation driven primarily by model complexity. SubRec (712 K parameters) is the slowest due to its substructure-level encoding. Autoregressive methods are substantially slower (50.71–68.83s), as sequential token-by-token generation scales linearly with prescription length. RxDiff's inference time of 30.26s is significantly faster than both autoregressive baselines. This advantage arises because each denoising step processes all drug positions in parallel, whereas autoregressive methods must generate tokens strictly one by one. Although RxDiff is slower than single-pass discriminative methods, its per-prescription latency of approximately 12.1ms remains well within acceptable limits for clinical decision support systems, where recommendations are generated once per patient visit rather than in real-time interactive loops. This efficiency is largely attributable to the choice of $T{=}100$ diffusion steps, which, as shown in Appendix C.1, achieves near-optimal accuracy while maintaining a moderate computational budget. If faster inference is desired, reducing to $T = 50$ yields an approximate $2\times$ inference speedup with less than 3% F1 degradation, offering a practical trade-off for latency-sensitive deployment scenarios. The inference-time interaction-aware modulation introduces negligible additional cost, requiring only a sparse lookup over the DDI matrix at each denoising step.

Model size. RxDiff contains 625 K trainable parameters, comparable to most baselines in this study. The parameters primarily reside in the condition encoder and the denoising backbone, both of which employ Transformer architectures, rather than in the diffusion process itself. Compared to methods that adopt GRU-based recurrent encoders for longitudinal EHR modeling, such as SafeDrug (423 K) and GAMENet (454 K), RxDiff introduces additional parameters, yet the inherent parallelism of Transformer computation means that this increase does not translate into proportional training overhead. In practice, RxDiff trains in 3.83h, on par with these baselines (3.39h and 3.91h respectively). The ablation study in Appendix C.5 further validates that this model capacity is effectively utilized, as replacing the Transformer backbone with lighter alternatives such as MLP or GRU leads to consistent degradation in both recommendation accuracy and DDI control.

C.3 Case Study

We present three representative cases from the MIMIC-III test set to illustrate the qualitative behavior of RxDiff. Drug names denote their ATC level-4 therapeutic categories throughout this section. Table 8 compares quantitative performance across methods, and Table 9 highlights key qualitative differences in predicted medications. Figure 6 visualizes the denoising trajectory for Case C.

Case A: DDI safety control and combinatorial dependency. The ground-truth prescription contains a Heparin–Antiplatelet pair, which is a major-severity interaction with elevated bleeding risk [13]. SafeDrug's DDI regularization leads it to omit Antiplatelet, avoiding this interaction. SubRec retains the pair, achieving higher recall but with 4 DDI pairs. RxDiff also omits Antiplatelet via inference-time modulation, while uniquely predicting Sucralfate, a gastroprotective agent whose indication

Table 8: Case study: quantitative comparison on three MIMIC-III test patients.

Case	Method	Accuracy				Safety	
		Jacc.	F1	#Hit	#Miss / #FP	DDI%	#DDI
A (12 drugs)	SafeDrug	0.57	0.73	8	4 / 2	4.44	2
	SubRec	0.69	0.82	9	3 / 1	8.89	4
	RxDiff	**0.77**	**0.87**	**10**	2 / 1	**1.82**	**1**
B (12 drugs)	SafeDrug	0.57	0.73	8	4 / 2	0.00	0
	SubRec	0.69	0.82	9	3 / 1	2.22	1
	RxDiff	**0.85**	**0.92**	**11**	1 / 1	**0.00**	**0**
C (9 drugs)	SafeDrug	0.55	0.71	6	3 / 2	10.71	3
	SubRec	0.70	0.82	7	2 / 1	17.86	5
	RxDiff	**0.80**	**0.89**	**8**	1 / 1	**5.56**	**2**

mainly arise from the combination of anticoagulation and mechanical ventilation rather than from any single diagnosis alone [7]. This illustrates the advantage of bidirectional denoising in capturing inter-drug dependencies that reflect combinatorial clinical reasoning.

Table 9: Case study: key prediction differences. ✓ = correctly predicted, ✗ = missed, + = false positive. Red drug names indicate involvement in DDI pairs within that method's prediction.

Case	Diagnoses	Key Drug	SafeDrug	SubRec	RxDiff
A	Liver disease, Aspiration pneum., Resp. failure, Hypertension	Antiplatelet (B01AC)	✗	✓	✗
		Heparin (B01AB)	✓	✓	✓
		Sucralfate (A02BX)	✗	✗	✓
		Lactulose (A06AD)	✗	✓	✓
		Amiodarone (C01BD)	+	—	—
		Digoxin (C01AA)	—	+	+
B	CHF, CKD, Diabetes, Acute renal fail., Hyperlipidemia	Insulin (A10AE)	✗	✓	✓
		Metformin (A10BA)	—	+	—
		Atropine (A03BA)	✗	✗	✓
		Ondansetron (A04AA)	✗	✓	✓
		Simvastatin (C10AA)	+	—	+
C	Hypertension, Diabetes, GI bleeding, Diverticulosis	Losartan (C09CA)	✓	✗	✓
		Amlodipine (C08CA)	—	+	—
		Sulfonylurea (A10BB)	✗	✗	✓
		Metoprolol (C07AB)	✓	✓	✓
		Ranitidine (A02BA)	+	—	—

Case B: Diagnosis-drug compatibility. Glycemic control in this CKD patient requires careful drug selection. SubRec predicts Metformin, which is generally contraindicated in renal impairment [9]; the ground truth uses Insulin instead. This also introduces a Metformin–Furosemide DDI pair. SafeDrug avoids this error but misses several ground-truth drugs. RxDiff correctly predicts Insulin and uniquely identifies Atropine, suggesting that iterative denoising may help condition drug predictions on the broader diagnostic context.

Case C: Denoising revision. SubRec predicts Amlodipine instead of the ground-truth Losartan, introducing an interaction with the co-prescribed Metoprolol. Figure 6 visualizes how the prescription evolves across denoising steps. As denoising progresses, high-frequency drugs such as Heparin stabilize first (t=90), while less common drugs like Sulfonylurea require more steps to converge (t=20). The key revision occurs when Metoprolol stabilizes by t=80, and Amlodipine subsequently emerges as a candidate at t=70. At t=60, both are simultaneously present and marked as a DDI pair. The interaction-aware modulation then suppresses Amlodipine's logit, and Losartan replaces it by t=40, resolving the interaction. This trajectory provides an intuitive illustration of how iterative refinement and inference-time safety control jointly enable the progressive construction of prescriptions that are more accurate and interaction-aware.

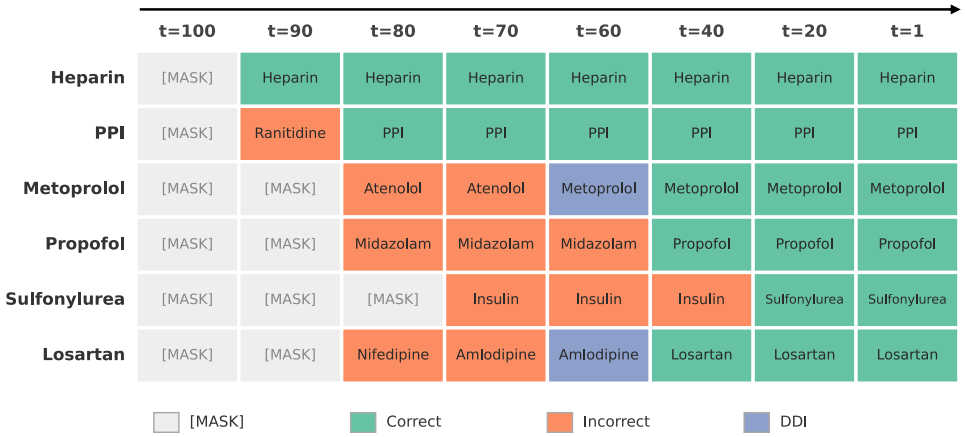

Figure 6: Denoising trajectory for Case C across six drug positions and eight timesteps.

C.4 Rare Disease Analysis

To evaluate robustness across varying levels of diagnostic rarity, we stratify test patients on MIMIC-III by the training-set frequency of their rarest diagnosis. For each patient, we identify the diagnosis with the lowest occurrence count in the training set and partition patients into three equally-sized groups: high-frequency (top tercile), mid-frequency, and low-frequency (bottom tercile). Figure 7 compares RxDiff against three representative baselines on F1 and DDI rate within each group.

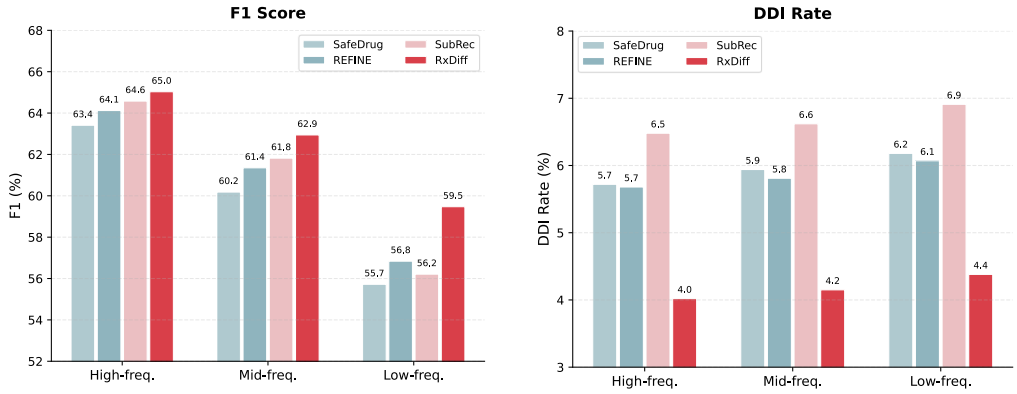

Figure 7: F1 score and DDI rate across diagnostic frequency groups on MIMIC-III. Patients are stratified by the training-set frequency of their rarest diagnosis.

RxDiff degrades most gracefully on rare diagnoses. All methods exhibit lower F1 when moving from high-frequency to low-frequency patients, as expected from reduced training coverage of the diagnosis-to-prescription mapping. However, the degradation is substantially smaller for RxDiff: its F1 drops by 5.56 points (65.03→59.47), compared to 7.69 for SafeDrug, 7.28 for REFINE, and 8.37 for SubRec. This relative robustness can be attributed to two properties of the diffusion framework. First, iterative denoising allows the model to progressively refine predictions through multiple passes over the entire prescription state, effectively performing implicit search over the combinatorial space even for unfamiliar diagnostic contexts. Second, the frequency-guided forward process trains the model to recover patient-specific prescriptions from population-level drug statistics, providing a meaningful fallback prior for prescriptions when patient-specific diagnostic signals are scarce.

DDI rates remain relatively stable across frequency groups. DDI rates exhibit mild variation across the three groups for all methods. This is expected as DDI control mechanisms, whether training-time loss penalties or inference-time modulation, operate at the drug-pair level, which is largely independent of diagnostic frequency. Most prescribed medications are individually common even for rare diagnoses. The small residual increase in the low-frequency group reflects the fact that while individual drugs may be common, the specific drug combinations prescribed for rare diagnoses

can be unusual at the combinatorial level, creating interaction patterns less frequently encountered during training. Across all groups, RxDiff maintains a consistent DDI advantage over all baselines, confirming that inference-time modulation provides reliable safety control regardless of diagnostic rarity.

C.5 Backbone Ablation

We ablate the denoising backbone by replacing the default 2-layer bidirectional Transformer with alternative architectures while keeping all other components (condition encoder, forward process, inference-time modulation) fixed. Table 10 reports results on MIMIC-III.

Table 10: Backbone ablation on MIMIC-III. All variants use identical diffusion configurations (T=100, γ=0.2, τ=1.5) and inference-time modulation. Best results are in **bold**.

Backbone	Interaction	Jaccard↑	F1↑	PRAUC↑	DDI↓
Architecture type					
MLP	None	44.82 (0.26)	58.63 (0.29)	66.41 (0.32)	5.87 (0.14)
GRU	Unidirectional	46.35 (0.23)	60.24 (0.25)	68.12 (0.28)	5.14 (0.12)
Bi-GRU	Bidirectional	47.21 (0.21)	61.18 (0.23)	69.05 (0.25)	4.72 (0.10)
Causal Trans. (2L)	Unidirectional	47.48 (0.20)	61.52 (0.22)	69.43 (0.24)	4.58 (0.09)
Transformer depth					
Transformer (1L)	Bidirectional	47.76 (0.19)	61.74 (0.21)	69.68 (0.23)	4.41 (0.08)
Transformer (2L)	Bidirectional	**48.49** (0.17)	**62.57** (0.24)	**70.59** (0.20)	**4.19** (0.07)
Transformer (4L)	Bidirectional	48.27 (0.22)	62.31 (0.26)	70.38 (0.24)	4.25 (0.09)

Positional interaction is essential. The MLP backbone predicts each drug position independently without any cross-position communication, yielding the lowest accuracy (Jaccard 44.82) and the highest DDI rate (5.87%). This confirms that combinatorial drug dependencies cannot be captured by per-position prediction alone, and that explicit inter-position interaction within the denoising backbone is critical for both recommendation quality and safety.

Bidirectional interaction outperforms unidirectional. Comparing architectures with the same directionality reveals a consistent advantage for bidirectional modeling. The GRU, which processes positions sequentially in one direction, improves substantially over MLP but falls short of Bi-GRU across all metrics (Jaccard 46.35 vs. 47.21). Similarly, the causal Transformer (2L), which restricts attention to preceding positions, underperforms the bidirectional Transformer (2L) by 1.01 Jaccard and 1.05 F1 despite having identical capacity. This gap directly validates the motivation of adopting diffusion over autoregressive generation that bidirectional access to the full prescription state at every denoising step enables more informed predictions than unidirectional conditioning on a partial prefix.

Global attention provides further gains over recurrent interaction. The bidirectional Transformer (1L) outperforms Bi-GRU (Jaccard 47.76 vs. 47.21) with fewer parameters, demonstrating that the direct pairwise attention mechanism captures inter-drug dependencies more effectively than the indirect hidden-state propagation in recurrent architectures. This advantage is particularly relevant for DDI modeling, where interactions are fundamentally pairwise: attention weights can directly attend to potentially conflicting drug positions, while recurrent hidden states must compress this information through a sequential bottleneck.

Moderate depth suffices. Increasing the Transformer from 1 to 2 layers yields a meaningful improvement (Jaccard +0.73, F1 +0.83), but further increasing to 4 layers leads to marginal degradation (Jaccard -0.22, F1 -0.26), likely due to overfitting on the relatively small medication vocabularies. The 2-layer configuration achieves the best trade-off between model capacity and generalization.

D Limitations

While RxDiff demonstrates consistent improvements across multiple datasets, we acknowledge several limitations that warrant discussion and future investigation.

Inference latency from multi-step denoising. The $T = 100$ step reverse process requires sequential forward passes through the denoising backbone, resulting in approximately $1.8\times$ inference time compared to single-pass discriminative methods. Although the per-prescription latency ($\sim$12ms) is clinically acceptable for visit-level recommendation, it may become a concern in scenarios requiring real-time interactive response, such as iterative prescription editing at the point of care. Accelerated sampling strategies from the diffusion literature, such as step skipping or distillation, could mitigate this overhead and represent a promising direction for future work.

Binary DDI matrix assumption. The current framework treats drug-drug interactions as binary (present or absent), whereas in clinical practice interactions vary substantially in severity, from mild pharmacokinetic alterations to life-threatening adverse events. REFINE [5] has explored severity-weighted DDI modeling within the discriminative paradigm. Extending the interaction-aware modulation in Eq. (13) to incorporate graded severity scores, for instance by replacing the binary $A_{m,j}$ with a continuous severity weight, is a natural extension that would enable clinically more nuanced safety control.

Length predictor error propagation. The prescription length is estimated independently by the length predictor before the reverse process begins. Errors in this estimate directly affect the generation space: overestimation introduces spurious drug positions, while underestimation truncates the prescription. Although the length predictor achieves high accuracy in our experiments (as reflected by the close match between predicted and ground-truth #Meds in Tables 1 and 2), the error is non-recoverable once the reverse process is initialized. A dynamic length adjustment mechanism that allows the model to add or remove drug positions during denoising could alleviate this limitation.

Population coverage and fairness. Our evaluation is conducted on two ICU-centric databases from the United States (MIMIC-III, MIMIC-IV) and one general inpatient dataset from a single tertiary hospital in China. The demographic composition, disease spectrum, and prescribing conventions in these datasets may not fully represent the diversity of global clinical practice. Evaluating and mitigating potential disparities across patient demographics is an important direction for responsible deployment.

Static DDI knowledge. Although the inference-time modulation mechanism can incorporate updated DDI matrices without retraining (as demonstrated in the robustness experiment in Section 5.3), the current framework assumes that the DDI matrix is fixed and externally provided. It does not actively discover or predict novel interactions from the data. Integrating DDI prediction capabilities, for instance through learned interaction representations, could further strengthen the safety mechanism by anticipating interactions not yet catalogued in existing knowledge bases.

E Broader Impact

This work develops a medication recommendation framework intended to assist clinical decision-making by generating effective and safe drug combinations. We discuss both the potential positive impacts and the risks that should be carefully considered before deployment.

Positive impacts. Medication errors and adverse drug interactions represent a significant source of preventable patient harm in healthcare systems worldwide. RxDiff's dual-stage safety mechanism, which combines training-time DDI awareness with inference-time modulation, offers a principled approach to reducing harmful drug combinations in recommended prescriptions. The inference-time controllability is particularly valuable in clinical practice, as it allows physicians to adjust the safety-accuracy trade-off based on the risk tolerance appropriate for individual patients. For example, a stricter modulation strength may be applied for elderly patients with polypharmacy, who are at elevated risk for adverse interactions. Furthermore, the ability to update the DDI matrix without retraining supports deployment in real-world settings where pharmacological knowledge evolves continuously, reducing the maintenance burden for healthcare institutions.

Risk of over-reliance. A central concern with any clinical decision support system is the potential for over-reliance by healthcare professionals. If clinicians treat model recommendations as authoritative rather than advisory, errors in the model's predictions, whether due to distribution shift or incomplete training data, could propagate into actual prescriptions without adequate scrutiny. RxDiff is designed as a decision support tool to augment, not replace, clinical judgment. Deployment should incorporate

clear interface design that communicates model uncertainty and encourages physician review before finalizing prescriptions.

Fairness and equity concerns. The model is trained on retrospective EHR data that inherits the biases present in historical clinical practice. Prescribing patterns may systematically differ across demographic groups due to socioeconomic factors, insurance coverage, or institutional conventions, and the model may perpetuate or amplify these disparities. Patients from underrepresented groups in the training data, such as racial minorities, pediatric populations, or those with rare conditions, may receive lower-quality recommendations. Rigorous subgroup evaluation and fairness auditing should be conducted prior to clinical deployment to identify and mitigate such disparities.

Scope of applicability. RxDiff is evaluated on inpatient settings (ICU and general ward) and should not be directly applied to outpatient, emergency, or pediatric contexts without further validation. The DDI matrix used in this work captures known pairwise interactions but does not account for higher-order multi-drug interactions, drug-food interactions, or patient-specific pharmacogenomic factors. Clinical deployment should complement model recommendations with comprehensive pharmaceutical review.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state three contributions. All claims are supported by theoretical analysis and experimental results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be

used reliably to provide closed captions for online lectures because it fails to handle technical jargon.

- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theoretical results are formally stated with explicit assumptions and complete proofs in Appendix. The main text provides intuition and references the formal proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The main text specifies key hyperparameters, and Appendix provides complete implementation details including model architecture, optimizer settings, noise schedule and baseline configurations. Anonymous code is also provided.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Anonymous source code and instructions are provided. Public datasets are accessible through PhysioNet. The real-world dataset cannot be publicly released due to institutional data sharing agreements, but we document the preprocessing pipeline and data format, and may share it with qualified researchers under a formal data use agreement upon reasonable request.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All the training and test details are described in Section 5.1 and Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Tables 1 and 2 report standard deviations across multiple runs in parentheses for all metrics. Statistical significance is assessed via paired t-tests ($p < 0.05$) as noted in the table caption.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments are conducted on a single NVIDIA V100 GPU as reported in Appendix B.4. Training and inference time comparisons are provided in Appendix C.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: The research uses de-identified public datasets (MIMIC-III, MIMIC-IV) and a de-identified real-world dataset. No personally identifiable information is used or released.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Appendix E discusses positive impacts and potential risks.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper poses no such risks for misuse.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: MIMIC-III [19] and MIMIC-IV [18] are cited and used under the PhysioNet Credentialed Health Data License. All baseline methods are cited with their original publications.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released anonymized code includes documentation with details about setting and training.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This research does not involve crowdsourcing or human subjects experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This research uses only de-identified retrospective datasets. MIMIC-III and MIMIC-IV are publicly available under PhysioNet with their own ethical approvals. The real-world dataset is fully de-identified with no direct interaction with patients, and does not constitute human subjects research.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not used as any component of the core methodology in this research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.