Anchoring LLM-based Chest X-ray Report Generation via Diffusion Language Planning

Jiechao Gao[1*†] **Chang Liu**[2*] **Yuandong Pan**[1] **Ying Liu**[3] **Michael Lepech**[1]
[1]Stanford University [2]The University of Hong Kong
[3]University of Chinese Academy of Sciences

Abstract

While Large Language Models (LLMs) have significantly advanced radiology report generation (RRG) with strong generative priors, standard autoregressive decoders still suffer from sequential error accumulation when they must infer all clinical structures directly from images. Diffusion Language Models (DLMs) have recently emerged as a non-autoregressive paradigm with iterative masking and denoising mechanisms. Instead of replacing the mature language decoder with a DLM, we reveal that the key value of DLMs for RRG lies in controllable clinical planning, where the model predicts globally consistent anchor points before free-text realization. To harness this complementary strength, we propose ANCHOR-GEN, the first framework that uses a DLM as an explicit diffusion language planner for RRG. Rather than forcing the autoregressive decoder to infer every critical structure from visual tokens alone, ANCHOR-GEN predicts a structured clinical canvas that serves as clinically grounded anchor points. An autoregressive LLM then leverages these anchors to complete report generation with both fluency and clinical faithfulness. By systematically combining non-autoregressive planning and autoregressive decoding, our unified pipeline is trained with structured denoising, hierarchical coarse-to-fine supervision, and planner-decoder binding objectives. Extensive experiments on the public MIMIC-CXR benchmark demonstrate that ANCHOR-GEN improves clinical faithfulness over prior works. Comprehensive analyses and ablations further show that diffusion planning and causal decoding offer complementary strengths for RRG, and that integrating them in one pipeline provides an effective route toward more faithful medical text generation.

1 Introduction

Automatic chest X-ray report generation aims to translate radiographic evidence into clinically faithful free-text reports. Traditionally, drafting clinical reports is labor-intensive and error-prone, so automatic algorithms have the potential to reduce the burden of radiologists. Recent studies increasingly shift toward large autoregressive decoders [1, 2, 3, 4, 5, 6, 7] because these models provide strong language priors and fluent long-form realization. However, real clinical utility depends on diagnostic accuracy rather than fluency alone. This requires the generated report to preserve decisive findings, anatomical locations, disease severities, and support-device states. Because autoregressive LLMs can still produce linguistically coherent but factually incorrect hallucinations, relying only on direct image-to-report alignment introduces a clear risk for clinical practice. Accordingly, improving clinical faithfulness becomes a central focus of recent studies [3, 4, 7].

*Equal contribution.
†Corresponding author: jiechao@stanford.edu.

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

To address the limitations of purely autoregressive decoding, diffusion language models provide a non-autoregressive alternative. Recent DLMs [8, 9, 10] show that this paradigm can already approach the generation quality of strong autoregressive models, and ECHO [11] further reports diffusion-based CXR-RG results under a normalized chest X-ray reporting protocol. Even so, current DLMs still do not replace mature LLM decoders in open-ended free-text generation. What they do offer is a more natural interface for controlled generation and iterative infill, as shown by Diffusion-LM [12] and SeqDiffuSeq [13]. This property matches the clinical demand of RRG because a report must jointly cover multiple constrained attributes, including disease existence, laterality, location, severity, and devices, rather than only produce a fluent sentence. An ideal RRG system first establishes factually grounded pathological anchors and then expands them into a coherent report. If such explicit planning can be coupled with fluent free-text decoding, the hallucination problem can be reduced without sacrificing report quality. This observation motivates us to combine DLM-based planning with LLM-based realization in a unified RRG pipeline. On one hand, the non-autoregressive structure of DLMs suits clinically grounded anchor prediction and avoids left-to-right error accumulation. On the other hand, autoregressive LLMs remain the strongest option for causal report generation.

Motivated by the aforementioned observations, we present ANCHOR-GEN, an integrated framework that leverages the complementary advantages of DLMs and LLMs through three core components, i.e., Diffusion Language Planner (DLP), Gated Planner-Memory Bridge (GPMB), and Anchored Causal Decoder (ACD). Instead of forcing the causal LLM to implicitly guess clinical structures from the visual representation, DLP performs non-autoregressive masked infill to capture factually grounded clinical attributes, which subsequently serve as the anchor points of the report generation process. Afterwards, GPMB bridges the planner and the LLM decoder through textual anchor prompting and planner-aware visual-state injection, allowing the decoder to use both explicit anchor phrases and hidden planner evidence during generation. Finally, ACD conditions the autoregressive report decoder on these anchored signals and verbalizes the structured findings into a linguistically coherent free-text report. In general, we summarize our contributions as follows:

- To our best knowledge, we are the first study to deploy a DLM as an explicit clinical planner for RRG, separating diffusion-based anchor prediction from autoregressive report realization.
- In this paper, we propose ANCHOR-GEN, with DLP for predicting factually grounded clinical anchors, GPMB for dual-channel planner-to-decoder bridging, and ACD for binding these anchors to cohesive reports through staged objectives.
- Extensive experiments on the widely used benchmark, i.e., MIMIC-CXR [14], illustrate our effectiveness compared to existing state-of-the-art methods and provide interpretable evidence for why structured diffusion planning improves clinical faithfulness.

2 Related Work

2.1 Radiology Report Generation

Early RRG studies mostly train task-specific encoder-decoder models from scratch, and focus on improving the cross-modal alignment between radiology images and reports. Specifically, prior works achieve this through memory [15], posterior-prior knowledge distillation [16], auxiliary medical interpretation [17], region grounding [18], expert token specialization [19], expert knowledge [20], disease-aware retrieval and self-correction [21], and longitudinal contrastive alignment [22, 23]. Despite these improvements, such methods fail in real-world clinical scenarios due to critical factual inaccuracies. Beyond their reliance on closed-set training paradigms that inherently limit their generalization capabilities, these models often fail to extract accurate clinical findings, which is a fundamental prerequisite for generating reliable reports.

With the rise of LLMs, recent studies introduce their strong autoregressive generative prior for RRG. Bootstrapping LLMs [1] presents a contrastive learning- and coarse-to-fine-based paradigm to tune generic LLM decoders specifically for report generation. LLM-RG4 [3] focuses on integrating diverse input contexts to facilitate RRG. HC-LLM [4] exploits historical constraints from prior studies. SelfAlign [24] uses iterative self-improvement to refine generated reports and improve overall factual correctness. S2D-Align [7] injects anatomically grounded auxiliary supervision into

the SFT alignment pipeline. Although these LLM-based methods produce strong improvements, their autoregressive decoding paradigm still suffers from sequential error accumulation, thus potentially hallucinating fine-grained clinical attributes when generating particularly long reports. This intrinsic limitation motivates us to integrate the DLM as an auxiliary planning tool in this paper.

2.2 Diffusion Language Models

Early DLM research mainly studies how to adapt diffusion models to handle the text modality, as vanilla diffusion models like DDPM [25] are originally designed for continuous-valued data, e.g., images. One line follows discrete diffusion and replaces the Gaussian noise with categorical corruption over a finite vocabulary. D3PM [26] establishes this line by introducing structured transition operators, showing that a BERT-style mask token can serve as a discrete alternative of Gaussian noise during the diffusion process. Later text-oriented variants [27, 28, 29] further adapt this family to non-autoregressive text generation with stronger masked denoising objectives. The other line follows continuous representations, where the model diffuses text in a latent or embedding space so that the forward process remains compatible with the standard diffusion formulation. Specifically, Diffusion-LM [12] optimizes diffusion in continuous word representations for controllable generation; DiffuSeq [30] extends this idea to sequence-to-sequence text generation; GENIE [31] studies diffusion pre-training with continuous paragraph denoising; and Analog Bits [32] further shows that continuous binary encodings can bridge standard diffusion and discrete-valued data. These early methods establish the feasibility of text diffusion, but they usually remain at relatively small scale and still lag behind Transformer-based autoregressive models on open-ended generation quality.

Recently, Diffusion Large Language Models (DLLM) mostly inherit the line of discrete diffusion to the regime of large-scale language modeling. Particularly, LLaDA [10] shows that discrete diffusion can approach the generation quality of strong autoregressive LLMs. LLaDA-V [9] further extends the same modeling principle to visual instruction tuning, and MMaDA [33] further generalizes the framework to multimodal generation. Recent studies also improve long-form controllability and sequence-level dependency modeling through energy-based diffusion [34], segment-level diffusion [35], and diffusion-based CXR-RG [11]. Among the works most relevant to our motivation, DiffLM [36] provides especially direct evidence that diffusion-based text generation becomes attractive when the output must satisfy explicit conditional constraints. This observation motivates us to leverage DLMs to accurately capture pathological planning anchors for RRG in this paper.

3 Method

3.1 Problem Formulation

Standard VLM-based Radiology Report Generation. Given a chest X-ray image $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, our goal is to generate a report token sequence $\mathbf{Y} = (y_1, \ldots, y_T)$ that is both linguistically coherent and clinically faithful. In training, the model receives $\mathbf{X}$ and the ground-truth report $\mathbf{Y}$ to learn the cross-modal alignment; while in inference, it only takes $\mathbf{X}$ and an optional textual prompt as the input to produce a free-text report. This process is normally modeled via a standard VLM pipeline, along with a frozen visual encoder $\mathcal{E}_{\text{vis}}$ to extract dense image embeddings, a trainable cross-modal connector to map them into the embedding space of the language decoder, and the LLM to produce the final report. This process is formally expressed as:

$$\mathbf{V} = \mathcal{E}_{\text{vis}}(\mathbf{X}), \qquad \mathbf{Z} = \mathcal{C}_\theta(\mathbf{V}). \tag{1}$$

Here, $\mathbf{V}$ denotes the visual tokens produced by $\mathcal{E}_{\text{vis}}$, and $\mathbf{Z}$ denotes the projected decoder-aligned visual states after the connector $\mathcal{C}_\theta$. The autoregressive decoder then predicts the report conditioned on $\mathbf{Z}$ and the text prompt. This decoding process is formally expressed as:

$$p(\mathbf{Y} \mid \mathbf{X}) = \prod_{t=1}^{T} p(y_t \mid y_{<t}, \mathbf{Z}). \tag{2}$$

Conventionally, this standard pipeline already provides a strong ability to generate diagnostic text, but it leaves the whole burden of clinical planning and report generation to the same autoregressive

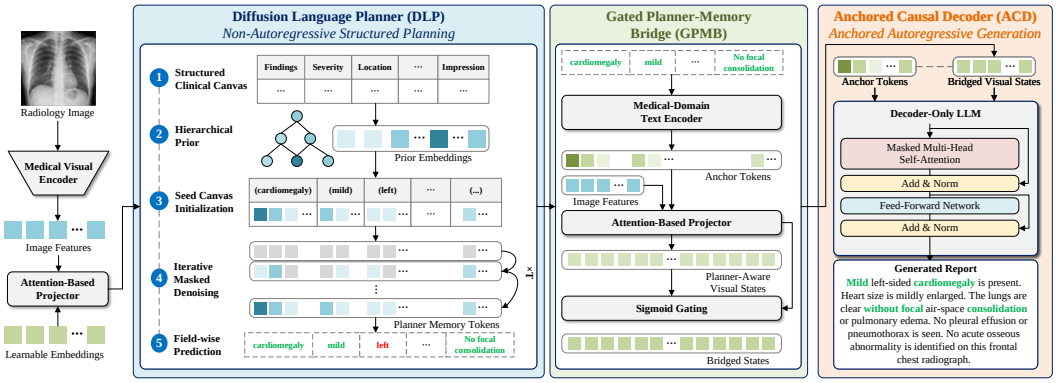


Figure 1: Overall framework of ANCHOR-GEN. The DLM planner predicts a structured clinical canvas, renders confident planner outputs into canonical phrases, and injects planner memory into the LLM through a gated bridge, eventually producing the free-text reports in an autoregressive manner.

decoder. As a result, the model often learns only coarse cross-modal alignment and may fail to preserve the key clinical cues that should anchor the final report.

Diffusion Language Models for Structured Planning. We address this gap with a Diffusion Language Model (DLM)-based planner rather than by replacing the autoregressive decoder. The reason is that a DLM naturally supports masked prediction and iterative remasking on discrete symbols, which makes it suitable for controlled generation under structured constraints. Specifically, we let it operate on a structured clinical canvas $\mathbf{S} = (s_1, \ldots, s_M)$ whose entries correspond to clinically meaningful fields such as global abnormality, device state, finding status, laterality, location, and severity. At iteration t, the planner receives a partially masked canvas $\widetilde{\mathbf{S}}^{(t)}$ and predicts the missing field values conditioned on the image context. This conditional infill process is expressed as:

$$p_{\text{plan}}\left(\mathbf{S} \mid \widetilde{\mathbf{S}}^{(t)}, \mathbf{X}\right) = \prod_{m=1}^{M} p_{\text{plan}}\left(s_m \mid \widetilde{\mathbf{S}}^{(t)}, \mathbf{X}\right). \tag{3}$$

Here, $p_{\text{plan}}(\cdot)$ denotes the conditional distribution modeled by the DLM planner. This factorization only describes the parallel field-wise prediction heads; the shared planner encoder still attends to the full canvas and image-conditioned context, so dependencies among fields are modeled before the classifier heads make local decisions. Fig. 1 provides an overview of the full pipeline, and the following sections sequentially detail Diffusion Language Planner (DLP), Gated Planner-Memory Bridge (GPMB), and Anchored Causal Decoder (ACD).

3.2 Diffusion Language Planner

The purpose of DLP is to predict a compact set of clinically grounded anchor points before the autoregressive decoder generates the final free-text reports. This design differs from earlier Transformer-based classifiers or planners, which usually take a pooled multimodal representation or a fixed set of learnable queries and then predict all clinical attributes in one shot with multiple classification heads [17, 16, 19]. Such one-step designs ignore the natural dependency between coarse- and fine-grained findings, thus lacking dedicated mechanisms for revising the generated plans. By contrast, we formulate planning as a masked discrete generation process on an explicit canvas, so the model can first decide whether an abnormality exists and then iteratively refine its attributes. Formally, we define an ordered field set $\mathcal{F} = \{f_1, \ldots, f_M\}$, where each field f_m takes one value from a small vocabulary $\mathcal{V}_m$. This canvas explicitly uses coarse states to describe coarse abnormality and the subsequent states to specify the attributes of positive findings. Let $s_m \in \mathcal{V}_m$ denote the value of field f_m. The full planner target is an ordered discrete vector $\mathbf{S} = (s_1, \ldots, s_M) \in \mathcal{V}_1 \times \cdots \times \mathcal{V}_M$. During training, we randomly mask valid fields in $\mathbf{S}$ to simulate partially observed planning states, and then enforce the DLM to reconstruct the full map correspondingly. During inference, we start from an all-mask canvas and let the planner iteratively fill it through iterative denoising.

To stabilize this process, DLP first predicts a hierarchical prior over the coarse fields. Concretely, before the iterative canvas denoising starts, we collect the image-conditioned contextual tokens into a

projected multimodal context matrix $\mathbf{C} \in \mathbb{R}^{B \times N_{\text{ctx}} \times D_{\text{plan}}}$:

$$\mathbf{C} = \mathcal{P}_{\text{ctx}} \left(\mathbf{Q}_{\text{plan}}, \mathbf{Z} \right). \tag{4}$$

Here, $\mathcal{P}_{\text{ctx}}$ denotes the attention-based context projector. We then summarize $\mathbf{C}$ with a pooling operator into a study-level hidden state $\mathbf{h}_{\text{ctx}} \in \mathbb{R}^{B \times D_{\text{plan}}}$. This prior step is expressed as:

$$\mathbf{h}_{\text{ctx}} = \text{Pool} \left(\mathbf{C} \right), \qquad \mathbf{p}_{\text{prior}}^{(m)} = \text{Softmax} \left(\mathcal{H}_{\text{prior}}^{(m)} \left(\mathbf{h}_{\text{ctx}} \right) \right). \tag{5}$$

Here, $\text{Pool}\left(\cdot\right)$ denotes the summary operator over the token dimension; $\mathcal{H}_{\text{prior}}^{(m)}$ denotes the classifier of field f_m; and $\mathcal{F}_{\text{coarse}} \subset \mathcal{F}$ denotes the set of coarse fields. We use high-confidence prior predictions as planner seeds so that the later denoising stages start from semantically plausible coarse decisions instead of from fully ambiguous states. Specifically, we convert the prior distributions into a seed canvas by keeping only confident coarse predictions and leaving all other fields masked. This initialization step is expressed as:

$$\hat{s}_m^{(0)} = \arg\max \mathbf{p}_{\text{prior}}^{(m)}, \qquad \widetilde{s}_m^{(0)} = \begin{cases} \hat{s}_m^{(0)}, & \text{if } m \in \mathcal{F}_{\text{coarse}} \text{ and } \max \mathbf{p}_{\text{prior}}^{(m)} > \tau_{\text{seed}}, \\ \texttt{[MASK]}, & \text{otherwise}, \end{cases} \tag{6}$$

where τ_{seed} denotes the confidence threshold for turning a prior prediction into a fixed coarse seed. The initial planner canvas is therefore $\widetilde{\mathbf{S}}^{(0)} = \left(\widetilde{s}_1^{(0)}, \ldots, \widetilde{s}_M^{(0)} \right)$, and the later denoising stages iteratively overwrite only the masked tokens. Afterwards, DLP performs masked infilling on the whole canvas. Given the partially observed canvas at iteration t, DLP embeds the field values or mask symbols in $\widetilde{\mathbf{S}}^{(t)}$, fuses them with the projected multimodal context, and outputs planner states for every field. Field-specific classifiers then map these states to categorical predictions via:

$$\mathbf{z}_m^{(t)} = \mathcal{H}_m \left(\mathbf{H}_{\text{plan},m}^{(t)} \right), \qquad \mathbf{p}_m^{(t)} = \text{Softmax} \left(\mathbf{z}_m^{(t)} \right). \tag{7}$$

Here, $\mathcal{H}_m$ denotes the classifier of field f_m. We follow a coarse-to-fine update order during inference, matching the real scenarios where radiologists give diagnoses—they first recognize the existence of a lesion then translate it into detailed attributes. At each step, DLP updates only the currently masked fields with high-confidence predictions and keeps the remaining fields masked for later refinement:

$$\widetilde{s}_m^{(t+1)} = \begin{cases} \arg\max \mathbf{p}_m^{(t)}, & \text{if } \widetilde{s}_m^{(t)} = \texttt{[MASK]} \text{ and } \max \mathbf{p}_m^{(t)} > \tau_t, \\ \widetilde{s}_m^{(t)}, & \text{otherwise}. \end{cases} \tag{8}$$

Here, τ_t denotes the stage-specific confidence threshold that controls how aggressively the planner commits to newly denoised fields.

The final planner output is a structured field vector. We render the predicted canvas into canonical phrases that summarize positive findings and device states, and we denote the resulting phrase set by $\mathcal{R}\left(\mathbf{S}\right) = \{a_1, \ldots, a_K\}$, which is conceptually close to recent phrase- or prompt-mediated radiology grounding strategies [3, 7]. This rendering step preserves the clinical semantics of the planner while avoiding degeneration into another free-text generator. Since not every candidate phrase is equally reliable, we score each candidate with its local planner evidence and the context, keeping only the confident abnormal or device phrases as final anchors. This process is expressed as:

$$q_k = \sigma \left(\mathcal{G} \left(\mathbf{h}_{a_k}, \mathbf{h}_{\text{ctx}} \right) \right), \qquad a_k \in \mathcal{R} \left(\mathbf{S} \right). \tag{9}$$

Here, $\mathbf{h}_{a_k}$ denotes the aggregated planner state associated with phrase a_k; $\mathcal{G}$ denotes the phrase scorer; and $\sigma\left(\cdot\right)$ denotes the sigmoid function. These retained phrases become the explicit anchor points that ground the subsequent autoregressive decoding stage.

3.3 Gated Planner-Memory Bridge

To bridge the predicted plannings and the LLM, we propose GPMB. Given the anchor phrases $\mathcal{A}$ and the final planner states $\mathbf{H}_{\text{plan}}$ as the input, GPMB outputs a textual anchor prompt and planner-aware visual states for the autoregressive decoder. Earlier phrase-assisted designs often convert predicted labels into free-text prompts for the decoder [3, 7]. Although this strategy naturally fits the data

structure that LLMs prefer, solely using text narratives fundamentally struggles to represent the field-wise dependency and structures of different pathological attributes, which are highly required by faithful report generation. We therefore construct a dual-channel bridging mechanism to facilitate the autoregressive decoder without sacrificing the structured nature of the planner with GPMB.

Concretely on one hand, let $\mathcal{A} = \{a_{k_1}, \ldots, a_{k_L}\}$ denote the anchor phrase set from DLP. We first incorporate these phrases into the text prompt, giving the LLM decoder an explicit summary of the clinically salient findings. In parallel, we feed the planner states back to calibrate the instance-level visual feature through an adaptive residual bridging network. The intuition is that short phrases are sufficient to tell the decoder what findings matter, but they are not sufficient to preserve how confident the planner is and which latent field interactions support these predictions. We therefore let the bridging network receive both the planner states and the originally extracted visual feature, and adopt a learned gate to automatically adjust how much the planner evidence should enter each visual position. This process is expressed as:

$$\mathbf{g} = \sigma\left(\mathcal{G}_{\text{gate}}\left(\mathbf{H}_{\text{plan}}, \mathbf{Z}\right)\right), \qquad \widehat{\mathbf{V}} = \mathbf{Z} + \mathbf{g} \odot \mathcal{B}_{\text{gate}}\left(\mathbf{H}_{\text{plan}}, \mathbf{Z}\right). \tag{10}$$

Here, $\mathbf{Z} \in \mathbb{R}^{B \times N_{\text{vis}} \times D_{\text{dec}}}$ denotes the decoder-aligned visual tokens before planner injection; $\mathbf{g} \in \mathbb{R}^{B \times N_{\text{vis}} \times 1}$ refers to the learned gate; and $\widehat{\mathbf{V}} \in \mathbb{R}^{B \times N_{\text{vis}} \times D_{\text{dec}}}$ represents the planner-aware visual tokens after bridging. In this way, the bridging network injects structured planner memory into the visual tokens while preserving the original image evidence for the decoder.

3.4 Anchored Causal Decoder

ACD integrates the textual anchors and planner-aware visual states into the LLM decoder, and eventually produces the clinically grounded free-text reports. Given the textual anchors $\mathbf{P}_{\text{text}}$ and the planner-aware visual states $\widehat{\mathbf{V}}$ as the input, ACD outputs the final free-text report $\mathbf{Y}$ in the standard autoregressive manner, following the same high-level decoder-side factorization used by recent LLM-based radiology generators, where this process is expressed as: $p\left(\mathbf{Y} \mid \mathbf{X}\right) = \prod_{t=1}^{T} p\left(y_t \mid y_{<t}, \mathbf{P}_{\text{text}}, \widehat{\mathbf{V}}\right)$. For optimization, we train the full model with a staged objective, which first pre-trains the planner to predict structured information and then jointly updates it with the autoregressive decoder. We use planner-side losses and decoder-side constraints in addition to the standard teacher-forced language modeling loss of the LLM.[1]

4 Experimental Settings

Datasets. We conduct experiments on MIMIC-CXR [14] following prior report-generation studies [15, 16, 19, 1, 3, 7]. MIMIC-CXR is a large-scale chest X-ray benchmark collected at Beth Israel Deaconess Medical Center and released through PhysioNet. It contains 377,110 radiographs from 227,827 imaging studies of 65,379 patients, and each study is paired with a free-text report that includes *findings* and *impression* sections. Following the official patient-level split, we use 270,790, 2,130, and 3,858 report-image pairs for training, validation, and testing, respectively.

To supervise the DLM planner, we derive the structured annotations from the ground-truth reports following the procedures below. For each study, we apply an off-the-shelf label extractor, i.e., CheXbert [37], to identify finding cues from various dimensions, including device mentions, negation or uncertainty markers, and attribute modifiers for severity, laterality, and anatomical region. The extracted evidence is then aggregated into a study-level clinical canvas, normalized with deterministic consistency rules, and finally rendered into canonical planner phrases for phrase-level supervision. Ultimately, this preprocessing produces planner annotations with 35 discrete fields organized into 4 denoising stages, including 3 device fields, 8 finding-status fields, 16 location fields, and 8 severity fields. Across the whole ontology, the planner uses a vocabulary of 31 discrete tokens and a phrase bank of 16 canonical anchors, covering 8 finding-oriented phrases and 8 device-oriented phrases.[2]

[1] In Appendix A.1, we provide the full mathematical formulations of the training objectives.

[2] In Appendix A.2, we describe the planner annotation process and data statistics. In Appendix A.3, we report the supplementary comparison on IU X-RAY.

METHOD	PARAMS (B)	B@1↑	B@2↑	B@3↑	B@4↑	MTR↑	R-L↑	P↑	R↑	F1↑
R2GEN [15]	-	0.353	0.218	0.145	0.103	0.142	0.277	0.333	0.273	0.300
CA [46]	-	0.350	0.219	0.152	0.109	0.151	0.283	–	–	–
PPKED [16]	-	0.360	0.224	0.149	0.106	0.149	0.284	–	–	–
R2GENCMN [47]	-	0.353	0.218	0.148	0.106	0.142	0.278	0.334	0.275	0.302
R2GENRL [48]	-	0.381	0.232	0.155	0.109	0.151	0.287	0.342	0.294	0.316
ITA [49]	-	0.395	0.253	0.170	0.121	0.147	0.284	–	–	–
WARMSTART [50]	-	0.392	0.245	0.169	0.124	0.153	0.285	0.359	0.412	0.384
KIUT [51]	-	0.393	0.243	0.159	0.113	–	0.285	0.371	0.318	0.342
RGRG [18]	-	0.373	0.249	0.175	0.126	0.168	0.264	0.461	0.475	0.468
PROMPTMRG [52]	-	0.398	–	–	0.112	–	0.258	0.501	0.509	0.505
EKAGEN [20]	-	0.419	0.258	0.170	0.119	–	0.287	0.517	0.483	0.499
XRAYGPT [53]	7	0.128	0.045	0.014	0.004	0.079	0.111	–	–	–
R2GENGPT [2]	7	0.396	–	–	0.113	–	0.273	0.506	0.414	0.456
CHEXAGENT [54]	7	0.189	–	–	0.040	–	0.208	0.506	0.306	0.381
MED-PALM [55]	562	0.317	–	–	0.115	–	0.275	–	–	0.378
MAIRA-1 [56]	7	0.392	–	–	0.142	–	0.289	–	–	0.553
R2-LLM [1]	14.2	0.402	0.262	0.180	0.128	0.175	0.291	0.465	0.482	0.473
INVERGE [57]	7	0.425	0.240	0.132	0.100	–	0.309	–	–	–
LLM-RG4 [3]	7	0.377	–	–	0.144	–	0.318	0.583	0.593	0.588
S2D-ALIGN [7]	7	0.422	0.263	0.183	0.149	–	0.332	0.613	0.606	0.608
ANCHOR-GEN (Ours)	**4.0**	**0.427**	**0.266**	**0.186**	**0.151**	**0.177**	**0.335**	**0.616**	**0.609**	**0.612**
DLM DECODER [10]	7	0.282	0.176	0.117	0.081	0.124	0.223	0.336	0.351	0.343
BASE	**3.8**	0.376	0.248	0.174	0.125	0.167	0.266	0.458	0.473	0.465
+TRANS. PLANNER	**3.8**	0.399	0.260	0.179	0.127	0.173	0.289	0.463	0.480	0.471
+LLM PLANNER	7.6	0.388	0.254	0.177	0.126	0.170	0.281	0.460	0.477	0.468
+BERT PLANNER	4.1	0.392	0.262	0.181	0.123	0.169	0.284	0.423	0.414	0.418
+DLM Planner (Ours)	4.0	**0.427**	**0.266**	**0.186**	**0.151**	**0.177**	**0.335**	**0.616**	**0.609**	**0.612**

Table 1: Quantitative comparison (top) of ANCHOR-GEN with existing methods and baseline variants (bottom) on MIMIC-CXR [14] w.r.t. the NLG and CE metrics. Here, we illustrate the parameter count of LLM-based methods in PARAMS (B).

Evaluation Metrics. Following the existing studies, we evaluate the model with two main categories of metrics, i.e., Natural Language Generation (NLG) and Clinical Efficacy (CE) metrics. Specifically, the NLG metric evaluates the lexical coherence of the generated reports, where we use BLEU-1 and BLEU-4 from BLEU [38], ROUGE-L [39], and METEOR [40] in our experiments. The CE metric measures the clinical label consistency between the produced and doctor-annotated reports, where we use CheXbert [37] as the report labeler and compute precision, recall, and F1 as the final scores. For the planner itself, we further report field-level and anchor-level diagnostics in the supplementary analysis to verify whether the predicted anchors are accurate and actually used by the decoder.

Implementation Details. Our implementation uses RAD-DINO [41] as the frozen visual encoder and `Phi-3-mini-4k-instruct` [42] as the autoregressive decoder updated with LoRA adapters [43]. Here, the rank, scaling factor, and dropout rate of the LoRA weight matrix are set to 32, 64, and 0.1. The DLM planner is a 4-layer Transformer [44] with hidden size 768, 8 attention heads, and a feed-forward hidden size of 2,048. For our full model, we conduct experiments with various hyper-parameter settings on the validation set and choose the best-performing setting. Specifically, we use the AdamW optimizer [45] with an initial learning rate of 3×10^{-4} and a batch size of 4, cosine decay with a warmup ratio of 10%. All experiments are executed on a single NVIDIA A40.

5 Main Experiments

Comparison with the Existing Methods. Tab. 1 presents the quantitative comparison of ANCHOR-GEN with existing state-of-the-art methods on MIMIC-CXR. In detail, we summarize this comparison into three main types of methods with respect to their architectural designs, i.e., conventional task-specific methods based on from-scratch Transformer training, LLM-based methods, and the hybrid method represented by ANCHOR-GEN. Conventional task-specific methods obtain limited performance because they are optimized under a closed training distribution and usually lack strong generative priors for long-form reports. LLM-based methods improve the overall results with the assistance of pretrained generative priors, while ANCHOR-GEN achieves the best directly comparable performance on all standard NLG and CE metrics. The gains over the strongest directly compa-

rable baseline are statistically significant under paired bootstrap resampling with $p < 0.05$. This comparison validates the effectiveness of our hybrid design, where non-autoregressive planning first establishes clinical anchors and autoregressive decoding then verbalizes them into faithful reports. In the next subsection, we further detail and analyze the internal functionalities of both the planner and the decoder by comparing ANCHOR-GEN to several baseline variants.

Comparison with Baseline Variants. In Tab. 1, we also compare our full model ANCHOR-GEN to several baseline variants. Specifically, DLM DECODER represents the baseline with an off-the-shelf Diffusion Large Language Models (DLLM), i.e., LLaDA [10], as the language backbone. BASE refers to the standard VLM pipeline tuned for RRG with the same visual encoder and text decoder as ANCHOR-GEN. On top of this, +TRANS. PLANNER adds a Transformer-based planner to autoregressively predict a planning sequence for report generation; +LLM PLANNER further initializes this planner with the same foundation model [42] and LoRA setup as our report generator; +BERT PLANNER models planning as multi-label classification and uses BERT [58] as the planner prior; and +DLM PLANNER represents our full model. First, one can see that simple DLM-based decoder performs worse than LLM-based backbone, confirming our observation where directly applying the non-AR paradigm to RRG fails to replace existing autoregressive report generation manner. Adding a planner improves over BASE, confirming the motivation of anchoring LLM report generation with predicted clinical findings. Second, different planners produce different representation forms: Transformer- and LLM-based planners output clinical findings as text sequences; the BERT planner predicts multiple discrete finding labels; and the DLM planner predicts a structured canvas for the LLM. Transformer- and LLM-based planners bring only small gains, showing that free-form sequential planning alone does not reliably preserve the structured relations among findings. The BERT planner improves several lexical metrics, but its CE scores drop because flat labels cannot preserve the dependency among laterality, severity, and co-occurring abnormalities. By contrast, the DLM planner achieves the best results because its canvas-based denoising process provides field-wise anchors that are both structured and compatible with downstream report generation.[3]

Case Study. Tab. 2 presents a selected abnormal example from the testing split of MIMIC-CXR, together with the radiograph, the reference report, and representative report outputs. The results align well with the quantitative comparisons. Compared with the base model, the BERT planner variant recovers more explicit anchors, e.g., *bilateral pleural effusions*, *basilar atelectasis*, *right basilar opacity*, *possible consolidation*, and *cardiomegaly*, but incorrect observations are still generated in the final results. This pattern suggests that a label-style planner can improve coarse finding coverage, while it remains limited in preserving the full field-wise dependency among laterality, severity, and co-occurring abnormalities, where a structured representation form is ideal to alleviate this issue. By contrast, ANCHOR-GEN covers all major pathological anchors in the reference case and converts them into clinically consistent reports, showing that DLP and GPMB jointly help the decoder follow structured clinical evidence rather than relying only on fluent autoregressive priors.

6 Ablation Studies

Tab. 3 reports two ablations on MIMIC-CXR, investigating the dual-channel bridge and the staged training objective of ANCHOR-GEN.

In Tab. 3a, we ablate GPMB via removing each channel. By bridging the planner and the LLM via a prompt-only variant, this baseline already achieves reasonable results, as the LLM decoder receives explicit anchor phrases in its prompt input. Yet, the improvement remains limited, possibly because increasing pathological complexity enlarges the context length and makes it difficult for the model to jointly attend to all anchor phrases, which in turn suggests that effective use of these anchors requires not only prompt-level guidance but also dedicated architectural adaptation. When we move from prompt-only injection to visual-only injection, both BLEU-4 and F1 increase, indicating that hidden-state fusion preserves richer planner evidence than a short textual summary. When we further combine textual anchors and planner-aware visual states, the full dual-channel GPMB achieves the best result. This row-by-row trend shows that the prompt channel explicitly tells the decoder which findings should be covered, while the visual channel provides architectural access to the latent planner memory that supports these anchors.

[3] In Appendix A.4, we further analyze planner-field accuracy, anchor recall, anchor usage, and hallucination rate.

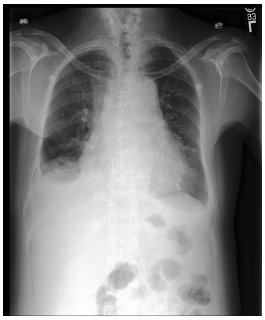
Radiograph	Reports
	**Ground-truth report:** Interval increase/development of **bilateral**, **right greater than left**, pleural effusions with overlying atelectasis. Right base opacity may be due to a combination of pleural effusion and atelectasis; underlying consolidation cannot be excluded. Cardiomegaly and minimal pulmonary vascular congestion are present. **BASE:** A moderate right-sided pleural effusion is present with adjacent atelectasis. A small left pleural effusion is also suspected. Mild right basilar opacity may reflect atelectatic change, and consolidation cannot be excluded. The heart is mildly enlarged. Pulmonary vascular congestion is not clearly mentioned. **+BERT PLANNER:** **Bilateral**, **right-dominant** pleural effusions are seen with overlying basilar atelectasis. There is a right basilar opacity, and possible consolidation remains. The cardiac silhouette is mildly enlarged. No edema is reported. **ANCHOR-GEN:** **Bilateral**, **right-dominant** pleural effusions are present with adjacent basilar atelectatic change. A hazy right basilar opacity accompanies the effusion and atelectasis, and superimposed consolidation remains possible. The cardiac silhouette is enlarged. Minimal pulmonary vascular congestion is also seen, without pneumothorax.

Table 2: A selected case study from the testing set of MIMIC-CXR [14]. Colored backgrounds denote different pathological attributes, green underlined texts mark correct key modifiers, and underlined texts in red refer to incorrect or missed diagnoses.

VARIANTS	B@4↑	F1↑
PROMPT-ONLY	0.141	0.573
VISUAL-ONLY	0.146	0.592
DUAL-CHANNEL (GPMB)	**0.151**	**0.612**

(a) Ablation of the GPMB design.

OBJECTIVE	B@4↑	F1↑
BASE	0.125	0.465
+PLANNER-SIDE LOSS	0.143	0.587
+DECODER-SIDE LOSS	**0.151**	**0.612**

(b) Ablation of the training objectives.

Table 3: Ablations of GPMB and training objectives on MIMIC-CXR w.r.t. BLEU-4 and F1.

Tab. 3b analyzes the objective design by progressively adding the planner- and decoder-side losses.[4] Here, BASE follows the same setting as that in Tab. 1, representing the baseline using a pure autoregressive decoder. On top of the LLM-only baseline, adding planner-side losses brings a clear gain. This improvement shows that it is vital to isolate the training of the planner to ensure its structured estimation ability before joint optimization, as this offers more fine-grained purpose-specific supervision for anchor prediction. When decoder-side losses are further added, the model reaches the best results because the report generator is no longer only conditioned on planner outputs but is also explicitly constrained to preserve and use them. This experiment highlights that planner learning and decoder binding solve two different problems: the former makes the anchors clinically meaningful, while the latter makes these anchors operational during report generation.

7 Conclusion

This paper proposes ANCHOR-GEN, a DLM-anchored report generation framework to improve the clinical faithfulness of conventional large autoregressive decoders for RRG. Specifically, ANCHOR-GEN introduces the DLP to capture structured findings as anchor points for the LLM; the GPMB to bridge the planner and the LLM; and the ACD to verbalize the structured findings into free-text, cohesive reports. Extensive experiments and ablations show that structured planning, dual-channel bridging, and planner-decoder binding offer an effective alternative for the RRG paradigm. However, the current framework still depends on a pre-defined clinical ontology and on structured labels derived from reports, which limits the coverage of the planner for rare, ambiguous, or more nuanced findings. Future work should extend this framework to richer ontologies, stronger planner supervision, and broader radiology settings, where we believe explicit diffusion-based planning can serve as a reference for building more faithful and more controllable medical generation systems.

[4] In Appendix A.5, we provide a fine-grained ablation study on each used loss function.

References

[1] Chang Liu, Yuanhe Tian, Weidong Chen, Yan Song, and Yongdong Zhang. Bootstrapping Large Language Models for Radiology Report Generation. In *AAAI*, pages 18635–18643, 2024.

[2] Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. R2GenGPT: Radiology Report Generation with Frozen LLMs. *arXiv preprint arXiv:2309.09812*, 2023.

[3] Zhuhao Wang, Yihua Sun, Zihan Li, Xuan Yang, Fang Chen, and Hongen Liao. LLM-RG4: Flexible and Factual Radiology Report Generation Across Diverse Input Contexts. In *AAAI*, pages 8250–8258, 2025.

[4] Tengfei Liu, Jiapu Wang, Yongli Hu, Mingjie Li, Junfei Yi, Xiaojun Chang, Junbin Gao, and Baocai Yin. HC-LLM: Historical-Constrained Large Language Models for Radiology Report Generation. In *AAAI*, pages 5595–5603, 2025.

[5] Shruthi Bannur, Kenza Bouzid, Daniel C. Castro, Anton Schwaighofer, Sam Bond-Taylor, Maximilian Ilse, Fernando Pérez-García, Valentina Salvatelli, Harshita Sharma, Felix Meissen, Mercy Ranjit, Shaury Srivastav, Julia Gong, Fabian Falck, Ozan Oktay, Anja Thieme, Matthew P. Lungren, Maria Teodora Wetscherek, Javier Alvarez-Valle, and Stephanie L. Hyland. MAIRA-2: Grounded Radiology Report Generation. *arXiv preprint arXiv:2406.04449*, 2024.

[6] Xi Zhang, Zaiqiao Meng, Jake Lever, and Edmond S. L. Ho. Libra: Leveraging Temporal Images for Biomedical Radiology Analysis. In *Findings of ACL*, pages 17275–17303, 2025.

[7] Jiechao Gao, Chang Liu, and Yuangang Li. S2D-Align: Shallow-to-Deep Auxiliary Learning for Anatomically-Grounded Radiology Report Generation. In *AAAI*, pages 30780–30788, 2026.

[8] Samar Khanna, Siddhant Kharbanda, Shufan Li, Harshit Varma, Eric Wang, Sawyer Birnbaum, Ziyang Luo, Yanis Miraoui, Akash Palrecha, Stefano Ermon, Aditya Grover, and Volodymyr Kuleshov. Mercury: Ultra-Fast Language Models Based on Diffusion. *arXiv preprint arXiv:2506.17298*, 2025.

[9] Zebin You, Shen Nie, Xiaolu Zhang, Jun Hu, Jun Zhou, Zhiwu Lu, Ji-Rong Wen, and Chongxuan Li. LLaDA-V: Large Language Diffusion Models with Visual Instruction Tuning. *arXiv preprint arXiv:2505.16933*, 2025.

[10] 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.

[11] Lifeng Chen, Tianqi You, Hao Liu, Zhimin Bao, Jile Jiao, Xiao Han, Zhicai Ou, Tao Sun, Xiaofeng Mou, Xiaojie Jin, and Yi Xu. ECHO: Efficient Chest X-Ray Report Generation with One-Step Block Diffusion. *arXiv preprint arXiv:2604.09450*, 2026.

[12] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM Improves Controllable Text Generation. In *NeurIPS*, 2022.

[13] Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Fei Huang, and Songfang Huang. SeqDiffuSeq: Text Diffusion with Encoder-Decoder Transformers. *arXiv preprint arXiv:2212.10325*, 2022.

[14] Alistair E. W. Johnson, Tom J. Pollard, Seth J. Berkowitz, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih ying Deng, Roger G. Mark, and Steven Horng. MIMIC-CXR: A Large Publicly Available Database of Labeled Chest Radiographs. *arXiv preprint arXiv:1901.07042*, 2019.

[15] Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. Generating Radiology Reports via Memory-Driven Transformer. In *EMNLP*, pages 1439–1449, 2020.

[16] Fenglin Liu, Xian Wu, Shen Ge, Wei Fan, and Yuexian Zou. Exploring and Distilling Posterior and Prior Knowledge for Radiology Report Generation. In *CVPR*, pages 13753–13762, 2021.

[17] Hoang T. N. Nguyen, Dong Nie, Taivanbat Badamdorj, Yujie Liu, Yingying Zhu, Jason Truong, and Li Cheng. Automated Generation of Accurate and Fluent Medical X-ray Reports. In *EMNLP*, pages 3552–3569, 2021.

[18] Tim Tanida, Philip Müller, Georgios Kaissis, and Daniel Rueckert. Interactive and Explainable Region-guided Radiology Report Generation. In *CVPR*, pages 7433–7442, 2023.

[19] Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. METransformer: Radiology Report Generation by Transformer with Multiple Learnable Expert Tokens. In *CVPR*, pages 11558–11567, 2023.

[20] Shenshen Bu, Taiji Li, Yuedong Yang, and Zhiming Dai. Instance-level Expert Knowledge and Aggregate Discriminative Attention for Radiology Report Generation. In *CVPR*, pages 14194–14204, 2024.

[21] Sang-Jun Park, Keun-Soo Heo, Dong-Hee Shin, Young-Han Son, Ji-Hye Oh, and Tae-Eui Kam. DART: Disease-Aware Image-Text Alignment and Self-Correcting Re-Alignment for Trustworthy Radiology Report Generation. In *CVPR*, pages 15580–15589, 2025.

[22] Fuying Wang, Shenghui Du, and Lequan Yu. HERGen: Elevating Radiology Report Generation with Longitudinal Data. In *ECCV*, pages 183–200, 2024.

[23] Kang Liu, Zhuoqi Ma, Xiaolu Kang, Yunan Li, Kun Xie, Zhicheng Jiao, and Qiguang Miao. Enhanced Contrastive Learning with Multi-View Longitudinal Data for Chest X-Ray Report Generation. In *CVPR*, pages 10348–10359, 2025.

[24] Ting Xiao, Lei Shi, Yang Zhang, HaoFeng Yang, Zhe Wang, and Chenjia Bai. Online Iterative Self-Alignment for Radiology Report Generation. In *ACL*, pages 27799–27814, 2025.

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *NeurIPS*, 2020.

[26] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured Denoising Diffusion Models in Discrete State-Spaces. In *NeurIPS*, pages 17981–17993, 2021.

[27] Lin Zheng, Jianbo Yuan, Lei Yu, and Lingpeng Kong. A Reparameterized Discrete Diffusion Model for Text Generation. *arXiv preprint arXiv:2302.05737*, 2023.

[28] Kun Zhou, Yifan Li, Xin Zhao, and Ji-Rong Wen. Diffusion-NAT: Self-Prompting Discrete Diffusion for Non-Autoregressive Text Generation. In *EACL*, pages 1438–1451, 2024.

[29] Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and Effective Masked Diffusion Language Models. In *NeurIPS*, 2024.

[30] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models. In *ICLR*, 2023.

[31] Zhenghao Lin, Yeyun Gong, Yelong Shen, Tong Wu, Zhihao Fan, Chen Lin, Nan Duan, and Weizhu Chen. Text Generation with Diffusion Language Models: A Pre-training Approach with Continuous Paragraph Denoise. In *ICML*, pages 21051–21064, 2023.

[32] Ting Chen, Ruixiang Zhang, and Geoffrey E. Hinton. Analog Bits: Generating Discrete Data using Diffusion Models with Self-Conditioning. In *ICLR*, 2023.

[33] Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. MMaDA: Multimodal Large Diffusion Language Models. *arXiv preprint arXiv:2505.15809*, 2025.

[34] Minkai Xu, Tomas Geffner, Karsten Kreis, Weili Nie, Yilun Xu, Jure Leskovec, Stefano Ermon, and Arash Vahdat. Energy-Based Diffusion Language Models for Text Generation. In *ICLR*, 2025.

[35] Xiaochen Zhu, Georgi Karadzhov, Chenxi Whitehouse, and Andreas Vlachos. Segment-Level Diffusion: A Framework for Controllable Long-Form Generation with Diffusion Language Models. In *ACL*, pages 4163–4183, 2025.

[36] Ying Zhou, Xinyao Wang, Yulei Niu, Yaojie Shen, Lexin Tang, Fan Chen, Ben He, Le Sun, and Longyin Wen. DiffLM: Controllable Synthetic Data Generation via Diffusion Language Models. In *Findings of ACL*, pages 20638–20658, 2025.

[37] Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y. Ng, and Matthew P. Lungren. CheXbert: Combining automatic labelers and expert annotations for accurate radiology report labeling using BERT. *arXiv preprint arXiv:2004.09167*, 2020.

[38] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL*, pages 311–318, 2002.

[39] Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, 2004.

[40] Alon Lavie and Abhaya Agarwal. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *WMT@ACL*, pages 228–231, 2007.

[41] Fernando Pérez-García, Harshita Sharma, Sam Bond-Taylor, Kenza Bouzid, Valentina Salvatelli, Maximilian Ilse, Shruthi Bannur, Daniel C. Castro, Anton Schwaighofer, Matthew P. Lungren, Maria Wetscherek, Noel Codella, Stephanie L. Hyland, Javier Alvarez-Valle, and Ozan Oktay. RAD-DINO: Exploring Scalable Medical Image Encoders Beyond Text Supervision. *arXiv preprint arXiv:2401.10815*, 2024.

[42] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Joao P. Barreiro, Hany ElHoushi, Khaled Kourakos, Vignesh Manoharan, Aaron Mathews, Youssef Mehdad, Surya Narayan, Yara Rizk, Huseyin S. Sayed, Marcel Smorkalov, Zhiruo Wang, Muhammad F. Babar, and Rami Al-Rfou. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. *arXiv preprint arXiv:2404.14219*, 2024.

[43] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*, 2022.

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *NeurIPS*, pages 5998–6008, 2017.

[45] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2019.

[46] Fenglin Liu, Changchang Yin, Xian Wu, Shen Ge, Ping Zhang, and Xu Sun. Contrastive Attention for Automatic Chest X-ray Report Generation. In *Findings of ACL-IJCNLP*, pages 269–280, 2021.

[47] Zhihong Chen, Yaling Shen, Yan Song, and Xiang Wan. Cross-modal Memory Networks for Radiology Report Generation. In *ACL-IJCNLP*, pages 5904–5914, 2021.

[48] Han Qin and Yan Song. Reinforced Cross-modal Alignment for Radiology Report Generation. In *Findings of ACL*, pages 448–458, 2022.

[49] Lin Wang, Munan Ning, Donghuan Lu, Dong Wei, Yefeng Zheng, and Jie Chen. An Inclusive Task-Aware Framework for Radiology Report Generation. In *MICCAI*, pages 568–577, 2022.

[50] Aaron Nicolson, Jason Dowling, and Bevan Koopman. Improving Chest X-Ray Report Generation by Leveraging Warm Starting. *Artificial Intelligence in Medicine*, 2023.

[51] Zhongzhen Huang, Xiaofan Zhang, and Shaoting Zhang. KiUT: Knowledge-injected U-Transformer for Radiology Report Generation. *arXiv preprint arXiv:2306.11345*, 2023.

[52] Haibo Jin, Haoxuan Che, Yi Lin, and Hao Chen. PromptMRG: Diagnosis-Driven Prompts for Medical Report Generation. In *AAAI*, pages 2607–2615, 2024.

[53] Omkar Thawkar, Abdelrahman Shaker, Sahal Shaji Mullappilly, Hisham Cholakkal, Rao Muhammad Anwer, Salman Khan, Jorma Laaksonen, and Fahad Shahbaz Khan. XrayGPT: Chest Radiographs Summarization using Medical Vision-Language Models. *arXiv preprint arXiv:2306.07971*, 2024.

[54] Zhihong Chen, Maya Varma, Justin Xu, Magdalini Paschali, Dave Van Veen, Andrew Johnston, Alaa Youssef, Louis Blankemeier, Christian Bluethgen, Stephan Altmayer, Jeya Maria Jose Valanarasu, Mohamed Siddig Eltayeb Muneer, Eduardo Pontes Reis, Joseph Paul Cohen, Cameron Olsen, Tanishq Mathew Abraham, Emily B. Tsai, Christopher F. Beaulieu, Jenia Jitsev, Sergios Gatidis, Jean-Benoit Delbrouck, Akshay S. Chaudhari, and Curtis P. Langlotz. A Vision-Language Foundation Model to Enhance Efficiency of Chest X-ray Interpretation. *arXiv preprint arXiv:2401.12208*, 2024.

[55] Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Chuck Lau, Ryutaro Tanno, Ira Ktena, Basil Mustafa, Aakanksha Chowdhery, Yun Liu, Simon Kornblith, David Fleet, Philip Mansfield, Sushant Prakash, Renee Wong, Sunny Virmani, Christopher Semturs, S Sara Mahdavi, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Karan Singhal, Pete Florence, Alan Karthikesalingam, and Vivek Natarajan. Towards Generalist Biomedical AI. *arXiv preprint arXiv:2307.14334*, 2023.

[56] Stephanie L. Hyland, Shruthi Bannur, Kenza Bouzid, Daniel C. Castro, Mercy Ranjit, Anton Schwaighofer, Fernando Pérez-García, Valentina Salvatelli, Shaury Srivastav, Anja Thieme, Noel Codella, Matthew P. Lungren, Maria Teodora Wetscherek, Ozan Oktay, and Javier Alvarez-Valle. MAIRA-1: A Specialised Large Multimodal Model for Radiology Report Generation. *arXiv preprint arXiv:2311.13668*, 2023.

[57] Ankan Deria, Komal Kumar, Snehashis Chakraborty, Dwarikanath Mahapatra, and Sudipta Roy. InVERGe: Intelligent Visual Encoder for Bridging Modalities in Report Generation. In *CVPRW*, pages 2028–2038, 2024.

[58] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*, pages 4171–4186, 2019.

[59] Fenglin Liu, Shen Ge, and Xian Wu. Competence-based Multimodal Curriculum Learning for Medical Report Generation. In *ACL-IJCNLP*, pages 3001–3012, 2021.

[60] Dina Demner-Fushman, Marc D. Kohli, Marc B. Rosenman, Sonya E. Shooshan, Laritza Rodriguez, Sameer K. Antani, George R. Thoma, and Clement J. McDonald. Preparing a Collection of Radiology Examinations for Distribution and Retrieval. *J. Am. Medical Informatics Assoc.*, 23(2):304–310, 2016.

A Supplementary Material

We construct our supplementary material following the structures below:

- Appendix A.1 details the complete optimization objectives of ANCHOR-GEN, including planner denoising, hierarchical prior supervision, phrase scoring, language modeling, planner-decoder consistency, prompt usage, and staged objective composition.
- Appendix A.2 describes the annotation process of the DLP, including the field definitions used by the clinical canvas and the rendering rules for canonical anchor phrases.
- Appendix A.3 reports the supplementary IU X-RAY comparison for evaluating the cross-dataset generality of ANCHOR-GEN.
- Appendix A.4 reports planner-specific evaluation that connect anchor prediction quality to downstream report faithfulness.
- Appendix A.5 provides a fine-grained ablation study on the loss function setting, explaining how different objectives contribute to the final model.

A.1 Optimization Objectives

This subsection provides the exact training objectives used by ANCHOR-GEN. We first summarize the objective family and then introduce each loss separately:

- The planner-side losses include the stage-wise masked infill loss $\mathcal{L}_{\text{plan}}$, the hierarchical prior loss $\mathcal{L}_{\text{prior}}$, and the phrase-scoring loss $\mathcal{L}_{\text{phrase}}$. These losses train the DLP to recover the structured clinical canvas, stabilize the coarse abnormality and device decisions, and select reliable anchor phrases before the autoregressive decoder is attached.
- The LLM-side losses include the teacher-forced language modeling loss $\mathcal{L}_{\phi}$, the planner-report consistency loss $\mathcal{L}_{\text{cons}}$, and the prompt-usage loss $\mathcal{L}_{\text{usage}}$. These losses train the decoder to generate fluent reports while preserving and using the planner anchors.

Stage-wise masked infill loss. The first planner-side objective supervises the DLP to reconstruct masked fields at different denoising stages. Let $\mathcal{T}_{\text{den}}$ denote the ordered denoising stages; $\mathcal{F}_r$ denote the active fields in stage r; and w_r denote the stage weight. The loss is defined as:

$$\mathcal{L}_{\text{plan}} = -\frac{1}{\sum_{r\in\mathcal{T}_{\text{den}}} w_r} \sum_{r\in\mathcal{T}_{\text{den}}} w_r \frac{1}{|\mathcal{F}_r|} \sum_{m\in\mathcal{F}_r} \sum_{c\in\mathcal{V}_m} \mathbb{I}\left[c = s_m\right] \log \text{Softmax}\left(\mathbf{z}_m^{(r)}\right)_c . \tag{11}$$

Here, $\mathbf{z}_m^{(r)}$ denotes the logit vector of field f_m at denoising stage r; $\mathcal{V}_m$ denotes the candidate value set of field f_m; and $\mathbb{I}\left[\cdot\right]$ denotes the indicator function. This objective uses field-wise cross-entropy over the masked clinical canvas.

Hierarchical prior loss. The second planner-side objective supervises the coarse prior before full denoising starts, so that the later canvas updates begin from plausible global abnormality and device states. Given the coarse field set $\mathcal{F}_{\text{coarse}}$, this loss is defined as:

$$\mathcal{L}_{\text{prior}} = -\frac{1}{|\mathcal{F}_{\text{coarse}}|} \sum_{m\in\mathcal{F}_{\text{coarse}}} \omega_m^{\text{prior}} \sum_{c\in\mathcal{V}_m} \mathbb{I}\left[c = s_m\right] \log p_{\text{prior},c}^{(m)}. \tag{12}$$

Here, ω_m^{prior} denotes the field-dependent reweighting factor that upweights abnormal and device-related coarse states, and $p_{\text{prior},c}^{(m)}$ denotes the prior probability of assigning field f_m to class c. This objective is also implemented as weighted cross-entropy because each coarse field takes one value from a finite vocabulary.

Phrase-scoring loss. The third planner-side objective calibrates which rendered planner phrases should be passed to the decoder. For the candidate phrase set $\mathcal{R}\left(\mathbf{S}\right) = \{a_1, \ldots, a_K\}$, this binary objective is defined as:

$$\mathcal{L}_{\text{phrase}} = -\frac{1}{K} \sum_{k=1}^{K} \omega_k^{\text{phrase}} \left[y_k^{\text{phrase}} \log q_k + \left(1 - y_k^{\text{phrase}}\right) \log\left(1 - q_k\right) \right]. \tag{13}$$

Here, $y_k^{\text{phrase}} \in \{0, 1\}$ indicates whether anchor a_k should be retained as a valid planner phrase. This objective uses weighted binary cross-entropy, where positive phrases are derived from canonical anchor phrases matched to the silver clinical canvas.

Teacher-forced language modeling loss. The primary LLM-side objective trains the autoregressive decoder to generate the reference report conditioned on the textual anchors $\mathbf{P}_{\text{text}}$ and planner-aware visual states $\widehat{\mathbf{V}}$. This standard teacher-forced language modeling loss is defined as:

$$\mathcal{L}_{\phi} = -\sum_{t=1}^{T} \log p\left(y_t \mid y_{<t}, \mathbf{P}_{\text{text}}, \widehat{\mathbf{V}}\right). \tag{14}$$

Here, y_t denotes the t-th target report token, and $y_{<t}$ denotes the previously observed target tokens under teacher forcing. This objective is the standard token-level cross-entropy used by the autoregressive decoder.

Planner-report consistency loss. The second LLM-side objective requires the decoder states to preserve the clinical content predicted by the planner. Let $\widehat{\mathbf{p}}_m^{\text{dec}}$ denote the field-wise prediction

decoded from the report hidden states; and let $\mathcal{F}_{\text{cons}}$ denote the set of planner fields supervised by the consistency head. This loss is defined as:

$$\mathcal{L}_{\text{cons}} = -\frac{1}{|\mathcal{F}_{\text{cons}}|} \sum_{m \in \mathcal{F}_{\text{cons}}} \omega_m^{\text{cons}} \sum_{c \in \mathcal{V}_m} \mathbb{I}\left[c = s_m\right] \log \widehat{p}_{m,c}^{\text{dec}}. \tag{15}$$

Here, ω_m^{cons} denotes the consistency weight of field f_m, and $\widehat{p}_{m,c}^{\text{dec}}$ denotes the decoder-side probability of assigning field f_m to class c. This objective uses weighted cross-entropy to align decoder-side clinical predictions with the planner canvas.

Prompt-usage loss. The third LLM-side objective explicitly supervises whether the decoder uses the selected planner anchors in the generated report. Let u_k denote the predicted usage probability of anchor a_k in the decoder, and let $y_k^{\text{use}} \in \{0, 1\}$ denote its gold usage target. This loss is defined as:

$$\mathcal{L}_{\text{usage}} = -\frac{1}{K} \sum_{k=1}^{K} \omega_k^{\text{use}} \left[y_k^{\text{use}} \log u_k + (1 - y_k^{\text{use}}) \log (1 - u_k) \right]. \tag{16}$$

Here, u_k denotes the predicted probability that anchor a_k is used by the decoder; and y_k^{use} denotes the binary usage label. We construct y_k^{use} by normalizing the reference report and checking whether the clinical concept behind a_k is expressed by a matched mention or a synonym-preserving canonical phrase. This objective uses weighted binary cross-entropy and directly penalizes selected anchors that disappear from the generated report.

Overall staged objective. Finally, we combine the planner-side and LLM-side losses with a stage-dependent coefficient $\alpha(\tau)$, where $\alpha(\tau) = 0$ in the planner-only stage, $\alpha(\tau) \in (0, 1)$ in the ramp stage, and $\alpha(\tau) = 1$ in the final joint stage. The full training objective is defined as:

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{\text{plan}} &+ \lambda_{\text{prior}} \mathcal{L}_{\text{prior}} + \lambda_{\text{phrase}} \mathcal{L}_{\text{phrase}} \\ &+ \alpha(\tau) \left(\mathcal{L}_{\phi} + \lambda_{\text{cons}} \mathcal{L}_{\text{cons}} + \lambda_{\text{usage}} \mathcal{L}_{\text{usage}} \right). \end{aligned} \tag{17}$$

This staged design first makes the planner itself reliable, then progressively exposes the decoder to planner-driven supervision, and finally performs full joint optimization.

A.2 Slot Ontology and Silver-label Construction

The planner uses a structured field ontology that includes global abnormality, device state, and finding-specific status, laterality, region, and severity fields. This subsection provides the exact ontology table, the rendering rules used for canonical prompt phrases, and a brief audit of the silver-label quality.

To supervise the DLM planner, we derive the structured annotations from the ground-truth reports. For each study, we apply an off-the-shelf label extractor, i.e., CheXbert [37], to identify finding cues from various dimensions, including device mentions, negation or uncertainty markers, and attribute modifiers for severity, laterality, and anatomical region. The extracted evidence is aggregated into a study-level clinical canvas, normalized with deterministic consistency rules, and rendered into canonical planner phrases for phrase-level supervision. Ultimately, this preprocessing produces planner annotations with 35 discrete fields organized into 4 denoising stages, including 3 device fields, 8 finding-status fields, 16 location fields, and 8 severity fields. Across the whole ontology, the planner uses a vocabulary of 31 discrete tokens and a phrase bank of 16 canonical anchors, covering 8 finding-oriented phrases and 8 device-oriented phrases.

We build silver labels with three deterministic steps. First, CheXbert identifies coarse finding states from the reference report. Second, rule-based span matching maps modifiers such as laterality, anatomical region, and severity to the corresponding finding fields. Third, the completed canvas is rendered into canonical anchors by keeping positive findings and support-device states, removing contradictory phrases, and merging repeated mentions. The final audit checks whether each rendered phrase can be traced back to at least one normalized report mention. This design keeps the planner supervision reproducible while avoiding free-form phrase generation during annotation.

Table 4: Illustration of the planner ontology.

STAGE	FIELD IDS	FIELD FAMILY	CURRENT VALUES
COARSE PRIOR	1–3	Device fields	None, Present, Absent, Uncertain; Endotracheal tube, Nasogastric tube, central venous catheter, PICC line, pacemaker, chest tube, postoperative device, generic device
FINDING STATUS	4–11	Status fields	None, Present, Absent, Uncertain
FINDING LOCATION	12–27	Laterality and region fields	None, Left, Right, Bilateral, Diffuse; Basilar, lower zone, upper zone, apical, retrocardiac, perihilar
FINDING SEVERITY	28–35	Severity fields	None, Trace, Mild, Moderate, Severe

METHOD	PARAMS (B)	B@1↑	B@2↑	B@3↑	B@4↑	MTR↑	R-L↑
R2GEN [15]	-	0.470	0.304	0.219	0.165	–	0.371
CA [46]	-	0.492	0.314	0.222	0.169	0.193	0.381
PPKED [16]	-	0.483	0.315	0.224	0.168	–	0.376
R2GENCMN [47]	-	0.475	0.309	0.222	0.170	0.191	0.375
R2GENRL [48]	-	0.494	0.321	0.235	0.181	0.201	0.384
CMCL [59]	-	0.473	0.305	0.217	0.162	0.186	0.378
XRAYGPT [53]	7	0.177	0.104	0.047	0.007	0.105	0.203
R2-LLM [1]	14.2	0.499	0.323	0.238	0.184	0.208	0.390
S2D-ALIGN [7]	7	0.512	–	–	0.195	–	0.407
ANCHOR-GEN (OURS)	**4.0**	**0.519**	**0.331**	**0.242**	**0.199**	**0.213**	**0.413**

Table 5: Quantitative comparison on IU X-RAY [60] with respect to the NLG metrics.

VARIANT	FIELD ACC.↑	ANCHOR P↑	ANCHOR R↑	USAGE↑	HALLUC.↓
+BERT PLANNER	0.642	0.713	0.668	0.623	0.204
+DLM PLANNER	0.734	0.792	0.781	0.762	0.126
FULL +GPMB	**0.746**	**0.811**	**0.798**	**0.824**	**0.091**

Table 6: Planner-specific evaluation on MIMIC-CXR. FIELD ACC. evaluates the complete clinical canvas, ANCHOR P and ANCHOR R evaluate retained canonical phrases, USAGE measures whether selected anchors are covered by the generated report, and HALLUC. measures abnormal anchors.

A.3 Supplementary Results on IU X-Ray

Tab. 5 reports the quantitative comparison on IU X-RAY with existing methods. Similar trends are observed as that in Tab. 1, where we obtain the best results among the compared methods, suggesting that DLP provides transferable clinical anchors beyond pure autoregressive report generation.

A.4 Planner Diagnostics

Planner diagnostics measure whether the DLP predicts clinically useful anchors and whether these anchors are preserved by the decoder. We report FIELD ACC. for the complete canvas, ANCHOR P and ANCHOR R for retained canonical phrases, USAGE for phrase coverage in the final report, and HALLUC. for generated abnormal anchors unsupported by the silver labels.

The diagnostics show a clear progression from label-style planning to diffusion language planning. The BERT planner predicts several coarse labels correctly, but its ANCHOR R and USAGE remain limited because flat classification does not model the dependency between findings and their attributes.

SETTING	$\mathcal{L}_{\phi}$	$\mathcal{L}_{\text{plan}}$	$\mathcal{L}_{\text{prior}}$	$\mathcal{L}_{\text{phrase}}$	$\mathcal{L}_{\text{cons}}$	$\mathcal{L}_{\text{usage}}$	B@4↑	F1↑
LLM ONLY	✓						0.125	0.465
+CANVAS DENOISING	✓	✓					0.136	0.548
+HIERARCHICAL PRIOR	✓	✓	✓				0.140	0.574
+PHRASE SCORING	✓	✓	✓	✓			0.143	0.587
+CONSISTENCY	✓	✓	✓	✓	✓		<u>0.148</u>	<u>0.602</u>
FULL OBJECTIVE	✓	✓	✓	✓	✓	✓	**0.151**	**0.612**

Table 7: Ablation for the loss function setting on MIMIC-CXR.

The DLM planner improves both ANCHOR P and ANCHOR R, indicating that masked denoising on a structured canvas better recovers the intended clinical anchors. After adding GPMB, USAGE further increases and HALLUC. decreases, showing that the bridge does not merely expose more anchors to the decoder but makes these anchors actionable during generation.

A.5 Fine-grained Loss Ablation

This subsection further decomposes the objective design used in Tab. 3b. Specifically, the first row uses only the language modeling objective and therefore behaves like a standard autoregressive report generator. Adding $\mathcal{L}_{\text{plan}}$ improves F1 more than BLEU-4, which suggests that learning the canvas primarily improves clinical content coverage. Adding $\mathcal{L}_{\text{prior}}$ and $\mathcal{L}_{\text{phrase}}$ further stabilizes coarse abnormality decisions and filters unreliable anchors, leading to a more reliable planner output before the decoder is attached. The final two rows add $\mathcal{L}_{\text{cons}}$ and $\mathcal{L}_{\text{usage}}$, showing that decoder-side constraints are necessary for converting correct anchors into faithful report text. This trend indicates that ANCHOR-GEN requires both reliable diffusion planning and explicit planner-decoder binding.

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 Sec. 1 state the core contribution as a DLM-based planner combined with an autoregressive LLM decoder for radiology report generation.

   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: Sec. 7 explicitly states that the current framework depends on a pre-defined clinical ontology and report-derived structured labels, which may limit coverage for rare, ambiguous, or nuanced findings. This limitation will also serve as our direction for future works.

   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: [N/A].

Justification: The paper does not present new theoretical theorems or proof-based claims.

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 method, dataset split, metrics, major model components, and key optimization settings are described in Secs. 3–4 and Appendix A.1.

   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: The paper uses the public credentialed MIMIC-CXR benchmark and cites the dataset source.

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: Sec. 4 reports the MIMIC-CXR split, planner annotation construction, evaluation metrics, backbone choices, LoRA settings, DLM planner size, optimizer, learning rate, batch size, and scheduler policy. These details are sufficient to understand the reported comparisons.

   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: All results pass statistical significance tests with $p < 0.05$.

   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: Sec. **??** describes the main optimization setup and the hardware environment of the used GPU resources.

   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 study uses the de-identified MIMIC-CXR benchmark through its established research-access protocol and focuses on algorithmic report generation rather than autonomous clinical deployment. The paper does not introduce identifiable patient data or new human-subject interventions.

   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: [N/A].

    Justification: There is no societal impact of the work performed.

    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., deploy-

ment 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 current paper does not release a new high-risk model, scraped dataset, or patient-level asset. The work relies on the existing controlled-access MIMIC-CXR benchmark and does not describe an external model release.

    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: The paper cites the used existing assets, including MIMIC-CXR, CheXbert, RAD-DINO, the LLM backbone (Phi-3), LoRA, and the compared methods.

    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: [N/A].

    Justification: Sec. 4 and Appendix A.2 describe internally constructed planner annotations for training, but the current submission does not release a new dataset, code package, model checkpoint, or benchmark asset.

    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: The paper does not conduct crowdsourcing experiments and does not recruit new human subjects. It uses an existing de-identified public benchmark for retrospective model evaluation.

    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: The work does not involve newly recruited participants or a new human-subject study. It uses MIMIC-CXR as a de-identified benchmark released for research access.

    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: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    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.