SAT-RRG: Self-Adaptive Training for Radiology Report Generation Leveraging LLMs for Dynamic Token-Level Refinement

Anonymous ACL submission

Abstract

Existing radiology report generation (RRG) methods are predominantly trained using cross-entropy (CE) loss, which only reinforces tokens that exactly match the reference report. As a result, these models overlook semantically meaningful alternatives and fail to recognize or correct factual errors. Also, these methods lack semantic feedback mechanisms during training, treating all tokens uniformly and failing to prioritize critical corrections. We propose **SAT-RRG**, a self-adaptive training framework that leverages the underlying LLM in the RRG model to dynamically identify semantically incorrect tokens during training—without requiring human annotation. Based on these pseudo-labeled error spans, we introduce two token-level losses: **CTAL** reinforces confidently correct predictions, while **ETAPL** penalizes overconfident semantic errors. This dual supervision allows the model to focus learning where it matters most: strengthening reliable tokens and correcting factual mistakes. SAT-RRG builds upon a unified LLM backbone for both generation and error detection, incurs no additional inference-time cost and eliminates the need for manual error labels. SAT-RRG achieves state-of-the-art performance on MIMIC-CXR and IU-Xray. Code will be released upon publication.

1 Introduction

Automating radiology report generation (RRG) has gained attention for its potential to reduce radiologists' workload and improve diagnostic consistency. Recent advancements in deep learning, particularly with encoder-decoder architectures and large language models (LLMs) (Vaswani, 2017; Chen et al., 2020, 2022; Wang et al., 2022a; Huang et al., 2023; Bu et al., 2024), have shown promising results in generating textual descriptions from radiological images. These models predict reports word by word, treating each word as a separate class and assigning probabilities at each step. Training typically relies on cross-entropy (CE) loss, which maximizes the likelihood of selecting the correct word based on a reference report.

However, CE loss enforces strict word-level alignment between generated and reference reports, making existing models overly sensitive to exact phrasing while failing to recognize meaning-equivalent variations. Moreover, existing RRG methods generally lack a feedback mechanism to assess semantic quality during training, such as evaluating whether the generated content is textually and clinically consistent with the reference report, let alone leveraging such feedback to guide model updates.

In the literature, token-level feedback has been explored in general NLG via reinforcement learning (Ziegler et al., 2019), contrastive objectives (Liu et al., 2024a), or correction-based post-hoc tuning. However these mechanisms typically require either manually designed reward signals or additional labeled data. However, they cannot be effectively applied to the RRG task due to the high cost of expert annotation.

To address this situation, we propose a novel training framework that introduces a more adaptive learning process for RRG. Instead of resorting to external reward models or annotations, we tactically exploit the LLM in an RRG task for its inherent semantic understanding capability to obtain dynamic, token-level supervision during training. Since the LLM is already part of the RRG model, our method imposes no extra structure: the model itself evaluates and generates online token-level labels during the training process, identifying semantic inconsistencies between its predictions and the ground truth report.

Specifically, we design an LLM-based mechanism to locate semantically mismatched tokens between generated and reference reports. Unlike existing methods, our approach dynamically ad-

justs token-level probabilities, allowing the model to focus on correcting specific errors while reinforcing correct predictions. To achieve this, two custom loss functions are designed: the **Correct Token Augment Loss (CTAL)**, which reduces updates for correct tokens to preserve stability, and the **Error Token Adaptive Penalty Loss (ETAPL)**, which penalizes incorrect tokens to enforce targeted correction. Together, the two losses support a balanced optimization that encourages accuracy while enabling adaptive self-correction at the token level.

Our contributions are summarized as follows:
First, we introduce a novel RRG training framework that leverages the LLM's semantic understanding capability to dynamically refine token-level predictions. By pinpointing where the model needs stronger updates, it enables a more informed learning process, focusing on error correction while preserving fluency and coherence. Unlike conventional word-level alignment with reference reports, our approach identifies error tokens at the phrase level, better capturing semantic inconsistencies.
Second, to realize our framework, we propose two custom loss functions: Correct Token Augment Loss (CTAL), which minimizes updates for confidently predicted correct tokens, reinforcing stable predictions, and Error Token Adaptive Penalty Loss (ETAPL), which intensifies corrections adaptively for errors that are predicted with high-confidence, ensuring more precise learning.
Third, Our method achieves state-of-the-art performance on MIMIC-CXR and IU-Xray. By using the same LLM backbone for both report generation and error detection, it avoids significant computational overhead and preserves inference efficiency.

2 Related Work

Radiology Report Generation. Radiology report generation (RRG) aims to produce clinically accurate diagnostic narratives from radiological images. Traditional encoder-decoder methods (Chen et al., 2020; Liu et al., 2021a) rely on word-level CE loss, which limits semantic flexibility and penalizes clinically valid paraphrases. Recent works attempt to mitigate this limitation through domain knowledge integration and architectural innovations. For example, METransformer (Wang et al., 2023a) introduces expert tokens to refine cross-modal attention; EKAGen (Bu et al., 2024) incorporates instance-level expert knowledge to supervise attention; KiUT (Huang et al., 2023) leverages clinical knowledge via U-shaped modality connections for improved alignment. However, these methods still lack fine-grained feedback during training, treating generated reports uniformly and failing to distinguish correct from incorrect semantics. *Our method addresses this by moving beyond rigid token-level supervision. It identifies semantically correct tokens despite surface mismatches, allowing more flexible, meaning-preserving training.*

LLM-Based Report Generation. Large language models (LLMs) such as LLaMA (Touvron et al., 2023), Qwen (Yang et al., 2024), and their RRG adaptations (Wang et al., 2023b; Liu et al., 2024c) improve performance by aligning visual features with LLM token embeddings. For instance, R2GenGPT uses a linear visual mapper for LLM input adaptation, while Bootstrapping utilizes in-domain prompts and coarse-to-fine decoding. Yet, most existing LLM-based models lack mechanisms for token-level feedback or error-aware learning. *In contrast, we introduce a token-level error tagging mechanism that enables precise, supervision-free feedback on which tokens require refinement.*

Token-Level Supervision and Feedback Learning. Token-level feedback has been explored in general NLG via reinforcement learning (Ziegler et al., 2019), contrastive objectives (Liu et al., 2024a), or correction-based post-hoc tuning. However, such approaches often require manually defined reward signals or additional labeled data, limiting scalability. In medical NLP, few RRG methods apply token-level semantic supervision, due to high annotation costs and limited modeling strategies. *To overcome this, we use in-context prompting of a frozen LLM to automatically tag semantically inconsistent tokens (<e>...</e>) in generated reports. This enables scalable error detection without human annotation. Additionally, we introduce a dual-loss framework that penalizes overconfident errors (ETAPL) and reinforces correct predictions (CTAL), aligning training dynamics with clinical accuracy. Intuitively, CTAL functions as a semantic stabilizer, and ETAPL as a semantic corrector.*

3 Methodology

An overview of our SAT-RRG framework is illustrated in Fig. 1, which consists of three key components: (a) report generation (Section 3.1), (b) error token identification (Section 3.2), and (c) inference

Algorithm 1: Training Phase of SAT-RRG

Input: Chest X-ray image X_{img}, Prompt P, Ground truth report R_{GT}
Output: Trainable loss L_{total}

Step 1: Input Construction
Extract visual features: $Z_v \leftarrow f_{\text{img}}(X_{\text{img}})$
Project to embedding space: $H_v \leftarrow \text{MLP}(Z_v)$
Tokenize: $P_{\text{tok}} \leftarrow \text{Tokenizer}(P)$,
$R_{\text{GT_tok}} \leftarrow \text{Tokenizer}(R_{\text{GT}})$
Concatenate: $X \leftarrow \text{Concat}(H_v, P_{\text{tok}}, R_{\text{GT_tok}})$
Build causal attention mask M over $R_{\text{GT_tok}}$

Step 2: Forward Pass
$\hat{y} \leftarrow \text{LLM.forward}(X, \text{mask} = M)$

Step 3: Self-Critique Error Detection
Generate decoded prediction: $y \leftarrow \text{Argmax}(\hat{y})$
Detect semantic errors:
`error_tokens` $\leftarrow$ `LLM.detect`(R_{GT}, y)
$T_{\text{correct}} \leftarrow \{y_j \mid y_j \notin \texttt{error_tokens}\}$
$T_{\text{error}} \leftarrow \{y_k \mid y_k \in \texttt{error_tokens}\}$

Step 4: Loss Computation
$\mathcal{L}_{\text{correct}} \leftarrow -\sum_{j \in T_{\text{correct}}} p(y_j) \log p(y_j)$
$\mathcal{L}_{\text{error}} \leftarrow \sum_{k \in T_{\text{error}}} p(y_k) \log p(y_k)$
$\mathcal{L}_{\text{GT}} \leftarrow -\sum_{i=1}^{T} y_i^{\text{GT}} \log p(\hat{y}_i)$
Total loss: $\mathcal{L}_{\text{total}} \leftarrow \lambda(\mathcal{L}_{\text{correct}} + \mathcal{L}_{\text{error}}) + \mathcal{L}_{\text{GT}}$

Step 5: Backpropagation
Update model using $\nabla_\theta \mathcal{L}_{\text{total}}$

process(right part of Figure 1). The training procedure is detailed in Algorithm 1, and a step-by-step example is provided in Section 3.4.

3.1 Report Generation

Our report generation routine includes three components: (1) a visual encoder to extract features from the input image, (2) a visual mapper to project these features into the LLM's word embedding space, and (3) an LLM that generates reports from both visual and textual prompts.
Given an input chest X-ray image X_v, we employ the Swin Transformer (Liu et al., 2021b) to extract visual features, denoted as $Z_v = \text{Swin}(X_v)$, where Z_v represents the feature map from the last layer, capturing the relevant image details for report generation. To align these features with the LLM's word embedding space, we apply a multi-layer perceptron (MLP) visual mapper, transforming them as $H_v = \text{MLP}(Z_v)$, where H_v is the high-dimensional visual representation compatible with the LLM. Next, the mapped visual features H_v are combined with a textual prompt P and the ground truth report R_{GT}, both of which are tokenized for processing by the LLM. The final input to the LLM is then expressed as:

$$X = \text{Concat}(H_v, \text{Tokenizer}(P), \text{Tokenizer}(R_{\text{GT}})),$$

where P provides the context: "Generate a comprehensive and detailed diagnosis report for this chest X-ray image." The LLM generates a sequence of tokens $y = \{y_1, y_2, \ldots, y_T\}$, predicted under word-level cross-entropy loss.

3.2 Dynamic error token identification and correction

In addition to the report generation routine, we introduce an error correction process that uses the LLM's semantic understanding to detect phrase-level discrepancies between generated and reference reports. As shown in Fig. 1 (b), the error token identification process classifies tokens as correct or erroneous based on semantic consistency. Specifically, by feeding the LLaMA3 both the reference and generated reports, marking meaning-altering discrepancies with <e></e> while leaving correct tokens unmarked. Our prompt provides rules to help the LLM recognize semantic errors and alternative expressions with the same meaning. We also use a few-shot approach, providing examples to achieve the desired results. (Check prompt detail in Section: 5). Some examples will be provided in Fig. 4 in the experimental part.

Fig. 4 shows examples of error tokens. The <e></e> tags are applied only to clear semantic errors in the generated report. For example, "right lower lobe pneumonia" contradicts the ground truth "no evidence of pneumonia", while "no pleural effusion" conflicts with "small right and moderate left pleural effusion", marking both as errors. Semantically consistent expressions, such as "no evidence of pneumonia" vs "there is no focal consolidation concerning for pneumonia", are not tagged as errors since they are semantically consistent despite different wording. This allows us to identify semantic errors rather than lexical differences.

Distinguishing between correct and erroneous tokens enables applying targeted loss functions to refine the training process, as shown in Fig. 2. Correct tokens receive an augment loss (L_{CTAL}), where gradient descent increases their predicted probability, reinforcing confidence in accurate predictions. Conversely, error tokens are penalized using an adaptive correction loss (L_{ETAPL}), where gradient descent reduces their probability in future predictions, discouraging repeated errors. The loss functions are detailed in the next section.

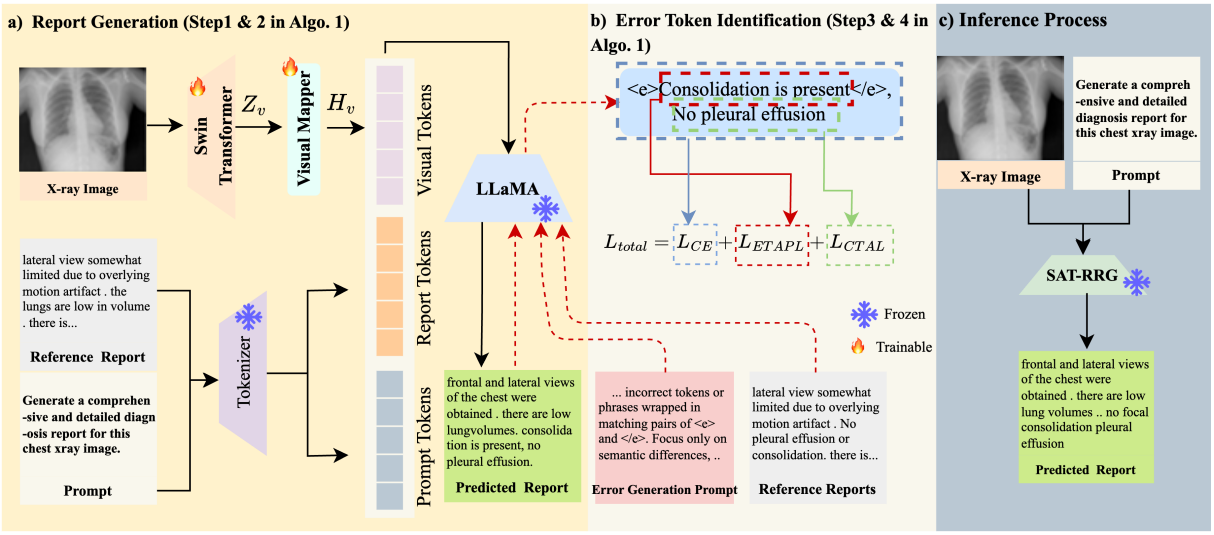

Figure 1: Overview of the SAT-RRG framework. It includes three components: (a) Report generation process, (b) Error token identification via LLM-based semantic comparison, and (c) Inference process. Tokens or modules with the same color represent the same entity across different stages. After the generator produces the initial predicted report, it is passed through the LLM again for feedback. At this stage, each token in the predicted report retains its original generation probability, since the model has not yet been updated by any loss. The LLM-based comparison only annotates which tokens are semantically incorrect. These annotations serve as token-level supervision targets: incorrect tokens are penalized and correct tokens reinforced via two complementary losses (ETAPL and CTAL). The original probabilities are used by the losses to propagate training signals back into the model.

3.3 Error-Aware Training Objectives

While standard cross-entropy loss ensures fluency and grammaticality by aligning predictions with reference tokens, it treats all errors equally, overlooking the distinction between critical clinical mistakes and benign phrasing differences. To address this, we introduce two token-level auxiliary losses that apply differentiated supervision: one reinforces confidently correct tokens, while the other penalizes semantic errors predicted with high confidence, enabling more precise and semantically aware learning.

CTAL: Reinforcing Confidently Correct Tokens. Let T_{correct} denote the set of tokens identified as semantically correct (i.e., not enclosed in <e>...</e>). To reinforce these tokens, we define the Correct Token Augmentation Loss (CTAL) as:

$$\mathcal{L}_{\text{CTAL}} = -\sum_{j \in T_{\text{correct}}} p(y_j) \log p(y_j). \quad (1)$$

The gradient of this loss with respect to the predicted token probability is:

$$\frac{\partial \mathcal{L}_{\text{CTAL}}}{\partial p(y_j)} = -\log p(y_j) - 1. \quad (2)$$

This formulation encourages the model to further boost the probability of correct predictions. Correct tokens with low confidence receive stronger reinforcement, while correct tokens with high confidence are updated more conservatively, thus promoting stable and consistent learning. **ETAPL: Penalizing Overconfident Semantic Errors.** Let T_{error} be the set of tokens enclosed in <e>...</e>, indicating semantic inconsistencies. We define the Error Token Adaptive Penalty Loss (ETAPL) as:

$$\mathcal{L}_{\text{ETAPL}} = \sum_{k \in T_{\text{error}}} p(y_k) \log p(y_k). \quad (3)$$

Its gradient with respect to the model output is:

$$\frac{\partial \mathcal{L}_{\text{ETAPL}}}{\partial p(y_k)} = 1 + \log p(y_k). \quad (4)$$

This loss penalizes tokens that are both semantically incorrect and predicted with high confidence. Larger gradients are applied to high-confident mistakes (with higher $p(y_k)$), reducing their future predicted probabilities by decreasing the corresponding logits during backpropagation. Conversely, low-confidence errors (with lower $p(y_k)$) are penalized lightly, avoiding overcorrection and instability. **Total Loss.** Combining the two auxiliary objectives with the standard cross-entropy loss, the overall training objective is defined as:

$$\mathcal{L}_{\text{total}} = \lambda \cdot (\mathcal{L}_{\text{CTAL}} + \mathcal{L}_{\text{ETAPL}}) + \mathcal{L}_{\text{GT}}. \quad (5)$$

Here, $\mathcal{L}_{\text{GT}}$ is the conventional cross-entropy loss computed over the ground-truth report, and λ is a

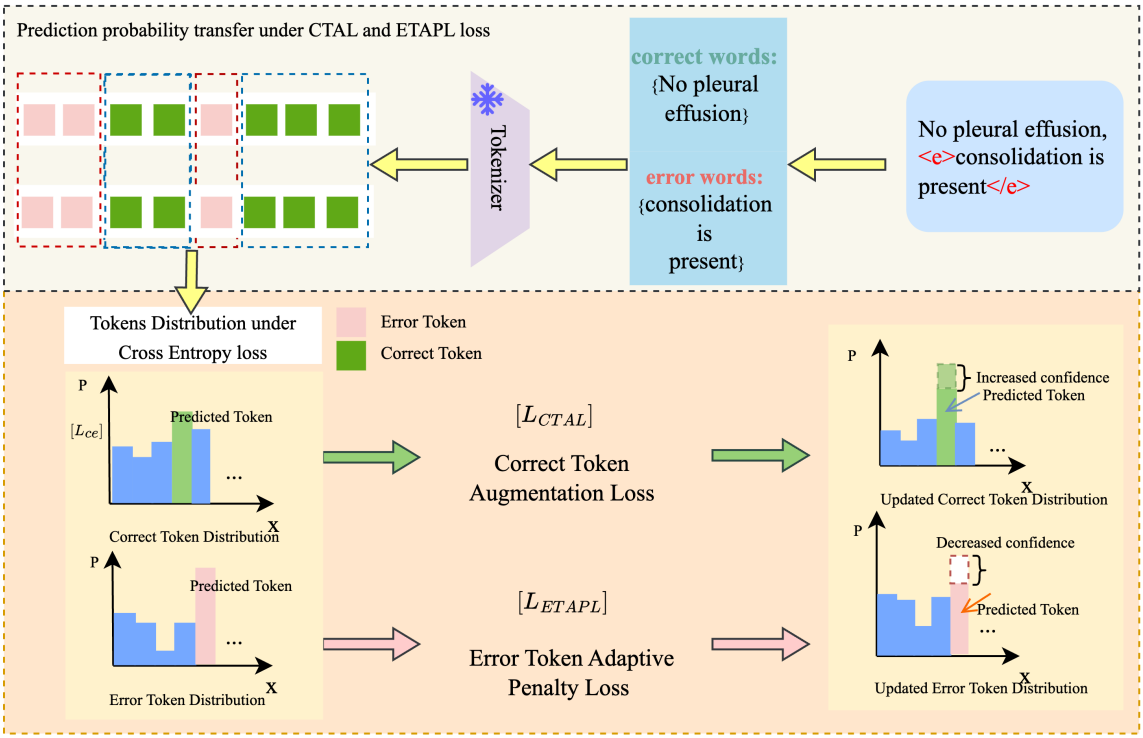

Figure 2: Overview of our proposed training framework with token-level supervision. During training, the predicted report is dynamically compared to the reference report to identify correct and error tokens using the LLM's own semantic understanding. Tokens are categorized as correct or error using inline tags. The **Correct Token Augmentation Loss (CTAL)** increases the confidence of already correct tokens by flattening their gradients, while the **Error Token Adaptive Penalty Loss (ETAPL)** penalizes overconfident incorrect tokens. This dynamic feedback enables the model to refine generation quality at the token level in an interpretable and adaptive manner.

balancing coefficient that controls the influence of the self-supervised token-level feedback. This joint formulation enables the model to reinforce confident correct tokens, penalize overconfident errors, and remain aligned with reference supervision.

The gradient of the total loss with respect to model parameters θ is:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{total}}}{\partial \theta} &= \lambda \sum_{j \in T_{\text{correct}}} \frac{\partial p(y_j)}{\partial \theta} \cdot (-\log p(y_j) - 1) \\ &+ \lambda \sum_{k \in T_{\text{error}}} \frac{\partial p(y_k)}{\partial \theta} \cdot (1 + \log p(y_k)) \\ &- \sum_{i=1}^{T} \frac{\partial p(y_i^{\text{pred}})}{\partial \theta} \cdot \frac{y_i^{\text{GT}}}{p(y_i^{\text{pred}})}. \end{aligned} \tag{6}$$

This gradient offers directionally interpretable updates: it strengthens low-confidence correct predictions, suppresses high-confidence semantic errors, and maintains fidelity to ground-truth references. Together, these components form a coherent learning signal for semantically aware training.

Loss Formulation and Gradient Analysis The gradients of our two auxiliary loss terms provide interpretable and directionally intuitive supervision signals, whichs shows in Figure 3. Specifically, they increase the confidence of semantically correct predictions and penalize overconfident errors, facilitating stable and accurate training.

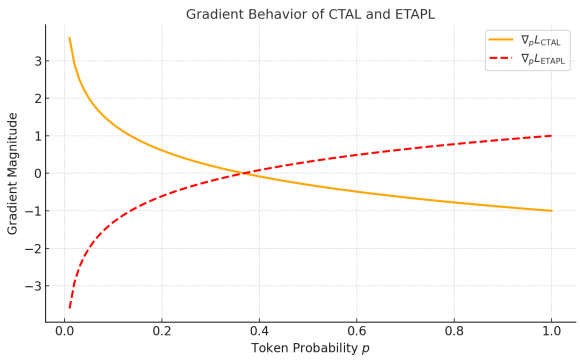

Figure 3: **Gradient dynamics of CTAL and ETAPL.** CTAL applies stronger gradients to low-confidence correct tokens, reinforcing them during training. ETAPL imposes larger penalties on high-confidence incorrect tokens, suppressing overconfident semantic errors. This dual behavior facilitates stable and semantically aware supervision.

Dataset	Methods	@B-1	@B-2	@B-3	@B-4	METEOR	ROUGE
MIMIC-CXR	AdaAtt (Lu et al., 2017)	0.314	0.198	0.132	0.094	0.128	0.131
	M2Transformer (Cornia et al., 2020)	0.332	0.210	0.142	0.101	0.134	0.142
	R2Gen† (Chen et al., 2020)	0.353	0.218	0.145	0.103	0.142	-
	R2GenCMN† (Chen et al., 2022)	0.353	0.218	0.148	0.106	0.142	-
	PPKED† (Liu et al., 2021a)	0.360	0.224	0.149	0.106	0.149	0.237
	METransformer† (Wang et al., 2023a)	0.386	0.250	0.169	0.124	0.152	0.291
	DCL† (Li et al., 2023)	-	-	-	0.109	0.150	0.284
	KiUT† (Huang et al., 2023)	0.393	0.243	0.159	0.113	0.160	0.285
	R2GenGPT† (Wang et al., 2023b)	0.411	0.267	0.186	0.134	0.160	0.297
	EKAGen† (Bu et al., 2024)	0.419	0.258	0.170	0.119	0.157	0.287
	Bootstrapping† (Liu et al., 2024c)	0.402	0.262	0.180	0.128	**0.175**	0.291
	CvT2DistilGPT2† (Nicolson et al., 2023)	0.393	0.248	0.171	0.127	-	0.155
	RaDialog-RG† (Pellegrini et al., 2023)	0.346	-	-	0.095	0.140	0.271
	Multi-Grained† (Liu et al., 2024b)	0.346	0.226	0.159	0.117	0.163	0.290
	PromptMRG† (Jin et al., 2024)	0.398	-	-	0.112	0.157	0.268
	Ours	**0.431**	**0.284**	**0.199**	**0.146**	0.168	**0.306**
IU-Xray	AdaAtt (Lu et al., 2017)	0.284	0.207	0.150	0.126	0.165	0.311
	M2Transformer (Cornia et al., 2020)	0.402	0.284	0.168	0.143	0.170	0.328
	R2Gen† (Chen et al., 2020)	0.470	0.304	0.219	0.165	0.187	0.371
	R2GenCMN† (Chen et al., 2022)	0.475	0.309	0.222	0.170	0.191	0.375
	METransformer† (Wang et al., 2023a)	0.483	0.322	0.228	0.172	0.192	0.380
	DCL† (Li et al., 2023)	-	-	-	0.163	0.193	-
	R2GenGPT† (Wang et al., 2023b)	0.488	0.316	0.228	0.173	0.211	0.377
	Bootstrapping† (Liu et al., 2024c)	0.499	0.323	0.238	0.184	0.208	0.390
	CvT2DistilGPT2† (Nicolson et al., 2023)	0.473	0.304	0.224	0.175	0.200	0.376
	Multi-Grained† (Liu et al., 2024b)	0.472	0.321	0.234	0.175	0.192	0.379
	PromptMRG† (Jin et al., 2024)	0.401	-	-	0.098	0.160	0.281
	EKAGen (Bu et al., 2024)	0.497	0.339	0.250	0.190	0.210	0.399
	Ours	**0.504**	**0.342**	**0.254**	**0.196**	**0.222**	**0.400**

Table 1: Comparison on MIMIC-CXR and IU-Xray datasets(@B stands for BLUE).

3.4 Example Workflow Illustration

To clarify how our self-adaptive token-level refinement framework operates, we illustrate the end-to-end process with a concrete example. The framework consists of four major steps: report generation, semantic error annotation, loss-based refinement, and token probability update.

Initial Report Generation (Step 1 & 2 in Algo. 1). Given an input chest X-ray image and a textual prompt, the frozen LLM generates a draft report:

> *Prediction:* Consolidation is present, no pleural effusion.

Each token is associated with a model-predicted confidence (probability). For example:

Token	Probability
`Consolidation`	0.85
`is`	0.90
`present`	0.92
`no`	0.98
`pleural`	0.96
`effusion`	0.93

Self-Critiqued Error Annotation (Step 3 in Algo. 1). The same LLM is used to compare the generated report against the reference report:

> *Reference:* No pleural effusion or consolidation.

Using few-shot prompting and semantic reasoning, the model identifies that the phrase "consolidation is present" contradicts the ground truth, and thus annotates it with error tags:

> *Annotated:*`<e>Consolidation is present</e>`, no pleural effusion.

Token-Level Supervision via CTAL and ETAPL ((Step 4 in Algo. 1). The identified correct tokens (`No`, `pleural`, `effusion`) are reinforced using the CTAL, which increases their predicted probability. In contrast, the erroneous tokens (`consolidation`, `is`, `present`) are penalized using the ETAPL, which pushes their probabilities downward, especially if the initial confidence was high.

Parameter Update (Step 5 in Algo. 1). After applying the adaptive loss, the token probabilities are adjusted via backpropagation. The resulting updates are as follows:

Token	Initial	Type	Updated
`Consolidation`	0.85	Penalize	0.60
`is`	0.90	Penalize	0.65
`present`	0.92	Penalize	0.58
`no`	0.98	Reinforce	0.99
`pleural`	0.96	Reinforce	0.98
`effusion`	0.93	Reinforce	0.97

This token-level correction mechanism enables the model to learn not just from the reference report but from its own semantic inconsistencies, driving more precise and fluent report generation over time.

4 Experiments

Datasets Our experiment was conducted on MIMIC-CXR and IU-Xray, the two most commonly used datasets for RRG tasks.

MIMIC-CXR (Johnson et al., 2019) is a large dataset with 377,110 chest radiographs and 227,835 reports from 64,588 patients, collected at Beth Israel Deaconess Medical Center (2011-2016). We follow the partitioning protocol from Chen et al. (Chen et al., 2020), using 270,790 images for training and 3,858 for testing for fair evaluation.

IU-Xray (Demner-Fushman et al., 2016) is a widely used dataset for radiology report generation, containing 3,955 reports and 7,470 chest X-ray images. We follow the data split protocol from Chen et al. (Chen et al., 2020), partitioning the dataset into training, validation, and test sets with a 7:1:2 ratio, and evaluating the test set for consistency.

Implementation Details We leverage LLaMA3-3B[1] as the large language model (LLM) and Swin Transformer[2] as the visual encoder. The loss balancing coefficient λ is set to 0.5. Training is conducted on two NVIDIA A6000 GPUs (48GB each) with a mini-batch size of 24 and a learning rate of 5e-5. For inference, we apply beam search with a beam width of 3 to balance computational efficiency and output quality.

Evaluation Metrics We assess the generated diagnostic reports using standard NLG metrics, including BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and ROUGE-L (Lin, 2004), following the standard evaluation protocol[3]. For clinical relevance evaluation, CheXBert (Irvin et al., 2019) is excluded due to past inconsistencies in the calculation (e.g., micro/macro/weighted variants) and lack of transparency in prior works, hindering fair comparisons. Instead, we adopt more robust metrics—RadGraph F1 (Jain et al., 2021), BERTScore (Zhang et al., 2019), and RadCliQ (Yu et al., 2023)—that better align with experts' assessments. Notably, RadCliQ integrates CheXBert in a standardized manner.

Main Results We compare our model with state-of-the-art (SOTA) methods on the MIMIC-CXR and IU-Xray datasets, as shown in Table 1, respectively. Table 2 presents comparisons of clinical metrics. Methods marked with † are quoted from their original papers, while others were re-run using publicly released code on the same training-test partition as our method.

As shown inTable 1, our model outperforms these SOTA methods across almost all metrics on both datasets, including image captioning models (Lu et al., 2017; Cornia et al., 2020), transformer-based encoder-decoder (Chen et al., 2020, 2022; Liu et al., 2021a; Wang et al., 2022b; Huang et al., 2023; Bu et al., 2024), and LLM-based methods (Wang et al., 2023b; Liu et al., 2024c; Pellegrini et al., 2023). Specifically, our model surpasses the LLM-based models R2GenGPT (Wang et al., 2023b) and Bootstrapping (Liu et al., 2024c) in BLEU score, with improvements of 7.5% and 12.5%, despite these two methods using larger LLMs (7B) compared to ours (3B). Additionally, methods leveraging disease knowledge (Bu et al., 2024; Huang et al., 2023) lack feedback mechanisms for guided learning, while Multi-Grained (Liu et al., 2024b) employs sentence-level contrastive learning but lacks token-level supervision, limiting its effectiveness. EKAGen, which uses 300×300 images instead of the standard 224×224, is not directly comparable but still underperforms our approach. On IU-Xray, our model consistently demonstrates strong performance. Beyond NLG metrics, overall, it achieves better performance on clinical relevance metrics—RadGraph F1, BERTScore, and RadCliQ. This improvement stems from our self-generated token-level supervision, which dynamically refines predictions by reinforcing correct tokens and suppressing incorrect ones.

Ablation Study Table 3 shows the ablation study results for the model's loss components. Both the CTAL loss (L_{correct}) and ETAPL loss (L_{error}) individually improve performance over the baseline, which excludes both losses. The best results occur

[1] https://huggingface.co/meta-llama/Llama-3.2-3B

[2] https://huggingface.co/microsoft/swin-base-patch4-window7-224

[3] https://github.com/tylin/coco-caption

Methods	RadGraph F1 (↑)	Bert Score (↑)	RadCliQ (↓)
R2Gen (Chen et al., 2020)	0.172	0.406	1.228
R2GenCMN (Chen et al., 2022)	0.182	0.418	1.182
CvT2DistilGPT2 (Nicolson et al., 2023)	0.196	0.374	1.220
RaDialog-RG† (Pellegrini et al., 2023)	-	0.400	-
R2GenGPT (Wang et al., 2023b)	0.187	0.415	1.207
PromptMRG (Jin et al., 2024)	0.190	0.357	1.169
EKAGen†† (Bu et al., 2024)	0.199	0.412	1.126
Ours	**0.205**	**0.422**	**1.150**

Note: EKAGen uses 300×300 images while the others use 224×224 images. We exclude CheXBert due to inconsistencies in prior evaluations and lack of transparency, which hinder fair comparison. Notably, RadCliQ integrates CheXBert in a standardized manner.

Table 2: Evaluation of Clinic-related Metrics on MIMIC-CXR.

Dataset	$L_{\text{error}}^{\text{penalty}}$	L_{correct}	@B-1	@B-2	@B-3	@B-4	METEOR	ROUGE_L
Mimic-cxr			0.423	0.273	0.188	0.135	0.166	0.295
	✓		0.426	0.277	0.192	0.139	0.166	0.298
		✓	0.428	0.281	0.196	0.141	0.167	0.303
	✓	✓	0.431	0.284	0.199	0.146	0.168	0.306

Table 3: Ablation study for loss components.

Ground Truth Report	Report Before TLS	Report After TLS
impression : no evidence of pneumonia or decompensated congestive heart failure . stable findings associated with the patients known lung malignancy . findings : pa and lateral radiograph of the chest once again demonstrate a right upper lobe mass with a fiducial marker in place as well as a right perihilar mass . this is consistent with the patients known malignancy . once again seen are a small right and moderate left pleural effusion with considerable left lower lobe atelectasis or consolidation	impression : <e>right lower lobe pneumonia</e> . findings : there is a new opacity in the <e>right lower lobe concerning for pneumonia</e> . there is <e>no pleural effusion</e> or pneumothorax . cardiomediastinal silhouette and hilar contours are unremarkable . <e>left pectoral pacemaker with leads in the right atrium and right ventricle</e> is unchanged . <e>median sternotomy wires</e> are intact .	impression : <e>small left pleural effusion</e> . findings : pa and lateral chest views were obtained with patient in upright position . analysis is performed in direct comparison with the next preceding similar study of . heart size is normal . cardiomediastinal contours are normal . there is <e>a small left pleural effusion</e> . there is no pneumothorax . there is no focal consolidation concerning for pneumonia . there is no pulmonary edema . bony structures are unremarkable .

Figure 4: Comparison of error tokens in the generated report before and after TLS. The highlighted errors are marked in corresponding colors to show the corrections made.

when both losses are combined, highlighting their complementary effects in enhancing the quality of the generated reports.

4.1 Qualitative Analysis

Figure 4 shows the changes with and without token-level supervision (TLS), demonstrating how the errors get corrected. Before applying TLS, the model makes errors, such as incorrectly identifying conditions or adding irrelevant details. For example, it mentions "right lower lobe concerning for pneumonia" when the ground truth states "no evidence of pneumonia." With TLS, most errors are corrected, as shown in the "Report After TLS" where the model removes incorrect terms like "right lower lobe pneumonia". The colors highlight corresponding concepts across reports.

5 Conclusion

We proposed **SAT-RRG**, a novel self-adaptive training framework for radiology report generation that incorporates token-level semantic supervision. By leveraging LLM-based feedback to identify and localize semantic inconsistencies, our approach enables fine-grained correction and reinforcement during training. We introduce two adaptive loss functions—CTAL and ETAPL—that selectively guide the model to consolidate correct predictions and suppress overconfident errors. Experimental results on MIMIC-CXR and IU-Xray demonstrate that SAT-RRG achieves SOTA performance over conventional CE-based training paradigms.

Limitations

In this work, we employ a relatively lightweight general-purpose LLM (LLaMA3-3B) for both report generation and semantic error detection, as a proof-of-concept. While our approach already demonstrates visible performance gains, we envision that using more powerful LLMs could further enhance semantic understanding and error correction capacity.

Despite these benefits, relying on a general-purpose LLM introduces dependency on its contextual understanding, which may occasionally misidentify errors in clinically ambiguous or underspecified cases. Although such instances are rare, future work could explore domain-adapted LLMs, improved prompt designs, or lightweight error detection modules jointly trained with the generation model to further boost robustness.

Importantly, the effectiveness of our proposed training framework does not hinge on perfect error identification. The current feedback mechanism already leads to significant improvements over conventional supervision. Further improvements in error tagging precision would only enhance our model's performance, making the framework even more reliable and clinically useful.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Shenshen Bu, Taiji Li, Yuedong Yang, and Zhiming Dai. 2024. Instance-level expert knowledge and aggregate discriminative attention for radiology report generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14204.

Zhihong Chen, Yaling Shen, Yan Song, and Xiang Wan. 2022. Cross-modal memory networks for radiology report generation. *arXiv preprint arXiv:2204.13258*.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. 2020. Generating radiology reports via memory-driven transformer. *arXiv preprint arXiv:2010.16056*.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2020. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10578–10587.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. 2016. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Zhongzhen Huang, Xiaofan Zhang, and Shaoting Zhang. 2023. Kiut: Knowledge-injected u-transformer for radiology report generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19809–19818.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, and 1 others. 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597.

Saahil Jain, Ashwin Agrawal, Adriel Saporta, Steven QH Truong, Du Nguyen Duong, Tan Bui, Pierre Chambon, Yuhao Zhang, Matthew P Lungren, Andrew Y Ng, and 1 others. 2021. Radgraph: Extracting clinical entities and relations from radiology reports. *arXiv preprint arXiv:2106.14463*.

Haibo Jin, Haoxuan Che, Yi Lin, and Hao Chen. 2024. Promptmrg: Diagnosis-driven prompts for medical report generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2607–2615.

Alistair EW Johnson, Tom J Pollard, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G Mark, Seth J Berkowitz, and Steven Horng. 2019. Mimic-cxr-jpg, a large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*.

Mingjie Li, Bingqian Lin, Zicong Chen, Haokun Lin, Xiaodan Liang, and Xiaojun Chang. 2023. Dynamic graph enhanced contrastive learning for chest x-ray report generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3334–3343.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Alex Liu, Yixuan Guo, Jianhui Yong, and Feng Xu. 2024a. Multi-grained radiology report generation with sentence-level image-language contrastive learning. *IEEE Transactions on Medical Imaging*.

Aohan Liu, Yuchen Guo, Jun-hai Yong, and Feng Xu. 2024b. Multi-grained radiology report generation with sentence-level image-language contrastive learning. *IEEE Transactions on Medical Imaging*.

Chang Liu, Yuanhe Tian, Weidong Chen, Yan Song, and Yongdong Zhang. 2024c. Bootstrapping large language models for radiology report generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18635–18643.

Fenglin Liu, Xian Wu, Shen Ge, Wei Fan, and Yuexian Zou. 2021a. Exploring and distilling posterior and prior knowledge for radiology report generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13753–13762.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021b. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 375–383.

Aaron Nicolson, Jason Dowling, and Bevan Koopman. 2023. Improving chest x-ray report generation by leveraging warm starting. *Artificial intelligence in medicine*, 144:102633.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Chantal Pellegrini, Ege Özsoy, Benjamin Busam, Nassir Navab, and Matthias Keicher. 2023. Radialog: A large vision-language model for radiology report generation and conversational assistance. *arXiv preprint arXiv:2311.18681*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Zhanyu Wang, Hongwei Han, Lei Wang, Xiu Li, and Luping Zhou. 2022a. Automated radiographic report generation purely on transformer: A multicriteria supervised approach. *IEEE Transactions on Medical Imaging*, 41(10):2803–2813.

Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. 2023a. Metransformer: Radiology report generation by transformer with multiple learnable expert tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11558–11567.

Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. 2023b. R2gengpt: Radiology report generation with frozen llms. *Meta-Radiology*, 1(3):100033.

Zhanyu Wang, Mingkang Tang, Lei Wang, Xiu Li, and Luping Zhou. 2022b. A medical semantic-assisted transformer for radiographic report generation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 655–664. Springer.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2 technical report. *CoRR*.

Feiyang Yu, Mark Endo, Rayan Krishnan, Ian Pan, Andy Tsai, Eduardo Pontes Reis, Eduardo Kaiser Ururahy Nunes Fonseca, Henrique Min Ho Lee, Zahra Shakeri Hossein Abad, Andrew Y Ng, and 1 others. 2023. Evaluating progress in automatic chest x-ray radiology report generation. *Patterns*, 4(9).

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

Appendix A: Prompt for Error Token Annotation

Instruction. Your task is to compare a *predicted report* with a *ground truth* medical report and identify specific tokens or phrases in the predicted report that are semantically incorrect or conflict with the ground truth. For this task:

1. **Incorrect tokens or phrases** are defined as parts of the predicted report that have a different meaning or contradict the ground truth.
2. Use a **matching pair of `<e>` and `</e>`** to wrap only the incorrect tokens or phrases in the predicted report.
3. Ensure that every error is marked precisely. Do not mark entire sentences—only the specific parts that are incorrect.
4. Preserve the structure of the predicted report. Do not split paragraphs or reformat the text.

Examples.

- **Example 1:**
  - Ground truth: *The lungs are clear and hyperinflated.*
  - Predicted report: *The lungs are clear and hyperinflation is present.*
  - Analysis: The predicted report matches the ground truth report.
  - Output: `The lungs are clear and hyperinflation is present.`
- **Example 2:**
  - Ground truth: *Findings: The lungs are low in volume. No focal airspace consolidation to suggest pneumonia.*
  - Predicted report: *Findings: The lungs are within normal volume. Focal consolidation is noted in the right lower lobe, concerning for pneumonia.*
  - Analysis: The predicted report incorrectly states *within normal volume* and adds *Focal consolidation*, which conflicts with the ground truth.
  - Output: `Findings: The lungs are <e>within normal volume</e>. Focal consolidation <e>is noted</e> in the right lower lobe, concerning for pneumonia.`
- **Example 3:**
  - Ground truth: *Impression: There is evidence of acute cardiopulmonary process.*
  - Predicted report: *Impression: No acute cardiopulmonary process.*
  - Analysis: The predicted report incorrectly negates the cardiopulmonary process described in the ground truth.
  - Output: `Impression: <e>No acute</e> cardiopulmonary process.`
- **Example 4:**
  - Ground truth: *The patient has a 12-cm calcified granuloma unchanged from the prior study.*
  - Predicted report: *The patient has a mass in the lower lung field.*
  - Analysis: The predicted report incorrectly describes a *mass in the lower lung field*, which conflicts with the *12-cm calcified granuloma* in the ground truth.
  - Output: `The patient has a <e>mass in the lower lung field</e>.`

Analyze the following reports and return the predicted report with incorrect tokens or phrases wrapped in matching pairs of `<e>` and `</e>`. Focus only on semantic differences, and ensure no extra modifications are made to the predicted report.