Reinforcement Learning with Imbalanced Dataset for Data-to-Text Medical Report Generation

Toru Nishino[1] **Ryota Ozaki**[1] **Yohei Momoki**[2] **Tomoki Taniguchi**[1]
Ryuji Kano[1] **Norihisa Nakano**[1] **Yuki Tagawa**[1] **Motoki Taniguchi**[1]
Tomoko Ohkuma[1] **Keigo Nakamura**[2]
[1]Fuji Xerox Co., Ltd. [2]Fujifilm Corporation

nishino.toru@fujixerox.co.jp, toru.nishino@fujifilm.com

Abstract

Automated generation of medical reports that describe the findings in the medical images helps radiologists by alleviating their workload. Medical report generation system should generate correct and concise reports. However, data imbalance makes it difficult to train models accurately. Medical datasets are commonly imbalanced in their finding labels because incidence rates differ among diseases; moreover, the ratios of abnormalities to normalities are significantly imbalanced. We propose a novel reinforcement learning method with a reconstructor to improve the clinical correctness of generated reports to train the data-to-text module with a highly imbalanced dataset. Moreover, we introduce a novel data augmentation strategy for reinforcement learning to additionally train the model on infrequent findings. From the perspective of a practical use, we employ a Two-Stage Medical Report Generator (TS-MRGen) for controllable report generation from input images. TS-MRGen consists of two separated stages: an image diagnosis module and a data-to-text module. Radiologists can modify the image diagnosis module results to control the reports that the data-to-text module generates. We conduct an experiment with two medical datasets to assess the data-to-text module and the entire two-stage model. Results demonstrate that the reports generated by our model describe the findings in the input image more correctly.

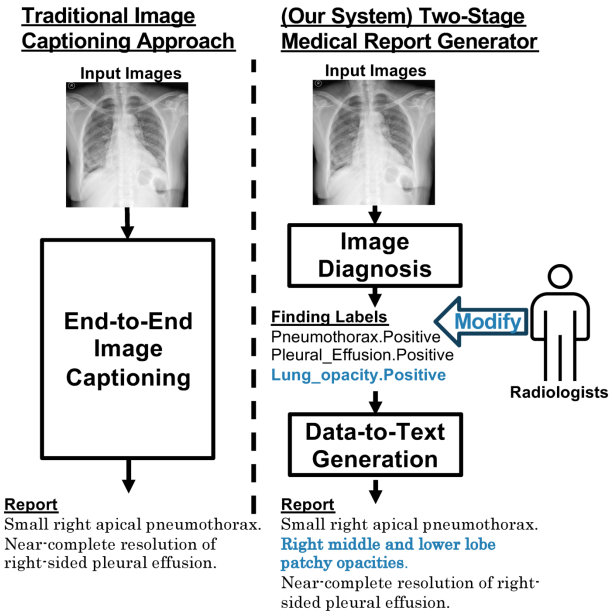

Figure 1: Overview of our Two-Stage Medical Report Generator (TS-MRGen) system.

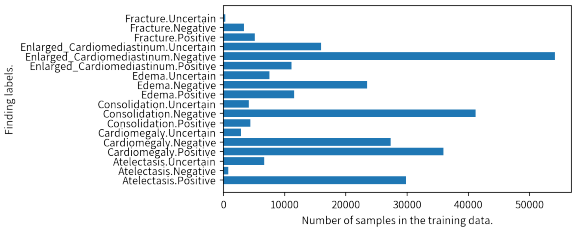

Figure 2: Imbalanced distribution of the part of the finding labels in the MIMIC-CXR dataset.

1 Introduction

Writing medical reports manually from medical images is a time-consuming task for radiologists. To write reports, radiologists first recognize what findings are included in medical images, such as computed tomography (CT) and X-ray images. Then radiologists compose reports that describe the recognized findings correctly without omission. Doctors prefer radiology reports written in natural language. Other types of radiology reports, such as tabular reports, are difficult to understand because of their complexity.

The purpose of our work is to build an automated medical report generation system to reduce the workload of radiologists. As shown in Figure 1, the medical report generation system should generate correct and concise reports for the input images. However, data imbalance may reduce the quality of automatically generated reports. Medical datasets are commonly imbalanced in their finding labels because incidence rates differ among

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 2223–2236

diseases; moreover, the ratios of abnormalities to normalities are also significantly imbalanced. Figure 2 shows an imbalanced distribution of finding labels in the MIMIC-CXR dataset (Johnson et al., 2019). For example, the finding label "Enlarged_Cardiomediastinum.Negative" appears approximately 70 times more frequently than the finding label "Atelectasis.Negative". As a result of that imbalance, the generation model tends to train only the frequent finding labels, and tends to omit descriptions of the infrequent labels. This tendency increases incorrectness of generated reports.

To improve the correctness of generated reports, we propose a novel reinforcement learning (RL) strategy for a data-to-text generation module with a reconstructor. We introduce a new reward, Clinical Reconstruction Score (CRS), to quantify how much information the generated reports retain about the input findings. The reconstructor calculates CRS and uses it as a reward for RL to train the model to generate a greater number of correct reports. Additionally, we introduce a new Reinforcement Learning with Data Augmentation method (RL-DA) to alleviate data imbalance problems that arise from infrequent findings.

To replace the entire workflow of radiologists, end-to-end image captioning approach is primarily considered (Monshi et al., 2020). They generate reports solely from input medical images. However, such approaches are difficult to apply to the real medical field for the following two reasons. First, the quality of generated reports is adversely affected by the insufficient accuracy of image diagnosis systems. To generate correct reports, radiologists must be able to correct wrong image diagnosis results. Second, end-to-end models cannot reflect the intentions of radiologists to reports. In contrast to abnormalities, normalities are less important but frequently appear in the images. Radiologists sometimes deliberately omit the descriptions of some normalities to write concise reports, especially at return visits. To generate concise reports, radiologists should be able to select which findings the system should include in the reports.

We employed the Two-Stage Medical Report Generator (TS-MRGen), a novel framework for controllable report generation. Figure 1 presents an overview of TS-MRGen. TS-MRGen consists of two separate stages: an image diagnosis module and a data-to-text generation module. The image diagnosis module recognizes the findings in the image. Subsequently, reports are generated by the data-to-text module. Radiologists can modify the wrong or unintended results of the image diagnosis module. Next, the modified findings are used as the input to the data-to-text module. This approach greatly improves the correctness and conciseness of generated reports.

Overall, the main contributions of this study are as follows:

- We introduce a reinforcement learning strategy with Clinical Reconstruction Score (CRS) to generate more clinically correct reports.
- We propose a novel Reinforcement Learning with Data Augmentation (RL-DA) to address data imbalance difficulties.
- We design and conduct experiments to validate the effectiveness of Two-Stage Medical Report Generator (TS-MRGen) with a modification process.

We evaluate the proposed approach on two datasets: the Japanese Computed Tomography (JCT) dataset and the MIMIC-CXR dataset. Automatic and manual evaluations on the JCT dataset show that our CRS and RL-DA improve the correctness of generated reports. An experiment conducted on the MIMIC-CXR dataset shows the generality of CRS and RL-DA; moreover, the experiment on the MIMIC-CXR dataset demonstrates that TS-MRGen with the modification process generates more correct reports than the two-stage model without a modification process.

2 Related Work

Medical Report Generation. Many end-to-end medical report generation models have been proposed (Monshi et al., 2020) to generate reports from images. Jing et al. (2018) introduced a co-attention mechanism to align semantic tags and sub-regions of images. However, this model tends to generate sentences describing normalities to an excessively degree. This tendency results from an imbalanced frequency of findings among the medical images. Jing et al. (2019) and Harzig et al. (2019) use different decoders to generate normalities or abnormalities to address these data imbalance difficulties.

Biswal et al. (2020) accepts doctors' anchor words for controllable medical report generation.

This model generates reports that are more faithful to doctors' preferences by retrieving template sentences from the word entered by the doctor.

Data-to-Text Generation. Data-to-text generation is a task to generate a fluent text that is faithful to the input data. Wiseman et al. (2017) proposed a data-to-text model with reconstruction-based techniques. The method trains the model so that the input data can be reconstructed from the decoder hidden state. This reconstruction makes it more likely that the hidden state of the decoder can capture the input data properly.

Ma et al. (2019); Moryossef et al. (2019) proposed a two-step data-to-text model, comprising a text planning module and a text realization module. This model not only generates a text that is more faithful to the input data than end-to-end models, but it also allows for user control over the generated text by supplying a modified plan to the text realization module.

Data Augmentation for Text Generation. Typical machine learning approaches that address data imbalance, such as undersampling and oversampling, are difficult to apply to this task because the input images or finding labels are sets of multiple finding class label and the target reports are discrete sequences. Kedzie and McKeown (2019) applied data augmentation method to data-to-text generation. To obtain additional training data, they generated data-text pairs by the model itself using the noise injection sampling method.

Text Generation with Reinforcement Learning (RL). Text generation with Reinforcement Learning (RL) enables the model to train with indifferentiable rewards, such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) metrics. Zhang et al. (2020b) improved the radiology report summarization model with RL using a factual correctness reward. Liu et al. (2019a) applied RL for medical report generation with Clinically Coherent Reward (CCR) to directly optimize the model for clinical efficacy. Both methods leverage CheXpert Labeler (Irvin et al., 2019), a medical observation annotator, to calculate rewards.

Our work addresses the data imbalance difficulties beyond the imbalance between normalities and abnormalities, as Jing et al. (2019) addressed. Moreover, with our approach, the doctors can reflect their intentions to reports more directly to a greater degree than Biswal et al. (2020). We extend the factual-based RL method (Liu et al., 2019a) to cases for which rule-based annotators are not available. Furthermore, we propose data augmentation (Kedzie and McKeown, 2019) for RL to train the model using only the input labels.

3 Method

Medical report generation is a task to generate reports consisting of a sequence of words $Y = \{y_1, y_2, ...y_N\}$ from a set of images $X = \{x_k\}_{k=1}^{M}$. Most cases Y include more than one sentence. We annotated a set of finding labels $F = \{f_1, f_2, ...f_T\}$ for each set of images. The finding labels include abnormalities (indicated as .Positive), normalities (indicated as .Negative) and uncertain findings (indicated as .Uncertain) . Each finding label can be disassembled into a sequence of words as $f_t = \{w_{t1}, w_{t2}, ...w_{tK}\}$. For example, an abnormality "Airspace_Opacity.Positive" label is divided into a sequence of {airspace, opacity, positive}.

3.1 Two-Stage Medical Report Generator

We employ Two-Stage Medical Report Generator (TS-MRGen), a framework that consists of two separate stages: an image diagnosis module and a data-to-text generation module. The image diagnosis module can be regarded as an image classification task that recognizes input images X and classifies them into a set of findings F. Radiologists can modify the image diagnosis module result F if errors are found in F. Alternatively, they can intentionally omit or append findings labels. The data-to-text generation module generates a report Y from F. We consider the text generation module as a data-to-text task.

3.2 Image Diagnosis Module

We train an image classification model that takes as input a single-view chest X-ray and output a set of probabilities of four types of labels (positive, negative, uncertain, and no mention) for each possible finding label. We use EfficientNet-B4 (Tan and Le, 2019) as a network architecture that was initialized with the pretrained model on ImageNet (Deng et al., 2009).

In some cases, the reports are described based on two images: front view and lateral view. Following Irvin et al. (2019), this module outputs the mean probability of the model between two images.

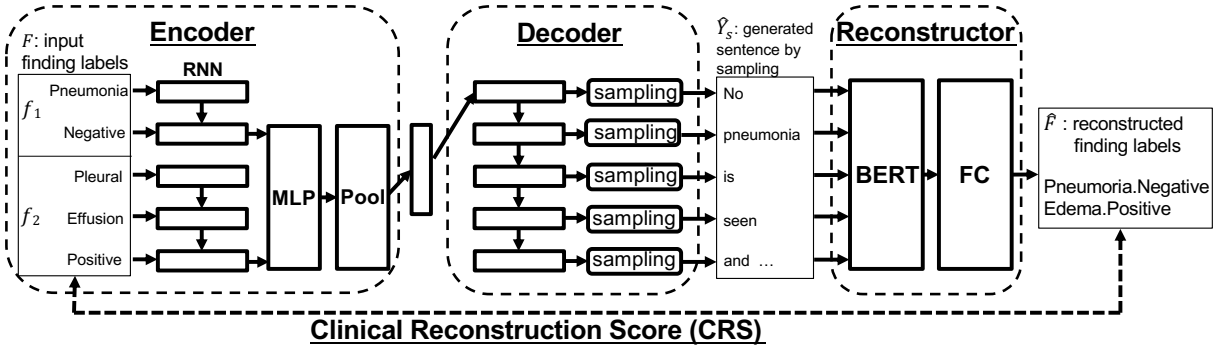

Figure 3: Overview of our reinforcement learning (RL) with a reconstructor. We leverage the clinical Reconstruction Score (CRS), which estimates the factual correctness of generated reports, as a reward for RL.

3.3 Text Generation Module

We adopt a table-to-text encoder-decoder model (Liu et al., 2018) for the text generation module to use words in the findings class labels. The encoder of the text generation module has two layers: a word-level encoder and a label-level layer.

$$h^w_{tk} = \mathrm{Enc}_{word}(w_{tk}, h^w_{tk-1}) \quad (1)$$

$$h^l_t = \mathrm{MLP}_{label}([h^w_{t0}, h^w_{tK}]) \quad (2)$$

Therein, $[h^w_{t0}, h^w_{tK}]$ denotes the concatenation of vectors h^w_{t0} and h^w_{tK}. MLP_{label} represents a multi-layer perceptron. We use a one-layer bi-directional gated recurrent unit (GRU) for the word level encoder.

For the decoder, we use one-layer GRU with an attention mechanism (Bahdanau et al., 2015):

$$y_n = \mathrm{Dec}(y_{n-1}, h^l, h^d_{n-1}, c_n) \quad (3)$$

where h^l represents the max-pooled vector from $\{h^l_0, ..., h^l_T\}$. The context vector c_n is calculated over the label-level hidden vectors h^l_t and the decoder hidden state h^d_n.

3.4 RL with Reconstructor

We use RL to train the text generation model to improve the clinical correctness of the generated reports. A benefit of RL is that the model can be trained to produce sentences that maximize the reward, even if the word sequence does not match the correct answer. Many studies of text generation with RL (Keneshloo et al., 2019) use rewards, such as the BLEU and ROUGE metrics, to improve the generated text. To improve the clinical correctness of the generated reports, Liu et al. (2019a) and Irvin et al. (2019) adopted clinically coherent rewards for RL with CheXpert Labeler (Irvin et al., 2019), a rule-based finding mention annotator. However, in the medical domain, no such annotator is available in most cases other than English chest X-ray reports.

We propose a new reward, Clinical Reconstruction Score (CRS), to quantify the factual correctness of reports with a reconstructor module. Figure 3 shows an overview of our method, RL with CRS. Contrary to the data-to-text generator, the reconstructor reversely predicts the appropriate finding labels from the generated reports. This reconstructor quantifies the clinical correctness of the reports. Therefore, we can estimate the correctness of reports without rule-based annotators.

We utilize BERT (Devlin et al., 2019) as a reconstructor and reconstructed the finding labels $\hat{F}$ as a multi-label text classification task:

$$\hat{F} = \mathrm{FC}(\mathrm{BERT}(\hat{Y})) \quad (4)$$

where FC and BERT represent the fully connected layer and the BERT layer, respectively. $\hat{Y}$ denotes a generated report. In addition, CRS is defined as an F-score of the predicted finding labels $\hat{F}$ against the input finding labels for the data-to-text module F. This BERT reconstructor is trained with a Class-Balanced Loss (Cui et al., 2019) to address imbalanced datasets.

We design the overall reward as a combination of ROUGE-L score and CRS:

$$R(Y) = \lambda_{rouge}\mathrm{ROUGE}(Y_t, Y) + (1 - \lambda_{rouge})\mathrm{CRS}(Y_t) \quad (5)$$

where Y_t represents a gold report regarding the predicted report Y and λ_{rouge} is a hyperparameter.

The goal of RL is to find parameters to minimize the negative expected reward $R(\hat{Y})$ for $\hat{Y}$:

$$L^{rl}_\theta = -\mathbb{E}_{\hat{Y} \sim P_\theta} R(\hat{Y}) \quad (6)$$

where P_θ denotes a policy network for the text generation model.

We adopt SCST (Rennie et al., 2017) to approximate the gradient of this loss:

$$\nabla_\theta L_\theta^{rl} \approx -\nabla_\theta \log P_\theta(\hat{Y}^s)(R(\hat{Y}^s) - R(\hat{Y}^g)) \quad (7)$$

where $\hat{Y}^s$ is a sampled sequence with a Monte Carlo sampling. We use the softmax function with temperature τ for sampling sequences. $R(\hat{Y}^g)$ is a baseline reward calculated from a greedily decoded sequence $\hat{Y}^g$.

To train the language model, RL with only CRS and ROUGE as a reward is insufficient. Therefore, we use the cross-entropy loss to generate fluent sentences. We design an overall loss function for training as a combination of the RL loss and cross-entropy loss L^{xent}:

$$L^{all} = \lambda_{rl} L^{rl} + (1 - \lambda_{rl}) L^{xent} \quad (8)$$

where L^{xent} is the cross-entropy loss calculated between the gold reports and generated reports, and λ_{rl} is a hyperparameter.

3.5 Reinforcement Learning with Data Augmentation (RL-DA)

We propose a novel method, RL with Data Augmentation method (RL-DA), to encourage the model to focus on infrequent findings. We focus on the asymmetricity between the augmentation cost of the input data and that of the target report sentences. The input data, which comprise a set of finding labels, can be augmented easily by adding or removing a finding label automatically. However, the augmentation cost is higher for the target reports than the input data because the target reports are written in natural language. Therefore, we introduce a semi-supervised reinforcement learning method to train the model solely by augmenting the input data.

We conduct a data augmentation process of RL-DA as the following steps.

Step 1: List and Filter all Candidate Finding Labels. Given a set of finding labels $F = \{f_1, f_2, ...f_T\}$, the objective of the data augmentation is to obtain a new set of finding labels $\tilde{F}$, for which an additional finding label f_{T+1} is added to F. We list all finding labels that can be appended to F. We filter the finding labels inappropriate for appending F according to the clinical relation between the labels. Some pairs of finding labels have clinically contradictory relations. We filter the labels based on the following two rules.

a. Contradictory Relation. We exclude a pair of contradictory finding labels. For example, the abnormality "Pleural_Effusion.Positive" and the normality "Pleural_Effusion.Negative" must not be included in the same set $\tilde{F}$.

b. Supplementary Relation. We exclude a pair of contradicting finding labels that supplement other finding labels in F. For example, "Pleural_Effusion.Mild" is excluded if "Pleural_Effusion.Positive" not in F.

Step 2: Assign Sample Finding Labels. We sample an additional finding label f_{T+1} to append to F. The label is extracted from a set of candidates by random sampling. The data imbalance is mitigated because the data augmentation process appends a new finding label irrespective of the frequency of this finding labels in the training data.

We use this augmented set of finding labels $\tilde{F}$ for RL. The overall loss function is as follows:

$$L^{all} = \lambda_{rl}(L^{rl} + \lambda_{aug} L^{aug}) + (1 - \lambda_{rl}) L^{xent} \quad (9)$$

where λ_{rl} and λ_{aug} are hyperparameters. L^{aug} denotes the RL loss calculated using the augmented set $\tilde{F}$. L^{aug} is calculated in the same way as L^{rl} with a reward $R(Y)$ under the condition of $\lambda_{rouge} = 0$. This is because no reference report is available for the augmented set $\tilde{F}$. Hence, RL-DA method enables training of the model with more data at a low cost.

4 Experiment

First, to evaluate the effects of our proposed CRS and RL-DA on the data-to-text module, we conduct an experiment with the Japanese Computed Tomography (JCT) dataset. Moreover, to evaluate the generality of CRS and RL-DA and the effects of the modification process on TS-MRGen, we conduct an experiment with the MIMIC-CXR dataset.

4.1 Evaluation on the JCT Dataset

Dataset and Experimental Settings. We evaluate the data-to-text module with the JCT dataset. The JCT dataset has pairs of input sets for finding labels and target medical reports written in Japanese. The JCT dataset is used only to evaluate the data-to-text module. Therefore, we did not prepare medical images for the JCT dataset.

We defined a system of finding labels after consultation with radiologists. Annotators with fully

sufficient knowledge in the radiology report manually annotate the finding labels to the reports. Descriptions that were unrelated to any finding labels in the reports were omitted from preprocessing for privacy reasons.

We chose all hyperparameters based on the CRS scores of the validation data. Details of our models, metrics, training, and dataset are included in the Supplementary section for reproducibility.

We compare the following six text generation models :

(1) **Table-to-Text (Baseline):** Table-to-text model without RL proposed in Section 3.3

(2) **1-NN:** calculates the relevance of input finding labels using TF-IDF and selects the most relevant reports from the training data.

(3) **Rule-Based:** generates reports based on manually prepared templates. We prepared one template sentence per one finding label, and the method concatenates template sentences to construct the entire reports.

(4) **Seq2Seq:** normal encoder-decoder model with GRU.

(5) **CNN-enc**: encoder-decoder model with a CNN encoder.

(6) **Hier-Dec:** encoder-decoder model with a hierarchical decoder (Jing et al., 2019).

Additionally, we compare the following four RL strategies to train the table-to-text text generation model:

(7) **$\mathbf{RL_R}$:** trains the model by RL using only ROUGE as a reward.

(8) **$\mathbf{RL_{CRS}}$:** trains the model by RL using only CRS as a reward.

(9) **$\mathbf{RL_{CRS+R}}$:** trains the model by RL with CRS and ROUGE as a reward.

(10) **RL-$\mathbf{DA_{CRS+R}}$:** trains the model by RL with CRS and ROUGE and applies RL-DA proposed in Section 3.5.

Results. The upper part of Table 1 presents automatic evaluation results regarding the text generation models. The rule-based method obtained the lowest ROUGE-L because it generated considerably redundant reports. Table-to-Text model achieved the best CRS, so we selected the table-to-text model as a text generation module.

The lower part of Table 1 presents automatic evaluation results regarding training strategies. This result demonstrates that application of ROUGE as a reward improves ROUGE-L scores, whereas application of CRS as a reward improves CRS scores. This indicates that RL improves the metric used as a reward. Our proposed RL-$\mathrm{DA_{CRS+R}}$ achieved higher both CRS and ROUGE scores than $\mathrm{RL_{CRS+R}}$.

	ROUGE	BLEU	CRS
Comparison of Text Generation Model			
(1) Table-to-Text (Baseline)	**66.4**	**39.7**	78.2
(2) 1-NN	62.4	33.6	73.1
(3) Rule-Based	60.9	36.2	**80.3**
(4) Seq2Seq	64.6	38.7	76.3
(5) CNN-enc	62.0	37.2	74.1
(6) Hier-Dec	62.1	37.0	77.5
Comparison of Training Strategy			
(7) $\mathrm{RL_R}$	**69.6**	**42.0**	79.3
(8) $\mathrm{RL_{CRS}}$	64.8	38.4	79.4
(9) $\mathrm{RL_{CRS+R}}$	67.2	40.5	80.7
(10) RL-$\mathrm{DA_{CRS+R}}$ (Proposed)	68.2	41.1	**81.3**

Table 1: Automatic evaluation results of the data-to-text module with the JCT dataset. The proposed RL-$\mathrm{DA_{CRS+R}}$ achieved the best score on CRS, whereas $\mathrm{RL_R}$ achieved the best score on BLEU and ROUGE. The CRS of the proposed model is statistically significant compared with the baseline model ($p < 0.01$).

	Correctness			Grammaticality
	P	R	F	
Baseline	89.7	70.2	77.8	94.0
$\mathrm{RL_R}$	92.9	75.1	82.2	**95.5**
RL-$\mathrm{DA_{CRS+R}}$	**95.1**	**75.2**	**83.1**	95.0

Table 2: Comparison of manual evaluation results of the data-to-text module on the JCT dataset. Correctness and fluency scores represent an average of the scores of two workers. P, R, and F denote the precision, recall, and F-score of correctness, respectively. The correctness scores of the proposed model are statistically significant compared with the baseline model ($p < 0.01$).

From the automatic evaluation results, we cannot conclude that our proposed CRS and RL-DA improved correctness because we have no means of deciding which metric is more appropriate for evaluating the generated reports. To estimate the effects of our proposed method on the data-to-text model, we also conducted a manual evaluation. As in previous research (Zhang et al., 2020a), two specialists who are knowledgeable in radiology reports measured 100 randomly selected samples for each experimental condition.

We defined the following metrics for a manual evaluation:

- **Grammaticality:** The percentages of reports that contain no grammatical errors.
- **Correctness:** Measure how well the reports

describe the clinically correct information. We define the correctness of $\hat{Y}$ as an F-score with the following precision and recall:

$$\text{Precision}(\hat{Y}) = \frac{N_{TP}}{N_{TP} + N_{FP}} \quad (10)$$

$$\text{Recall}(\hat{Y}) = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad (11)$$

where N_{TP} indicates the number of findings correctly noted in $\hat{Y}$, N_{FN} indicates the number of missing findings in $\hat{Y}$, and N_{FP} indicates the number of findings mistakenly noted in $\hat{Y}$.

Table 2 presents manual evaluation results. Compared with RL_{R}, our proposed $\text{RL-DA}_{\text{CRS+R}}$ also improves the correctness of manual evaluation. This indicates that proposed $\text{RL-DA}_{\text{CRS+R}}$ does not merely improve the CRS score; it improves the clinical correctness of the generated reports.

4.2 Evaluation on the MIMIC-CXR Dataset

Datasets. We evaluated the data-to-text module and the entire system on the MIMIC-CXR dataset, which includes chest X-ray images and their corresponding medical reports written in English. Notably, these reports include descriptions other than findings, such as indications and impressions. We omitted these descriptions other than findings because these descriptions cannot be generated from the input images. We used the CheXpert dataset (Irvin et al., 2019) to train the image diagnosis module.

We annotated finding labels to the MIMIC-CXR dataset with CheXpert Labeler (Irvin et al., 2019) and the image diagnosis module. CheXpert Labeler annotated findings labels for 14 categories of three types: positive, negative, and uncertain labels. For the training data of the data-to-text module, we labeled the reports using CheXpert Labeler only.

In addition to BLEU metrics, we adopted CheXpert accuracy, precision, and F-score metrics to quantify the correctness of generated reports. This is because the domain-agnostic metrics (such as BLEU) are doubtful in evaluating the quality of reports, and CheXpert-based metrics are more reliable metrics, as reported by (Boag et al., 2019). We chose all hyperparameters based on the F-scores of the validation data. Details of our models, metrics, training, and dataset are described in the Supplementary section for reproducibility.

Experimental Settings. For evaluation, we prepare the following four experimental conditions.

(a) Data-to-Text Evaluation. We provide only the gold finding labels as inputs to the data-to-text module, and then evaluate the generated reports. This evaluation is intended to assess whether our proposed method is also applicable to the MIMIC-CXR dataset or not. Therefore, in this evaluation, we focus only on the data-to-text module. We compare our proposed model $\text{RL-DA}_{\text{CRS+R}}$ which is trained by RL with CRS and ROUGE, and applied RL-DA with the baseline table-to-text model.

(b) End-to-End Evaluation. We compare our TS-MRGen with the end-to-end models, such as CNN-RNN (Boag et al., 2019) and CCR applied models (Liu et al., 2019a). As shown on the left side of Figure 1, the end-to-end model directly generates target reports from the input images. This evaluation setting do not use the finding labels in any way.

(c) Two-Stage Evaluation without Modification. We evaluate our TS-MRGen using the same inputs and outputs as the end-to-end models. As shown in Figure 1, TS-MRGen first predicts the finding labels to describe the findings in the input images. Next, it generates reports from the finding labels. We employ $\text{RL-DA}_{\text{CRS+R}}$ to the data-to-text module of TS-MRGen.

(d) Two-Stage Evaluation with Modification. In addition to (c) above, we apply the modification process to the finding labels predicted by the image diagnosis module. However, it is too expensive to evaluate the model in this condition because the cost of radiologist services is too high. Therefore, we imitate this modification flow using CheXpert Labeler using the following process.

(i) Obtain the output probability vector $p(\hat{f}_t|X)$ of the finding labels predicted by the image diagnosis module.

(ii) Classify the predicted finding labels as confident or untrustworthy according to probability $p(\hat{f}_t|X)$. If $p(\hat{f}_t|X)$ is within the range of $(p_{th}^{low}, p_{th}^{high})$, then we regard the predicted result $\hat{f}_t$ as untrustworthy, and the result is discarded.

(iii) Apply the modification process to the predicted finding labels. We obtain the finding labels using CheXpert Labeler and replace all untrustworthy labels classified in (ii).

This replacement process imitates the modification flow of radiologists.

Results. The upper part of Table 3 presents a comparison related to the data-to-text module. This part of the table shows that our proposed $\text{RL-DA}_{\text{CRS+R}}$

	Evaluation Condition	BLEU 1	BLEU 2	BLEU 3	BLEU 4	CheXpert Acc	CheXpert Prec	CheXpert F (micro)	CheXpert F (macro)
Comparison of Data-to-Text Module									
Baseline (Table-to-Text)	(a)	35.2	23.0	16.1	11.9	92.2	75.9	66.3	50.5
RL-DA$_{\mathrm{CRS+R}}$	(a)	**36.4**	**23.3**	**16.4**	**12.1**	**93.2**	**77.1**	**68.9**	**55.9**
Comparison of Entire Report Generation System									
End-to-End (CNN-RNN) (Boag et al., 2019)	(b)	30.5	20.1	13.7	9.2	83.7	30.4	30.2	18.6
End-to-End (CCR+NLG) (Liu et al., 2019a)	(b)	35.9	**23.7**	**16.4**	11.3	**91.8**	58.6	33.8	18.7
TS-MRGen w/o modification	(c)	21.7	11.8	7.3	4.8	87.3	48.2	29.6	21.7
TS-MRGen with modification	(d)	**36.3**	23.1	16.3	**12.1**	91.7	**71.0**	**63.4**	**49.5**

Table 3: Automatic evaluation of the data-to-text module using the MIMIC-CXR dataset. Acc, Prec, F (micro), and F (macro) indicate accuracy, precision, micro F-score, and macro F-score, respectively. CheXpert scores quantify the correctness of generated reports. For the data-to-text module, our proposed RL-DA$_{\mathrm{CRS+R}}$ achieved the best result (**bold**) for all metrics. For the entire report generation system, TS-MRGen with the modification process improved the correctness of the generated reports. The CheXpert scores of the proposed model and TS-MRGen with modification were statistically significant compared with the baseline model and TS-MRGen without modification ($p < 0.05$), respectively.

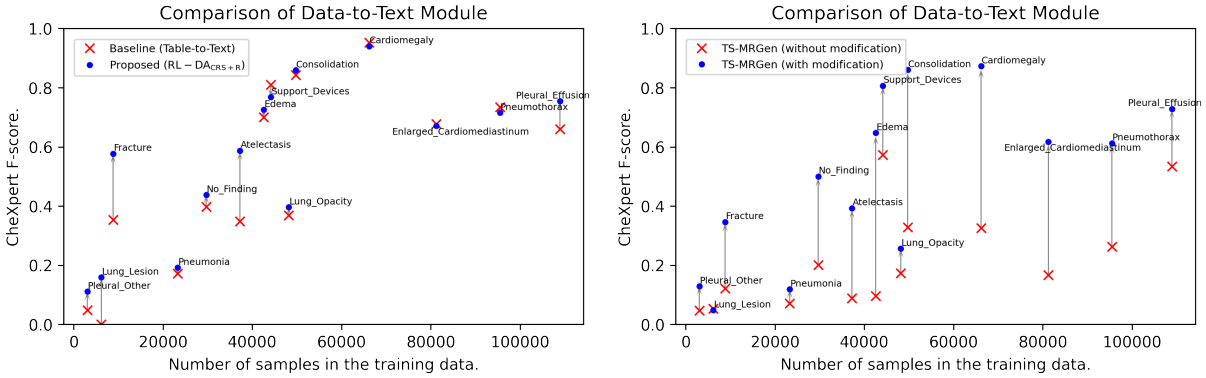

Figure 4: Evaluation of generated reports for each finding label. The horizontal axis shows the frequency of each finding label in the training data. The vertical axis shows the CheXpert F-scores for each finding label. The left plot presents a comparison between the proposed RL-DA$_{\mathrm{CRS+R}}$ and baseline method. Our proposed method improves CheXpert F-scores, especially for infrequent finding labels. The right plot presents an effect of the modification process. Our TS-MRGen presents the important benefit of improving correctness through modification processes.

improves the clinical accuracy of the generated reports for the MIMIC-CXR dataset.

The lower part of Table 3 presents a comparison related to the entire report generation system. Compared with the TS-MRGen without the modification process, the TS-MRGen with the modification process achieved significantly better result for BLEU, CheXpert precision, micro and macro F-scores. CheXpert F-score quantifies the clinical correctness more adequately. Therefore, this result demonstrates that our TS-MRGen has an important advantage because the system enables radiologists to modify the mistakenly predicted finding labels.

5 Discussion

5.1 Effects on an Imbalanced Dataset

Figure 4 presents an evaluation of generated reports for each finding label evaluated using CheXpert Labeler. Both our proposed RL-DA$_{\mathrm{CRS+R}}$ and the baseline method exhibit the same tendency: more infrequent finding labels in the training data are associated with the lower correctness of the generated reports. RL-DA$_{\mathrm{CRS+R}}$ outperforms the baseline model, especially for the infrequent finding labels, This result demonstrates that our proposed RL-DA and CRS generate more accurate reports, especially with infrequent labels in the training data.

5.2 Qualitative Results

The upper part of Table 4 presents an example of a generated report for the JCT dataset. The baseline model generated a report with an incorrect description: "*is accompanied by a pleural indentation*." The data imbalance causes such an error. "Pleural_Indentation.Positive" is more frequent finding label than "Pleural_Indentation.Negative" in

Example of generated reports of JCT dataset.	
Input Finding Labels: Nodule.Positive, Nodule.Solid, Pleural.Indentation.Negative, (truncated) Border.Well_Defined	
Report Generated by the Baseline Model There is a 20 mm dilated nodule in the lung. (truncated) It is well-defined and ***is accompanied by a pleural indentation***.	**Report Generated by the Proposed (RL-DA$_{CRS+R}$) Model** There is a 20 mm dilated nodule in the lung. (truncated) It is well-defined and **there is no pleural indentation**.

Examples of generated reports of MIMIC-CXR dataset.	
Gold Finding Labels: Cardiomegaly.Positive, Enlarged_Cardiomediastinum.Negative, Edema.Negative, Consolidation.Negative, Pneumothorax.Negative, Pleural_Effusion.Negative	
Labels Predicted by Image Diagnosis Module Cardiomegaly.Positive	**Modified Labels** Cardiomegaly.Positive, Enlarged_Cardiomediastinum.Negative, Edema.Negative, Consolidation.Negative, Pneumothorax.Negative, Pleural_Effusion.Negative
Report generated by TS-MRGen without Modification the heart is mildly enlarged. moderate cardiomegaly is unchanged.	**Report Generated by TS-MRGen with Modification** the lungs are clear without focal consolidation. no pleural effusion or pneumothorax is seen. the cardiac silhouette is mildly enlarged. the mediastinal and hilar contours are within normal limits. there is no pulmonary edema.

Table 4: (upper) Example of a report generated from the JCT dataset. The ***italic part*** represents the fault in the baseline model. The **underlined part** represents the correct description corresponding to the ***italic part***. A Japanese-English translation is applied. (lower) Example of a report generated from the MIMIC-CXR dataset. The modification process compensates for the missing labels predicted by the image diagnosis module. It thereby generates a report more faithful to the gold finding labels.

the training data. Therefore, the baseline model mistakenly outputted a more frequently occurring description. However, our proposed RL-DA generated a correct description: "*there is no pleural indentation*". This result demonstrates that our proposed RL-DA and CRS trained the model more accurately on infrequent finding labels.

The lower part of Table 4 presents an example of a generated report for the MIMIC-CXR dataset. Without modification processes, the generated report includes only the description for "Cardiomegaly.Positive." The image diagnosis module has a tendency to omit normalities because the image diagnosis module is not able to train the intention of radiologists of whether normalities are omitted or not. With modification processes, the generated reports include the exact description of the gold finding labels with no omissions. Modification processes correct the missing finding labels to the predicted labels, thereby generating more faithful reports.

6 Conclusion

We proposed a novel Clinical Reconstruction Score (CRS) and Reinforcement Learning and Data Augmentation (RL-DA) methods to train a data-to-text model for an imbalanced dataset. Additionally, we employed a Two-Stage Medical Report Generator (TS-MRGen) for controllable medical report generation from input medical images.

An evaluation of the data-to-text module revealed that our proposed CRS and RL-DA methods improved the clinical correctness of generated reports, especially for infrequent finding labels. An evaluation of the entire medical report generation system revealed that our TS-MRGen generated more correct reports than an end-to-end generation model.

In future work, we would like to explore whether our method is applicable to other domain tasks in data-to-text generation, such as sports summary generation and biography generation tasks.

Acknowledgement

Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used. The authors would like to thank the anonymous reviewers for their constructive reviews and suggestions.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *The 2015 International Conference on Learning Representation*.

Siddharth Biswal, Cao Xiao, Lucas M Glass, M Brandon Westover, and Jimeng Sun. 2020. CLARA: Clinical Report Auto-completion. In *Proceedings of The Web Conference 2020*.

William Boag, Tzu-Ming Harry Hsu, Matthew McDermott, Gabriela Berner, Emily Alsentzer, and Peter Szolovits. 2019. Baselines for Chest X-Ray Report Generation. In *Proceedings of Machine Learning Research Volume 116: Machine Learning for Health Workshop*.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of 2019 IEEE Conference on Computer Vision and Pattern Recognition*.

Jia Deng, Wei Dong, R Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

Philipp Harzig, Yan-Ying Chen, Francine Chen, and Rainer Lienhart. 2019. Addressing data bias problems for chest x-ray image report generation. *Proceedings of the British Machine Vision Conference 2019*.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. 2019. CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Thirty-Third AAAI Conference on Artificial Intelligence*.

Baoyu Jing, Zeya Wang, and Eric Xing. 2019. Show, Describe and Conclude: On Exploiting the Structure Information of Chest X-ray Reports. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Baoyu Jing, Pengtao Xie, and Eric Xing. 2018. On the Automatic Generation of Medical Imaging Reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. MIMIC-CXR: A large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*.

Chris Kedzie and Kathleen McKeown. 2019. A Good Sample is Hard to Find: Noise Injection Sampling and Self-Training for Neural Language Generation Models. In *Proceedings of the 12th International Conference on Natural Language Generation*.

Y Keneshloo, T Shi, N Ramakrishnan, and CK Reddy. 2019. Deep Reinforcement Learning for Sequence-to-Sequence Models. *IEEE transactions on neural networks and learning systems*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of Workshop on Text Summarization Branches Out*.

Guanxiong Liu, Tzu-Ming Harry Hsu, Matthew McDermott, Willie Boag, Wei-Hung Weng, Peter Szolovits, and Marzyeh Ghassemi. 2019a. Clinically Accurate Chest X-Ray Report Generation. In *Machine Learning for Healthcare Conference 2019*.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2019b. On the Variance of the Adaptive Learning Rate and Beyond. *arXiv preprint arXiv:1908.03265*.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2018. Table-to-text generation by structure-aware seq2seq learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shuming Ma, Pengcheng Yang, Tianyu Liu, Peng Li, Jie Zhou, and Xu Sun. 2019. Key Fact as Pivot: A Two-Stage Model for Low Resource Table-to-Text Generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Maram Mahmoud A Monshi, Josiah Poon, and Vera Chung. 2020. Deep learning in generating radiology reports: A survey. *Artificial Intelligence in Medicine*, page 101878.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-Step: Separating Planning from Realization in Neural Data-to-Text Generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

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 on Association for Computational Linguistics*.

Yifan Peng, Xiaosong Wang, Le Lu, Mohammadhadi Bagheri, Ronald Summers, and Zhiyong Lu. 2018. Negbio: a high-performance tool for negation and uncertainty detection in radiology reports. *AMIA Summits on Translational Science Proceedings*, 2018:188.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition*.

Toshinori Sato, Taiichi Hashimoto, and Manabu Okumura. 2017. Implementation of a word segmentation dictionary called mecab-ipadic-NEologd and study on how to use it effectively for information retrieval. In *Proceedings of the 23th Annual Meeting of the Association for Natural Language Processing*.

Mingxing Tan and Quoc Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*.

Sam Wiseman, Stuart M Shieber, and Alexander M Rush. 2017. Challenges in Data-to-Document Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

Yixiao Zhang, Xiaosong Wang, Ziyue Xu, Qihang Yu, Alan Yuille, and Daguang Xu. 2020a. When Radiology Report Generation Meets Knowledge Graph. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Yuhao Zhang, Derek Merck, Emily Bao Tsai, Christopher D Manning, and Curtis P Langlotz. 2020b. Optimizing the Factual Correctness of a Summary: A Study of Summarizing Radiology Reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

A Supplementary Material

A.1 Dataset and Preprocessing

the JCT Dataset. We built the JCT dataset to train the data-to-text module of the medical report generation system. For the JCT dataset, we collected 4,454 medical reports regarding pulmonary nodules from a hospital. To train an accurate medical report generation system, we focused only on the findings in the reports and excluded the sentences that violated patient privacy. During a consultation with radiologists, we defined 57 types of finding labels. As preprocessing, all descriptions that were not related to any findings were truncated by annotators. We lexicalized phrases referring to the existence of nodules and phrases referring to the size of the nodules to improve the stability of training of the data-to-text generation model. We used MeCab [1] and mecab-ipadic-NEologd (Sato et al., 2017) to tokenize the reports, and keep tokens with 2 or more occurrences.

To prevent data leakage in validation/test datasets, we split the dataset in a way to ensure that the same sets of finding labels are not included in the training, validation, and test data. Additionally, to avoid the negative influence of the imbalanced frequency of sets of finding labels, we omitted the samples with duplicated sets of finding labels in the validation/test dataset. These strategies for data splitting and duplicate input handling caused differences in average labels and lengths, as shown in Table 5. If samples contained shorter sentences and fewer input labels, the validation and test datasets tended to contain longer sentences and a greater number of input labels.

the MIMIC-CXR Dataset. Medical reports in the MIMIC-CXR dataset [2] contain descriptions that are irrelevant to the findings in the input images. Hence, we extracted the finding sections of the reports using the scripts provided in Boag et al. (2019) [3]. In training data, we truncated the sentences in the reports that were not related to any findings using CheXpert Labeler and NegBio (Peng et al., 2018) parser to improve the stability of training the model. We omitted the reports that did not mention any findings or had no finding sections from the training data. Note that the reports in the validation and test data may contain a description that does not mention any findings. We

[1] https://taku910.github.io/mecab/

[2] https://physionet.org/content/mimic-cxr/2.0.0/

[3] https://github.com/wboag/cxr-baselines

	Number of Reports	Average labels	Average length
the JCT dataset			
Training data	3,637	4.71	27.5
Validation data	418	9.46	52.7
Test data	399	9.49	51.4
the MIMIC-CXR dataset			
Training data	131,016	4.92	43.6
Validation data	1,156	4.90	44.5
Test data	2,299	5.01	54.7

Table 5: Statistics of the JCT dataset and the MIMIC-CXR dataset.

dataset	JCT	MIMIC-CXR
Data-to-Text Module Hyperparameters		
Vocabulary size	339	2222
Number of labels	57	40
Dropout rate	0.2	0.2
Word embedding size	32	64
Label embedding size	16	16
Hidden size	32	32
Beam search width	5	5
Training Hyperparameters		
Batch size	32	32
Optimizer	Adam	Adam
Learning rate	5.0×10^{-3}	2×10^{-4}
Learning rate decay	0.99	0.98
λ_{rouge}	0.2	0.2
λ_{rl}	0.2	0.03
λ_{aug}	0.1	0.05
τ (Softmax temperature)	0.5	0.4
Dropout	0.2	0.2
Gradient clipping	2.0	2.0

Table 6: List of hyperparameters of the data-to-text modules.

use this approach to align our experimental conditions with previous end-to-end research Boag et al. (2019). We used the Natural Language Toolkit [4] to tokenize the reports, and keep tokens with 10 or more occurrences. We have split the dataset into train, validation, and test data based on the split distributed in the MIMIC-CXR-JPG (Johnson et al., 2019) [5] dataset. Table 5 presents the statistics of the MIMIC-CXR dataset.

A.2 Training Details

Image Diagnosis Module All images were fed into a network with a size of 512×512 pixels. We set up the loss as the sum of the multi-class cross-entropy for each observations and used the RAdam (Liu et al., 2019b) optimizer with a learning rate of 1.0×10^{-4}. We trained the model for 5 epochs with the CheXpert dataset (Irvin et al., 2019).

Subsequently, we evaluated the image diagnosis module with the CheXpert dataset. To evaluate the accuracy of image classification correctly for the infrequent labels, we performed a 5-fold cross-validation. Table 7 presents F-scores for each finding labels evaluated in 5-fold cross-validation. Although the F-scores of the no-mention labels are high, the F-scores of the positive, negative, and uncertain finding labels are relatively low. This is because the CheXpert dataset is significantly imbalanced, and almost all finding labels in the training data are in the no-mention category.

Data-to-Text Module For the JCT and MIMIC-CXR datasets, we trained the data-to-text module for 50 and 20 epochs, respectively. We used a CRS score of the validation data as the stopping criteria. Finally, we reported evaluation scores that achieved the highest CRS score on the validation data. Table 6 presents hyperparameters used to train our models. Before we trained the model with RL, we pretrained the model with only cross-entropy loss for an epoch. The number of parameters of the data-to-text module was 127k for the JCT dataset and 463k for the MIMIC-CXR dataset.

Reconstructor Module To train the reconstructor for the JCT dataset, we used the pretrained Japanese BERT model [6]. We have split the training data of the data-to-text module into 4:1 and used the former part as training data and the latter part as validation data for the reconstructor. For fine-tuning, we used the AdamW optimizer with a learning rate of 2.0×10^{-5} for the BERT layer and 2.0×10^{-3} for the fully connected layer. We used binary cross-entropy loss to train the model, and applied Class Balanced Loss (CBL) (Cui et al., 2019) with $\beta = 0.999$. The number of parameters of the reconstruction module is 110M. We fine-tuned the model with 10 epochs, and the F-score on the validation dataset was 90.3.

To train the reconstructor for the MIMIC-CXR dataset, we use the pretrained bert-base-uncased model. We also verified the BioBERT model (Lee et al., 2020), but the results showed no significant differences with the bert-base-uncased model. For fine-tuning, we used the AdamW optimizer with a learning rate 2.0×10^{-5} for the BERT layer and 2.0×10^{-3} for the fully connected layer. By analogy with the JCT dataset, we have split the training data into 4:1 and used the former part as the training data and the latter part as the validation data

[4] https://www.nltk.org/

[5] https://physionet.org/content/mimic-cxr-jpg/2.0.0/

[6] https://github.com/cl-tohoku/bert-japanese

Labels	Negative	Positive	Uncertain	No_Mention
No_Finding	-	0.468	-	0.907
Enlarged_Cardiomediastinum	0.436	0.197	0.040	0.858
Cardiomegaly	0.209	0.525	0.013	0.873
Lung_Opacity	0.002	0.696	0.000	0.602
Lung_Lesion	0.150	0.246	0.092	0.936
Edema	0.223	0.615	0.254	0.740
Consolidation	0.489	0.215	0.254	0.740
Pneumonia	0.008	0.163	0.278	0.883
Atelectasis	0.002	0.333	0.325	0.713
Pneumothorax	0.458	0.513	0.000	0.770
Pleural_Effusion	0.524	0.759	0.036	0.639
Pleural_Other	0.335	0.217	0.165	0.963
Fracture	0.234	0.207	0.007	0.890
Support_Devices	0.046	0.844	0.007	0.771
Overall F1-Score	0.240	0.428	0.103	0.807

Table 7: Evaluation of the image diagnosis module for each finding label. All scores are measured by F-score in 5-fold cross validation.

Dataset	JCT	MIMIC-CXR
Optimizer	AdamW	AdamW
Learning rate of BERT layer	2.0×10^{-5}	2.0×10^{-5}
Learning rate of FC layer	2.0×10^{-3}	1.0×10^{-4}
CBL β (Cui et al., 2019)	0.999	0.999
Warm up steps	200	200

Table 8: List of hyperparameters of the reconstructor modules.

for the reconstructor. We used binary cross-entropy loss to train the model, and applied Class Balanced Loss (CBL) (Cui et al., 2019) with $\beta = 0.999$. The number of parameters of the reconstruction module was 109M. We fine-tuned the model with 10 epochs, and the F-score on the validation dataset was 97.9.

We used an Intel Core i7-6850K CPU and NVIDIA GTX 1080Ti GPU for training on the JCT dataset, and the training time was approximately 3 h. We used an Intel Xeon Gold 6148 CPU and NVIDIA Tesla V100 GPU for training on the MIMIC-CXR dataset, which required approximately 12 hours.

A.3 Evaluation Settings.

We use an approximate randomization test [7] to evaluate the statistical significance.

Evaluation Metrics on the JCT Dataset. For automatic evaluation on the JCT dataset, we used BLEU (Papineni et al., 2002), F-scores of ROUGE-L (Lin, 2004), and CRS as metrics. We used Natural Language Toolkit [8] to calculate BLEU scores, and the ROUGE Python library [9] to calculate ROUGE-L scores.

Evaluation Metrics on the MIMIC-CXR Dataset. For comparison with the previous image captioning approaches, we used BLEU-1, BLEU-2, BLEU-3, and BLEU-4 metrics calculated by the nlg-eval [10] library. However, word-overlap based metrics, such as BLEU, fail to assume the factual correctness of generated reports. We compared the labels assigned in CheXpert Labeler between the generated reports and gold reports to calculate the CheXpert accuracy, precision, micro F-score, and macro F-score. The micro F-score was obtained by the overall numbers of true positives, false positives, and false negatives. The macro F-score was obtained by the average of F-scores per class label. Although the micro F-score neglects infrequent labels, the score is significantly biased by the imbalanced distribution of the test dataset.

Note that precision and F-score are preferred to evaluate the clinical correctness of the reports in CheXpert. In contrast, CheXpert accuracy does not quantify the clinical correctness of the generated reports adequately. The imbalanced dataset results in an excessive number of true negatives rather than true positives. Hence, CheXpert accuracy overestimates the clinical correctness of generated reports if the reports comprise many descriptions that are not related to the findings.

Modification Flow We apply the modification process to the image diagnosis module result with the parameters of $(p_{th}^{low}, p_{th}^{high}) = (0.1, 0.9)$ for the positive finding labels. However, we regard all neg-

[7] https://github.com/smartschat/art
[8] https://www.nltk.org/
[9] https://github.com/pltrdy/rouge
[10] https://github.com/Maluuba/nlg-eval

ative and uncertain labels predicted by the image diagnosis module as unreliable. This is because negative or uncertain findings are highly dependent on the radiologist's judgment.