Learning to Guide Human Decision Makers with Vision-Language Models

Anonymous authors
Paper under double-blind review

Abstract

There is growing interest in AI systems that support human decision-making in *high-stakes* domains (e.g., medical diagnosis) to improve decision quality and reduce cognitive load. Mainstream approaches pair human experts with a machine-learning model, offloading low-risk decisions to the model so that experts can focus on cases that require their judgment. This ***separation of responsibilities*** setup, however, is inadequate for high-stakes scenarios. The expert may end up over-relying on the machine's decisions due to *anchoring bias*, thus losing the human oversight that is increasingly being required by regulatory agencies to ensure trustworthy AI. On the other hand, the expert is left entirely unassisted on the (typically hardest) decisions on which the model abstained. As a remedy, we introduce ***learning to guide*** (LTG), an alternative framework in which – rather than taking control from the human expert – the machine provides *guidance* useful for decision making, and the human is entirely responsible for coming up with a decision. In order to ensure guidance is *interpretable* and *task-specific*, we develop SLOG, an approach for turning *any* vision-language model into a capable generator of textual guidance by leveraging a modicum of human feedback. Our empirical evaluation highlights the promise of SLOG on both on a synthetic dataset and a challenging, real-world medical diagnosis task.

1 Introduction

High-stakes applications in healthcare, criminal justice and policy making can substantially benefit from the introduction of AI technology, yet full automation in these scenarios is not desirable, due to ethical, safety and legal concerns, if not explicitly forbidden by law (Government of Canada, 2019; European Commission, 2021). For these reasons, human-AI or ***Hybrid Decision Making*** (HDM) is becoming increasingly popular to tackle high-stakes tasks. HDM algorithms pair a human decision maker with an AI agent – often a machine learning model – capable of providing support, with the goals of improving *decision quality* and lowering *cognitive effort*.

Most current approaches to HDM follow a principle of ***separation of responsibilities***, in the sense that they route novel inputs to exactly one of the two agents – *either* the human *or* the AI – who is then responsible for coming up with a decision. Specifically, in existing approaches (Madras et al., 2018; Mozannar & Sontag, 2020; Keswani et al., 2022; Verma & Nalisnick, 2022; Liu et al., 2022; Wilder et al., 2021; De et al., 2020; Raghu et al., 2019; Okati et al., 2021), the AI first assesses whether an input can be handled in autonomy – e.g., it is low-risk or can be addressed with confidence – and defers to a human partner otherwise. These algorithms help humans focus on the cases the model flags as most needing attention.

We argue that this setup is *suboptimal* and potentially *unsafe*. It is suboptimal because, whenever the machine opts for deferral, the human is left resolving hard cases completely unassisted (as in Fig. 1, right). At the same time, it is unsafe, because humans are affected by *anchoring bias* (Rastogi et al., 2022; Eigner & Händler, 2024), a phenomenon whereby decision makers tend to blindly rely on an initial impression (the anchor) and refrain from exploring alternative hypotheses. When the anchor is provided by an algorithm, the bias is amplified as humans tend to over-trust the machine's decisions when available and ignore their own opinions, a phenomenon called *automation bias* (Cummings, 2012) (Fig. 1, middle). This effectively

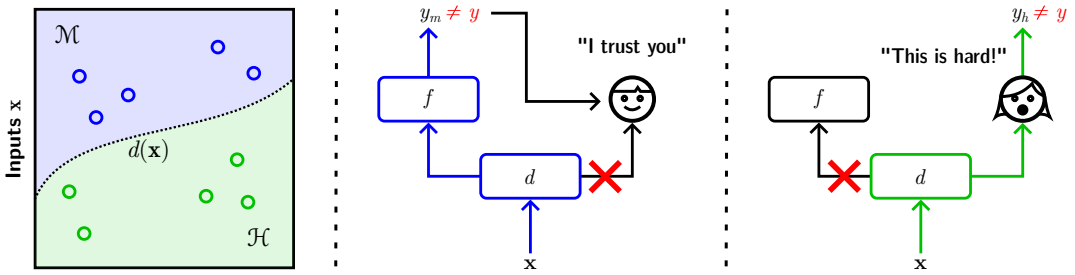


Figure 1: **Left**: Existing HDM algorithms employ a deferral function $d(\mathbf{x})$ to *partition* the input space $\mathcal{X}$ into $\mathcal{H}$ and $\mathcal{M}$. **Middle**: A predictor $f(\mathbf{x})$ handles the inputs falling in $\mathcal{M}$ (in **blue**). Because of *anchoring bias*, the human expert may end up blindly trusting its (possibly poor) decisions y_m. **Right**: The human, on the other hand, is left completely unassisted for those (possibly hard) decisions falling in $\mathcal{H}$, increasing the chance of mistakes in the human's decisions y_h (in **green**).

undermines *human oversight* over algorithmic decisions, which is increasingly being required by governments around the world to regulate the use of AI in high-stakes applications (Green, 2022).

As a remedy, we propose ***learning to guide*** (LTG), an alternative setup that side-steps these issues. In LTG, the machine is trained to supply its human partner with interpretable *guidance* highlighting those aspects of the input that are useful for coming up with a high-quality decision. For instance, in pathology prediction, the guidance highlights the pathologies present in an input X-ray scan that are indicative of possible diagnoses. In LTG, *by construction*, all decisions are taken by the human expert – thus preventing automation bias – but facilitated by accompanying machine guidance.

We showcase LTG on ***medical decision making*** focusing on guidance formulated in ***natural language***. Along with that, we also validated the effectiveness of LTG with a synthetic dataset. To this end, we introduce SLOG (Surrogate-based Learning to Guide), an algorithm for turning large vision-language models (VLMs) (Radford et al., 2021; Yan & Pei, 2022; Sharma et al., 2021) into high-quality guidance generators. In a nutshell, SLOG takes a VLM pre-trained for caption generation and fine-tunes it using feedback about the quality of downstream human decisions inferred from generated guidance. SLOG keeps annotation costs under control by training a *surrogate model* that predicts downstream decision quality on a modest amount of feedback, and then using it to fine-tune the VLM in an end-to-end fashion. Our experiments on a challenging medical diagnosis task indicate that VLMs fine-tuned with SLOG output interpretable task-specific guidance that can be used to infer high-quality decisions.

Contributions. In summary, we:

- Expose critical limitations in prevailing HDM algorithms that undermine their suitability for high-stakes decision-making.
- Propose *learning to guide* (LTG), a novel approach for assisting human decision-makers that keeps humans continuously in the loop.
- Present SLOG, an LTG approach tailored for natural language guidance that can convert large VLMs into interpretable, task-specific guidance generators.
- Demonstrate the effectiveness of SLOG on a challenging medical diagnosis task.

2 Hybrid Decision Making

We target decisions that must retain human oversight (e.g., medical diagnosis) because full automation poses safety risks.[1] Research on HDM develops AI assistants to augment human experts on such tasks.

[1]We focus on classification problems, with inputs $\mathbf{x} \in \mathbb{R}^d$ and categorical or multi-label decisions y. Despite this, our remarks apply to other prediction problems as well, e.g., regression (De et al., 2020).

Considering the AI assistant and the human expert have different abilities, expertise, and biases, the central question of HDM is how to best integrate them.

HDM by Separation of Responsibilities. Existing HDM strategies solve this problem by following a principle of *separation of responsibilities*: any given instance $\mathbf{x}$ is assigned to exactly one of the two agents, who is then in charge of decision making, cf. Fig. 1. Specifically, they implement a *classifier* $f : \mathbf{x} \mapsto \hat{y}$, playing the role of an AI agent, as well as a *deferral policy* $d : \mathbb{R}^d \to \{\texttt{machine}, \texttt{human}\}$ that partitions the input domain $\mathcal{X}$ into two disjoint subsets, $\mathcal{M}$ and $\mathcal{H}$. Novel inputs $\mathbf{x}$ falling in the former are handled by f and those falling in the latter are handled by the human expert. This setup is known under a variety of names, including *learning to defer* (Madras et al., 2018; Mozannar & Sontag, 2020), *learning under algorithmic triage* (Raghu et al., 2019; Okati et al., 2021), *learning under human assistance* (De et al., 2020; 2021), and *learning to complement* (Wilder et al., 2021; Bansal et al., 2021).

Approaches differ in how they partition the input space $\mathcal{X}$. Earlier methods build on *prediction with a reject option* (Cortes et al., 2016), in which the deferral policy d observes all incoming instances $\mathbf{x}$ and offloads those about which the predictor f is unsure (based on, e.g., predictive variance) (Raghu et al., 2019). Since f is fixed, the partition is static and depends only on the self-assessed uncertainty of the predictor. Assuming the latter is sufficiently well calibrated (Kendall & Gal, 2017), this strategy can perform well in practice (Liu et al., 2022). The main drawback with this setup is that the partitioning accounts for the machine's performance only, neglecting the human's expertise and biases. Madras et al. (2018) improve on this by *learning* the deferral policy d so that it optimizes some decision theoretic measure of *joint team performance*, thus explicitly taking the quality of human decisions into account. Follow-up works (De et al., 2020; 2021; Wilder et al., 2021) go one step further and train the deferral policy d and the predictor f *jointly*, so as to adapt one to the other. This setup has been extended to incremental (Keswani et al., 2021) and sequential (Joshi et al., 2021) decision making, and to bandit feedback (Gao et al., 2021). Theoretical studies have analyzed the consistency (Mozannar & Sontag, 2020) and calibration (Verma & Nalisnick, 2022) of the HDM pipeline and the structure of optimal deferral policies (Okati et al., 2021).

Issues with Separation of Responsibilities. At a high level, an HDM strategy should satisfy the following desiderata:

- **D1**. *Complementarity.* It should leverage the complementary capabilities of each agent to obtain better decisions *on average*, or equally good decisions at a lower cognitive cost, than each agent individually.
- **D2**. *Synergy.* It should combine the contributions of each agent to obtain better *individual* decisions, or equally good decisions at a lower cognitive cost, than each agent individually.
- **D3**. *Reliability.* It should produce decisions that are more reliable than those made by each agent individually.

Existing HDM approaches aim at enabling complementarity (**D1**). In fact, the main benefit of offloading decisions to an AI is that of lowering the human's cognitive effort. Moreover, depending on the relative performance of the predictor on inputs in $\mathcal{M}$ compared to the expert, they can also improve the quality of the team's decisions (on average across inputs, not necessarily for all inputs). Under suitable conditions, learning to defer can *provably* do so (Donahue et al., 2022). However, approaches complying with separation of responsibilities completely overlook synergy (**D2**). When the machine outputs a decision, the human is tempted to simply stick to it, thus *over-relying on the machine's decisions*, because of the previously mentioned anchoring (Rastogi et al., 2022) and automation (Cummings, 2012) biases. Conversely, whenever the machine opts for deferral, *the human is left resolving hard cases completely unassisted.* This also compromises reliability (**D3**), which is key in high-stakes applications, thus hindering the applicability of HDM.

3 Beyond Separation of Responsibilities with Learning to Guide

We propose *learning to guide* (LTG), a novel HDM framework that addresses the shortcomings of existing strategies by foregoing separation of responsibilities. In a nutshell, LTG aims to learn a *guidance generator* $\gamma(\mathbf{x})$, implemented as a machine learning model, that given an input $\mathbf{x}$, outputs guidance g that is useful

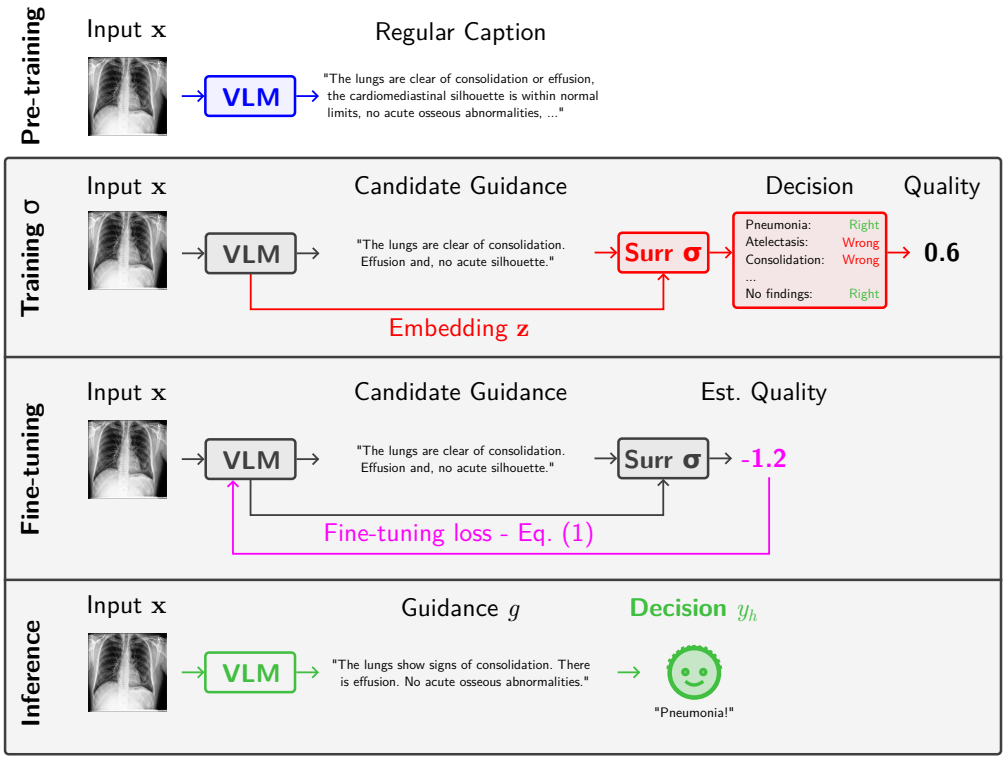


Figure 2: **The slog approach to learning to guide**. **Tier 1**: First we take a VLM (in **blue**) pre-trained to generate captions of visual inputs $\mathbf{x}$. **Tier 2**: Next, we train the surrogate σ_{quality} (in **red**) to estimate the quality of downstream decisions using a modicum of annotated guidance-quality pairs. The surrogate takes both images and text (embeddings) as input. **Tier 3**: Given a trained surrogate σ_{quality}, we fine-tune the VLM (in **magenta**) to output guidance g achieving high (estimated) decision quality. **Tier 4**: The fine-tuned VLM (in **green**) can readily be used for generating useful textual guidance.

for *assisting human decision making* on that input. In medical diagnosis, for instance, given a chest X-ray scan $\mathbf{x}$, the guidance g might describe pathology visible in the image that are useful for identifying pathologies and prescribe treatment, as shown in Fig. 2 (top). Critically, and in stark contrast with existing HDM approaches, in LTG the machine does not replace the human: *the final decision is always taken by the human partner, in collaboration with the AI*. This means the decision maker is always in the loop and responsible for the final decision.

Desiderata for Guidance. In order to support human decision making, guidance should satisfy the following natural properties:

D4. *Interpretability*. It should be *understandable* for the human expert at hand.

D5. *Informativeness*. It should be *informative* for the decision at hand.

If these are satisfied, then guidance can be used by human experts to address a specific downstream decision making task. Note that, satisfying these desiderata encourages satisfaction of **D1**–**D3**. In fact, if guidance is interpretable (**D4**) and extracts decision-relevant elements from the input (**D5**), it should help the human in taking accurate decisions on individual instances (**D2**) and as a consequence improving the average quality of the decisions being made (**D1**). Additionally, interpretability helps the human in judging the quality of the guidance received, and thus evaluate the reliability of the overall decision (**D3**).

Learning to Guide for VLMs. In this paper we focus on *textual guidance* expressed in natural language as a mean to enable interpretability (**D4**). Motivated by their state-of-the-art performance in text generation

tasks (Wei et al., 2022) and by their promise in pathological report generation (Shamshad et al., 2023; Chen et al., 2020; 2021; Hou et al., 2021; Kayser et al., 2022; Yunxiang et al., 2023; Bazi et al., 2023; Drozdov et al., 2020; Yan & Pei, 2022), we propose to leverage *vision-language models* (VLM) to generate guidance.

Off-the-shelf VLMs are not conceived for generating guidance for *specific* decision making tasks, and thus violate informativeness (**D5**). Clearly, a perfectly accurate medical report is also an optimal guidance for follow-up decisions, but generating highly accurate reports requires massive amounts of supervision, and reports generated by specialized VLMs are far from perfect (Shamshad et al., 2023).

The question is then how to *convert* such models into high-quality guidance generators. Focusing on (medical) decision making from image data, we address this problem by introducing SLOG, a novel approach for *turning vision-language models into guidance generators using human feedback* designed to comply with **D1**–**D5**. The rationale behind SLOG is to encourage VLMs to focus on accurately reporting those aspects of the input image that are most relevant *for the follow-up decisions*, possibly overlooking less important details. Next, we briefly discuss how SLOG uses annotations and then proceed to outline the main algorithm.

3.1 Estimating Downstream Decision Quality

Optimizing guidance for synergy (**D2**) requires knowing the quality of downstream decisions taken by a human expert supplied with the guidance itself. SLOG assumes access to quality ratings $\mathbf{q} \in [0,1]^d$, where each q_i encodes the quality of a downstream decision. For instance, if the expert has to determine the state of two conditions (e.g., "`pneumonia`" and "`fracture`"), then $d = 2$. Quality ratings for expert decisions can be obtained by comparing these against a gold standard (using, e.g., decision accuracy) or by consulting a second expert (using, e.g., a star rating system).

Clearly, there is a tension between the number of annotations necessary for fine-tuning a VLM and the cost of eliciting such annotations. SLOG addresses this issue by training a *surrogate model* $\sigma_{\text{quality}} : (\mathbf{x}, \mathbf{z}) \mapsto \widehat{\mathbf{q}}$ using a modicum of annotated quality ratings, and using it to estimate the quality of guidance g generated by the VLM during fine-tuning. In practice, SLOG fits the surrogate on a training set $\mathcal{D}_{\text{surr}} = \{(\mathbf{x}_i, \mathbf{z}_i, q_i)\}$, where $\mathbf{x}_i$ is an input image, $\mathbf{z}_i$ is the embedding of the VLM's guidance g_i for that input, and q_i is the quality of that guidance, by minimizing an average cross-entropy loss of the form:

$$\frac{1}{|\mathcal{D}_{\text{surr}}|} \sum_{(\mathbf{x},\mathbf{z},\mathbf{q}) \in \mathcal{D}_{\text{surr}}} \frac{1}{d} \sum_{i=1}^{d} \mathsf{CE}(q_i, \sigma_{\text{quality}}(\mathbf{x}, \mathbf{z})_i) \tag{1}$$

3.2 The SLOG Loop

In essence, SLOG takes a pre-trained VLM caption generator γ and fine-tunes it for a number of rounds T. Let $\mathcal{D}_{\text{train}}$ be a data set of image-caption pairs (for instance, a subset of the data that γ was trained on) and $\mathcal{D}_{\text{tune}}$ a larger set of *unlabeled* images from the target decision making task. In each round $t = 1, \ldots, T$, SLOG samples a batch $\{\mathbf{x}_1, \ldots, \mathbf{x}_B\}$ from $\mathcal{D}_{\text{tune}}$ uniformly at random with replacement, and computes guidance $g_i^t = \gamma(\mathbf{x}_i)$ and embeddings $\mathbf{z}_i^t$ for each input. Then, it evaluates the quality of the generated guidance using the frozen surrogate σ_{quality} and fine-tunes the VLM γ by minimizing an augmented loss of the form:

$$\mathsf{CE}(\gamma, \mathcal{D}_{\text{train}}) - \frac{\lambda}{|\mathcal{D}_{\text{tune}}|} \sum_{(\mathbf{x},\mathbf{z}) \in \mathcal{D}_{\text{tune}}} \frac{1}{d} \sum_{i=1}^{d} \sigma_{\text{quality}}(\mathbf{x}, \mathbf{z})_i \tag{2}$$

for a given number of epochs. Eq. (2) trades off text generation performance on the training set $\mathcal{D}_{\text{train}}$ – so as to discourage catastrophic forgetting – and estimated quality of downstream decisions on the fine-tuning set $\mathcal{D}_{\text{tune}}$. Here, $\lambda > 0$ is a hyper-parameter. Fine-tuning then amounts to applying gradient descent to batches comprising training and fine-tuning examples in equal proportions. Once done, the SLOG loop repeats. As long as the surrogate generalizes the quality rating annotations, the VLM gradually learns to output text that works well as guidance tailored for the target decision task.

3.3 Benefits and Limitations

In stark contrast with existing HDM strategies, SLOG ensures that the human receives guidance useful for decision making while keeping them in the loop. The cognitive load of LTG is entirely devoted to ensuring it can be safely employed in high-stakes applications, where there is little room for mistakes and humans *have* to be in control at all times (Zhang et al., 2020), as increasingly prescribed by legal frameworks (Government of Canada, 2019; European Commission, 2021). LTG and SLOG are designed explicitly for supporting HDM in these cases. SLOG is reminiscent of mainstream approaches to LLM alignment, such as reinforcement learning with human feedback (RLHF) (Ziegler et al., 2020; Ouyang et al., 2022), but differs from them in aims and technology. While RLHF strives to improve factuality and reduce harmfulness of generated content (Ouyang et al., 2022), ignoring the decisions these impinge on, SLOG specifically aims at improving quality of downstream human decisions for a specific decision making task. At the same time, SLOG foregoes reinforcement learning approaches (Schulman et al., 2017) in favor of a simpler and more direct end-to-end fine-tuning strategy.

One limitation of SLOG is that the performance of the guidance generator hinges on that of the surrogate, which in turn relies on the amount of quality ratings available for training. In Section 4 we present an ablation study showing that a limited amount of quality annotations are sufficient for SLOG to improve generated guidance. Another limitation of SLOG is that it currently assumes quality ratings are readily available, which is not always the case. One option is then to integrate SLOG with active learning strategies (Settles, 2012; Herde et al., 2021) to acquire informative quality ratings whenever needed. Doing so is however outside the scope of this paper and left to future work. Finally, the guidance output by VLMs may suffer from hallucination, that is, it may contain untrue statements. However, SLOG directly maximizes factuality of guidance on the fine-tuning set $\mathcal{D}_{\text{tune}}$, meaning that a simple way of reducing the chance of non-factual statements is to employ a larger fine-tuning set. This is relatively cheap to do, as no annotations are required. Moreover, large language models can be surprisingly well-calibrated (Kadavath et al., 2022), meaning that generated guidance can be filtered based on the VLM's own uncertainty estimates to prevent over-reliance (Eigner & Händler, 2024) and avoid low-quality decisions (Zhang et al., 2020).

4 Empirical Analysis

In this section, we investigate the following research questions:

Q1 Does SLOG work in a controlled setting?

Q2 Does SLOG improve generated guidance despite relying on a surrogate model?

Q3 Does SLOG improve the quality of the decisions made using its guidance?

Q4 Does SLOG help improve performance even when examined by a human physician?

In order to answer **Q1**, we investigate the effectiveness of SLOG in a controlled experimental setup using the ClevR dataset (Johnson et al., 2017). For the remaining questions, we employ a real world medical decision making dataset. We discuss the specifics of all experiments in the following. We will make all source code public upon acceptance of the manuscript.

4.1 Q1: The ClevR Task

Data set. We first evaluate SLOG on a variant of the CLEVR (Johnson et al., 2017) dataset. This consists of $60k$ images for training, $10k$ for validation and $15k$ for testing. Each image represents several three-dimensional objects with different shapes, colors, sizes, materials, and positions, and comes with a structured (non-textual) description of its contents. We translate these into natural language to obtain ground-truth textual descriptions, see Fig. 3 (left) for an example.

Decision-making task. We are interested in evaluating SLOG's ability of producing high-quality guidance for decision-makers. Since the CLEVR data set comes with no pre-specified decision task, we define one

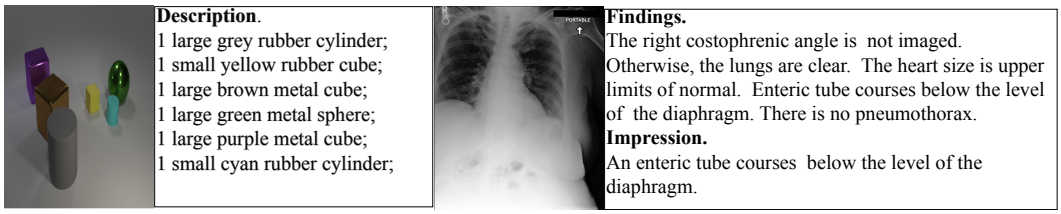


Figure 3: **Left**: Example CLEVR image and corresponding textual description. **Right**: Example of `Mimic-CXR-IV` radiograph and corresponding medical report, comprising *findings* and*impression* sections.

ourselves. Specifically, we construct the following six rules about the presence of certain objects in the image:

Rule 1 Does the image contain one large green sphere **or** one small rubber cube?

Rule 2 Does the image contain one large red sphere **or** one green object?

Rule 3 Does the image contain one large rubber cube **and** one sphere?

Rule 4 Does the image contain one rubber cylinder **and** two small objects?

Rule 5 Does the image contain one small red metal cube **or** two rubber cylinders?

Rule 6 Does the image contain one sphere **and** two small metal objects?

Each rule allows us to define positive images (i.e., those that satisfy the rule) and negative images (i.e., those that do not), yielding six labels per image. We construct these rules based on the frequency of object features in the images, selecting conditions with a balanced positive/negative ratio. Given an image, the decision task amounts to predicting what rules will fire.

In a nutshell, in this experiment we first train a VLM on the ground-truth (decision-agnostic) captions, then construct a surrogate model that simulates the human decision-maker, and then fine-tune the VLM with SLOG (Equation (2)) to produce guidance. Next, we detail each part of the pipeline.

Vision-language Model. We apply SLOG to a vision encoder-decoder model with a simple transformer (Vaswani et al., 2017) model. Given an image $\mathbf{x}$, the VLM outputs a textual caption describing what objects it contains. We trained the model with an nVidia A100 40 GB GPU with a batch size of 256, restricting the maximum length to 55. We kept a patience of 25 and retained the model achieving the best BLEU_4 score. The BLEU_k score is useful to evaluate the quality of machine-generated text with respect to a reference text, usually generated by a human. Specifically, BLEU_k considers the overlap of each k-gram between the machine-generated and the reference text (Papineni et al., 2002).

Simulating the human expert. To retain full control over the experimental setting, we *simulate* the human expert using a machine learning classifier, denoted HUMANPROXY, which takes as input an image $\mathbf{x}$ and the corresponding guidance g and uses them to infer the six binary labels. The surrogate HUMANPROXY is a multi-modal model with a RESNET101 model as visual extractor (He et al., 2016) and a transformer module as textual encoder. The sole purpose of this surrogate is to produce (possibly imperfect) decisions for all six rules for the entire data set, which are not provided by the original CLEVR dataset but are needed for training the SLOG surrogate model σ_{quality}.

Specifically, we obtain the prediction using a k-fold cross validation procedure: in each fold, we fit HUMAN-PROXY using the ground-truth labels (rule activations) from the training split, and produce predictions for all examples in the test split. Then we aggregate all test prediction to annotate the whole datasets. This step ensures there is no data leakage between train and test splits.

Quality surrogate model. The surrogate σ_{quality} takes the same inputs as HUMANPROXY, but estimates the quality of the predictions made by HUMANPROXY. To this end, we compare said predictions with the

Table 1: SLOG **boosts estimated quality of generated guidance without compromising text quality in** CLEVR. The results show that SLOG substantially improves estimated guidance quality as measured by the surrogate model (σ_{quality}) without affecting text quality as measured by BLEU scores over ground-truth caption data.

MODEL	BLEU$_1$	BLEU$_2$	BLEU$_3$	BLEU$_4$	BLEURT	σ_{quality}
Baseline	0.95	0.92	0.89	0.85	0.73	1.96
Fine-tuned	0.94	0.92	0.89	0.85	0.72	1.95
SLOG	**0.98**	**0.96**	**0.93**	**0.88**	**0.76**	**2.28**

Table 2: SLOG **boosts the quality on the downstream decision in** CLEVR, as shown by the precision, recall and F_1 performance of decisions entailed by SLOG's guidance compared to that of decisions entailed by the baseline VLM and a VLM fine tuned without the SLOG loss. Best F_1 results in **bold**.

	Baseline			Baseline (Fine-tuned)			SLOG		
Rule	Pr	Rc	F_1	Pr	Rc	F_1	Pr	Rc	F_1
Rule 1	92.33	95.91	94.09	93.55	94.03	93.79	94.52	97.71	**96.09**
Rule 2	99.98	97.47	98.71	100.00	95.84	97.88	99.92	99.05	**99.49**
Rule 3	99.75	84.93	91.75	99.75	84.89	91.72	99.77	90.09	**94.68**
Rule 4	94.44	96.67	95.54	96.74	94.63	95.67	96.97	98.33	**97.65**
Rule 5	95.47	93.38	94.41	98.05	92.25	95.06	97.91	96.25	**97.07**
Rule 6	86.59	98.26	92.05	93.88	94.70	94.28	93.36	99.45	**96.31**
MACRO AVG	94.76	94.44	94.42	97.00	92.72	94.73	97.07	96.82	**96.88**
MICRO AVG	94.42	95.14	94.78	96.84	93.32	95.05	96.94	97.31	**97.12**

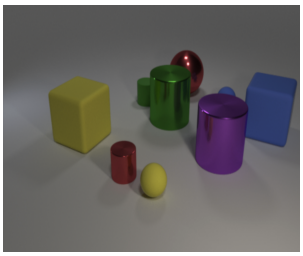
ground-truth rule activations and record whether they match or not. Then we train σ_{quality} to predict the *correctness* of predictions given only the image $\mathbf{x}$ and guidance g. We employed 70%-20%-10% splits for training, validation and test, respectively. We trained σ_{quality} with a cross entropy loss to predict the accuracy of the predictions, and selected the model that attained the best micro validation F_1. We set patience to 20 epochs, for a total of 75 training epochs.

Applying SLOG. Given the trained surrogate σ_{quality}, we ran the SLOG finetuning for 10 epochs. In this phase, we froze both σ_{quality} and the visual encoder of the VLM. We evaluated several values of the hyperparameter λ (cf. Eq. (2)) and chose $\lambda = 1$ as it yielded the highest F_1 score on a validation split.

Competitors and metrics. We assess the performance of SLOG both quantitatively and qualitatively, and compare it against two competitors. These include the original pretrained VLM (denoted "baseline") as well as the same VLM fine-tuned for the same number of epochs as the SLOG variant, but with $\lambda = 0$, so as to disable the SLOG loss term (denoted "fine-tuned").

We evaluate both the quality of down-stream decisions taken using the generated guidance and the textual guidance itself. For the former, we report the test set precision (Pr), recall (Rc), and F_1 score of the decisions, both per-label and (micro- and macro-) averaged over all six labels. We obtain the down-stream decision by applying the rules to the generated guidance, using a simple NLP pipeline that scans textual guidance for presence of individual objects and their properties (e.g., "a large red sphere") and checks which rules fire based on the patterns it matched. We compare the resulting decisions against the ground-truth labels. For the latter, we report the $BLEU_k$ score for $k = 1, \ldots, 4$ with respect to the ground-truth captions, the average guidance length, and the average estimated quality output by σ_{quality} as a sanity check.

Q1: SLOG **improves the quality of downstream decisions in our controlled setting**. Our results are summarized in Tables 1 and 2. Table 1 shows that SLOG's guidance yields a distinct improvement in terms of σ_{quality} score (second to last column), as expected. This highlights how the SLOG fine-tuning procedure succeeds in optimizing the SLOG loss (Eq. (2)). The remaining results suggest that doing so yields better



GT.1 small blue rubber sphere 1 large purple metal cylinder 1 small red metal cylinder 1 large yellow rubber cube 1 small green rubber cylinder 1 large green metal cylinder 1 large blue rubber cube 1 large red metal sphere 1 small yellow rubber sphere .
Base.1 small blue rubber sphere 1 small red metal cylinder 1 large purple metal cylinder 1 small yellow rubber sphere 1 large blue rubber cube 1 large yellow rubber cube 1 small green rubber cylinder 1 large green metal cylinder .
FT.1 small red metal cylinder 1 small blue rubber sphere 1 large red metal sphere 1 large yellow rubber cube 1 small yellow rubber sphere 1 large purple metal cylinder 1 large green metal cylinder 1 small green rubber cylinder .
SLOG.1 small red metal cylinder 1 small blue rubber sphere **1 large red metal sphere** 1 large green metal cylinder 1 large yellow rubber cube 1 small yellow rubber sphere 1 large purple metal cylinder 1 small green rubber cylinder **1 large blue rubber cube**.

Figure 4: **Example guidance generated by different competitors on** CLEVR**.** While SLOG neither misses nor hallucinates any description, each of the other two competitors, despite not hallucinating, misses one description. The missed descriptions are presented in green bold fonts in the above text.

Table 3: Comparison between the original and our filtered splits for the `Mimic-CXR-IV` dataset.

	ORIGINAL SPLIT			OUR SPLIT		
COMPONENTS	TRAIN	VAL	TEST	TRAIN	VAL	TEST
REPORTS	222,758	1,808	3,269	125,417	991	1,624
IMAGES	368,960	2,991	5,159	232,855	1,837	2,872

guidance. In fact, SLOG also out-performs both competitors in terms of textual quality: it achieves between +3 and +4 % in terms of BLEU scores and the BLEURT[2] (Sellam et al., 2020) metric (first five columns) compared to both the pretrained model and its fine-tuned variant. Simultaneously, Table 2 indicates that the guidance produced with SLOG facilitates inferring correct labels compared to the captions produced by the baseline models. This holds for each rule/label individually (first six rows), and on average (last two rows). This provides initial evidence that SLOG also improves the quality of down-stream decisions in a controlled setting.

To further illustrate the benefits of SLOG, we report in Figure 4 an example of the guidance it produces, compared to the captions output by the competitors. The example shows how SLOG's guidance describes objects that are relevant for the decision correctly that the competitors neglect.

4.2 Q2–Q4: The `Mimic-CXR-IV` Task

Data set. Next, we evaluate SLOG on the `Mimic-CXR-IV` data set (Johnson et al., 2019), one of the largest publicly available medical decision data sets, consisting of 227,835 radiology reports and 377,110 chest X-ray scans. We focus on the *findings* and the *impression* sections of the reports. As shown in Fig. 3 (right) the findings are text-based descriptions of what can be observed in the scan, and constitute the basis on top of which the expert forms their impression, i.e., their initial opinion about the potential pathologies of a patient. We discarded examples where either findings or impressions were not available, resulting in the *training*, *validation* and *test* splits presented in Table 3.

Vision-language Models. We apply SLOG to two vision-language models: `R2Gen` (Chen et al., 2020) and `R2GenCMN` (Chen et al., 2022). The former is a memory-driven transformer (Vaswani et al., 2017) specifically designed for pathological report generation from chest X-ray images, while the latter uses a cross modal network (CMN) in order to achieve better mapping between diverse modalities. The `R2Gen` architecture builds on the observation that similar radiographs may correspond to reports sharing similar patterns. To exploit this, it employs a pre-trained CNN model to extract patch features and encodes these into hidden states with an encoder. A decoder then maps the hidden states into words at each time point with the help of a relational memory and memory-driven conditional layer normalization. The relational memory component allows the transformer to store and repurpose shared patterns and thus generate more coherent reports

[2]While the BLEU scores measure the k-gram based overlap between the predicted and generated texts, the BLEURT is a BERT (Devlin et al., 2019) based regression model trained on human-ratings data.

(Chen et al., 2020). Chen et al. (2022) argues that the existing literature offers only limited scope for proper alignment across modalities. Addressing this issue, the authors developed `R2GenCMN`, a cross modal network where the encoded features of an image is fed to the CMN module to obtain the memory representations. A similar operation is done for the text embeddings. Thus, the shared information of the text and visual features can be stored in the memory. In particular, the CMN module employs a matrix where each row of the matrix is allotted for cross-modal memory information for image and texts.

Decision-making task. Our real-world experiment focuses on a critical step of the medical decision process: making the right diagnosis.The task is to diagnose 14 different pathologies (see Table 6 for the full list) from X-ray images. In our experiments, we pre-train the `R2Gen` and `R2GenCMN` VLMs to predict *findings* from images, and then fine-tune them with SLOG to improve their generated guidance. Given that the *impression* is the opinion that the expert forms about potential pathologies visible in the image, *we fine-tune our VLMs to produce textual guidance that – once interpreted by a human expert – leads to the same diagnosis entailed by the impression.*

Simulating the human expert. As in the case of CLEVR task, we simulate human decisions using a machine learning model denoted HUMANPROXY, for reproducibility. Specifically, HUMANPROXY takes a scan $\mathbf{x}$ and a corresponding VLM-generated report and diagnoses the 14 candidate pathologies using three classes: definitely present (*positive*), definitely absent (*negative*), and unclear (*ambiguous*). Following (Lovelace & Mortazavi, 2020a), we implement HUMANPROXY as a classification model that takes both reports and images as inputs and train it on ground-truth labels obtained by applying the `CheXpert` (Irvin et al., 2019) automated annotation tool to the ground-truth *impressions*. Please note that while the σ_{quality} surrogate is an integral part of SLOG, HUMANPROXY is an experimental detail necessary for evaluation.

In order to emulate a setting with sparse human supervision, we assume *ground-truth labels are available for 10% of the training data only*. This ground-truth dataset $\mathcal{D}_{\text{surr}}$ is used for training both the model simulating human decisions HUMANPROXY and the quality surrogate model σ_{quality} estimating the quality of these decisions. We rely on a stratified sampling procedure to select $\mathcal{D}_{\text{surr}}$ so as to maintain a reasonable coverage of the different classes.

Overall, we proceed as follows. First, HUMANPROXY is trained on $\mathcal{D}_{\text{surr}}$ to output a diagnosis given ground-truth findings (as these are the only ones for which we know the corresponding human diagnosis). Once trained, we use HUMANPROXY to produce quality ratings by computing the correctness of its predictions over $\mathcal{D}_{\text{surr}}$, which will later on be used for training the surrogate σ_{quality}.

To avoid biasing the quality rating supervision by computing it on training instances, we run k-fold cross validation on $\mathcal{D}_{\text{surr}}$ and collect quality ratings from the k validation folds. For each validation fold, we compute decisions using both VLM-generated guidance and ground-truth text, so as to provide examples of both predicted and ground-truth guidance to train the quality surrogate model.

Quality surrogate model. As explained in Section 3.1, the surrogate σ_{quality} should estimate the quality of human decisions when fed with the VLM guidance. In this experiment, human decisions are proxied with the 14 labels output by HUMANPROXY. The surrogate is thus trained to predict the *correctness* of each of the 14 predictions made by HUMANPROXY. Just like HUMANPROXY, the surrogate σ_{quality} used by SLOG is also implemented as mutli-modal architecture and trained to minimize the average cross entropy loss on the ground-truth dataset described in the previous paragraph. Albeit having different purposes, HUMANPROXY and σ_{quality} share the same model architecture. The multimodal functionality of both the models are established with a visual-encoder module and a text encoder module, where the former is a ResNet 101 based model that extracts the features of the radiology images and the latter is a transformer based module that extracts nuanced features of the reports. To this end, we concatenate the features obtained from the text extractor and image extractor before applying a fully connected layer onto the concatenation.

Overall pipeline. First, an `R2Gen` VLM is pre-trained on the training split $\mathcal{D}_{\text{train}}$ to generate findings. The VLM is then applied to the decision ground-truth dataset $\mathcal{D}_{\text{surr}}$ (10% of the training split). The generated guidance is fed to HUMANPROXY to obtain (simulated) human decisions and corresponding quality annotations. This information is then used to fit the quality surrogate model σ_{quality}. Finally, the VLM is

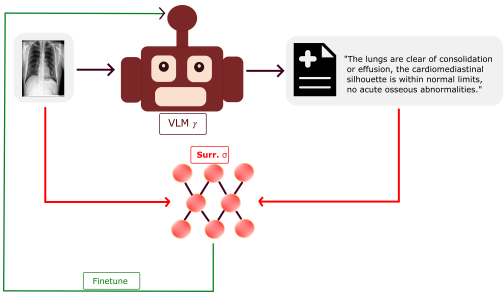


Figure 5: Finetuning policy of SLOG.

fine-tuned *on the entire training split* according to Eq. (2) for 10 epochs and evaluated on the test data. The source code is provided in the supplementary material. Fig. 5 depicts a summary of the overall pipeline.

Training the VLM. We trained our baseline VLM with an Nvidia A100 80GB GPU and with batch size 256. We restricted the maximum sequence length to 70 in order to avoid computational over head in later stages of our experiment. We fine-tuned the baseline models (`R2Gen` and `R2GenCMN`) on the same GPU with a batch size of 64. The values of all hyper-parameters were taken verbatim from (Chen et al., 2020) and (Chen et al., 2021), respectively. While training the baseline model, we used a patience of 20 and stored the best model with the highest BLEU-4 score.

Training the surrogate. In order to emulate a setting with sparse human supervision, we assume *ground-truth labels are available for 10% of the training data only*. This ground-truth dataset $\mathcal{D}_{\text{surr}}$ is used for training both the model simulating human decisions HUMANPROXY *and* the quality surrogate σ_{quality} estimating the quality of these decisions. HUMANPROXY, which acts as a proxy for the human annotator, is a multimodal classification model which takes the radiology image and corresponding report as inputs and predicts the 14 target symptoms (cf. Table 6). To this end, in addition to calculating the loss and classification metrics, we conducted a sample-wise comparison between the ground truth labels and the prediction with the purpose of generating training data for σ_{quality}. This comparison yielded an $m \times n$ matrix Y_q where $m = 14$ and n is the number of training examples. Let us consider G is the ground truth matrix of labels used for HUMANPROXY and P is the matrix of labels predicted by the model on the validation data. We define,

$$Y_q = y_{ij}, \quad \text{where } y_{ij} = \begin{cases} 1 & \text{if } G_{ij} = P_{ij} \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

In order to train HUMANPROXY, we use k-fold cross validation with $k = 5$. We assessed the performance of our model on the validation set by scrutinizing the micro F_1 score pertaining to the positive mentions.

The σ_{quality} takes same input as the HUMANPROXY, but instead of three labels, it outputs either 1 or 0 for the 14 classes (see Equation 3). In Table 4, we report the results obtained from the test split that was used to evaluate the performance of σ_{quality}. Results clearly indicate that σ_{quality} is capable of reliably predicting the correctness of HUMANPROXY when provided image and guidance.

Applying SLOG. We apply SLOG finetuning for 10 epochs. In this phase, we freeze both the σ_{quality} and the visual encoder layer of the baseline `R2Gen` model. We try with varying values of λ and the best model was chosen based on the validation F_1 score. Eventually, for `R2Gen`, we choose 10 as the value of hyperparameter λ as $\lambda = 10$ yielded the best F_1 during finetuning. Along with finetuning using $\lambda = 10$, we also experiment with $\lambda = 0$ to finetune without the σ_{quality}. In both cases, we finetune the baseline model for equal number of epochs. We follow the same pipeline for `R2GenCMN` and chose 0.01 as the value for hyperparameter λ.

Q2: SLOG improves informativeness on the test set without compromising BLEU score. A potential issue with using a surrogate model as a proxy of decision quality is that the fine-tuned VLM might end up overfitting the surrogate and produce guidance that, while seemingly informative, is unrelated to the actual input scan. SLOG prevents this by complementing estimated guidance quality as computed by the surrogate model with guidance appropriateness for the input image as measured by cross entropy over a training set of findings. Table 5 confirms the effectiveness of this strategy. SLOG substantially improves

Table 4: **Outcomes from the test split used to evaluate σ_{quality}.** Results of R2Gen and R2GenCMN showing per-class, macro, and micro averaged precision (Pr), recall (Rc), and F_1 scores.

PATHOLOGY	R2Gen			R2GenCMN		
	Pr	Rc	F_1	Pr	Rc	F_1
No Findings	54.88	87.53	67.46	86.22	79.69	82.82
Cardiomediastinum	84.68	93.47	88.85	92.65	93.24	92.94
Cardiomegali	90.24	94.11	92.13	94.25	94.65	94.45
Lung Lesion	78.28	93.07	85.03	91.93	91.89	91.91
Lung Opacity	92.07	94.21	93.13	93.61	95.48	94.54
Edema	92.52	93.87	93.19	94.99	96.82	95.90
Consolidation	91.99	92.71	92.35	93.12	94.65	93.88
Pneumonia	42.08	84.86	56.26	74.99	75.51	75.25
Atelectasis	82.87	94.67	88.38	93.58	92.36	92.97
Pneumothorax	60.89	91.55	73.13	87.55	85.46	86.49
Pleural Effusion	96.55	97.12	96.84	96.73	98.38	97.55
Pleural Other	62.86	87.97	73.33	84.00	81.90	82.94
Fracture	93.94	93.86	93.90	93.62	94.36	93.99
Support Devices	78.03	91.75	84.34	91.46	88.19	89.80
MACRO	92.96	80.84	86.19	90.62	90.18	90.39
MICRO	92.20	78.71	84.17	91.18	90.84	91.01

Table 5: SLOG **boosts estimated quality of generated guidance without compromising text quality**. The results show that SLOG substantially improves estimated guidance quality as measured by the surrogate model (σ_{quality}) without affecting text quality as measured by BLEU scores over ground-truth caption data.

MODEL	SETTING	BLEU$_1$	BLEU$_2$	BLEU$_3$	BLEU$_4$	BLEURT	σ_{quality}
R2Gen	Pretrained	**0.36**	**0.22**	**0.15**	**0.11**	**-0.38**	0.39
	Fine-tuned	0.33	0.21	0.14	0.10	-0.40	0.39
	SLOG	0.35	**0.22**	**0.15**	**0.11**	**-0.38**	**0.44**
R2GenCMN	Pretrained	**0.38**	**0.23**	**0.16**	**0.11**	-0.36	1.84
	Fine-tuned	0.37	0.22	0.15	**0.11**	-0.35	1.85
	SLOG	**0.38**	**0.23**	**0.16**	**0.11**	**-0.34**	**2.02**

estimated guidance quality (second term in Eq. (2)) without compromising text quality, as measured both in terms of BLEU and BLEURT scores over test examples. For the sake of fairness, we compared SLOG (with $\lambda = 10$) with both the pre-trained R2Gen model (before the fine-tuning stage), and the R2Gen model fine-tuned for the same number of epochs as SLOG, but with caption-level supervision only (i.e., setting $\lambda = 0$ in Eq. (2)), as well as with two R2GenCMN models fine-tuned in the same way. Fig. 6 shows a qualitative example of the improvement in guidance of SLOG with respect to the competitors. First, SLOG's guidance retains all pieces of text that any of the other approach shares with the ground-truth text (green text). On top of this, SLOG retrieves additional chunks of text that are shared with ground truth findings (blue text) and impression (magenta text), even if the latter are never explicitly included as training supervision, confirming the effectiveness of the quality surrogate in encouraging the generation of relevant guidance for the diagnosis.[3]

Q3: SLOG improves quality of decisions. Tables 6 and 7 show the results in terms of decision quality, as measured by the F_1 score of the positive label for all 14 classes (multi-label prediction). Results clearly

[3]Notice that while the SLOG guidance text is longer than the one of the competitors in this example, with various chunks of text which are not obviously connected to ground-truth ones, its overall quality is still much higher.

Table 6: SLOG **boosts quality of downstream decisions for R2Gen.** Results show per-class, macro and micro averaged precision, recall and F_1. Best F_1 results are boldfaced.

	R2Gen (pretrained)			R2Gen (fine-tuned)			SLOG		
PATHOLOGY	Pr	Rc	F_1	Pr	Rc	F_1	Pr	Rc	F_1
No Findings	34.42	59.73	43.67	33.08	59.95	42.64	38.09	54.98	**45**
Cardiomediastinum	0	0	0	1.92	5.56	**2.86**	0	0	0
Cardiomegaly	14.06	28.72	18.88	14.29	23.94	17.89	15.7	37.23	**22.08**
Lung Lesion	0	0	0	0	0	0	0	0	0
Lung Opacity	26.32	11	15.52	29.63	9.78	14.71	30.52	18.58	**23.1**
Edema	42.03	18.65	**25.84**	40	16.08	22.94	43.48	16.08	23.47
Consolidation	0	0	0	3.23	1.41	1.96	9.43	7.04	**8.06**
Pneumonia	0	0	0	0	0	0	0	0	0
Atelectasis	17.65	22.57	19.81	17.5	18.58	18.03	19.88	28.76	**23.51**
Pneumothorax	0	0	0	8.33	3.33	4.76	13.64	10	**11.54**
Pleural Effusion	48.82	28.3	**35.83**	44.39	23.9	31.07	45.02	28.57	34.96
Pleural Other	0	0	0	0	0	0	0	0	0
Fracture	0	0	0	0	0	0	0	0	0
Support Devices	17.64	34.66	23.64	19.14	35.23	24.8	18.93	42.05	**26.1**
MACRO	14.37	14.55	13.08	15.11	14.13	12.98	16.76	17.38	**15.56**
MICRO	26.72	25.41	26.05	26.57	23.73	25.07	27.19	27.57	**27.38**

Table 7: Performance of R2GenCMN finetuned with SLOG. Results show per-class, macro and micro averaged precision, recall and F_1. Best F_1 results are boldfaced.

	R2GenCMN			R2GenCMN (fine-tuned)			SLOG		
PATHOLOGY	Pr	Rc	F_1	Pr	Rc	F_1	Pr	Rc	F_1
No Findings	33.93	43.21	38.01	36.03	48.42	**41.31**	36.11	47.06	40.86
Cardiomediastinum	1.92	5.56	**2.86**	0.0	0.0	0.0	1.27	5.56	2.06
Cardiomegaly	16.33	55.85	25.27	17.65	57.45	**27.00**	17.2	51.6	25.8
Lung Lesion	0.0	0.0	0.0	25.0	1.67	3.12	33.33	1.67	**3.17**
Lung Opacity	22.96	21.27	22.08	25.29	15.89	19.52	28.41	14.91	**26.28**
Edema	35.0	6.75	11.32	35.71	12.86	18.91	43.55	17.36	**24.83**
Consolidation	10.0	9.86	9.93	7.29	9.86	8.38	10.11	12.68	**11.25**
Pneumonia	28.57	1.31	2.5	19.23	3.27	**5.59**	14.29	1.96	3.45
Atelectasis	19.9	34.51	**25.24**	19.94	28.32	23.4	19.58	32.74	24.5
Pneumothorax	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pleural Effusion	50.29	24.18	32.65	44.39	25.0	31.99	48.55	29.95	**37.01**
Pleural Other	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fracture	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Support Devices	17.95	48.86	**26.26**	18.18	43.18	25.59	17.61	44.32	25.2
MACRO	16.92	17.95	14.01	17.77	17.57	14.63	19.28	19.24	**16.03**
MICRO	23.44	26.61	24.93	24.64	26.81	25.68	25.56	29.32	**27.31**

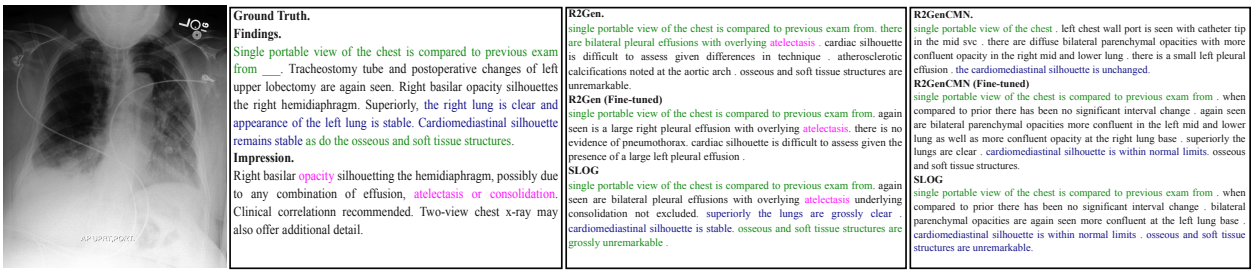


Figure 6: A qualitative example of the improvement of the guidance from SLOG with respect to the competitors. Green text indicates sentences that are (approximately) shared between the ground truth, SLOG and at least one of the competitors. Blue text indicates sentences shared between the ground truth findings (resp. impression) and SLOG, but missed by the competitors. No ground-truth sentences are shared between ground-truth text and competitors but missed by SLOG in this example.

indicate the effectiveness of the SLOG guidance in improving decision quality, despite the modest amount of supervision it received. On `R2Gen`, SLOG outperforms the competitors in 7 out of 14 classes. All methods fail to identify any positive occurrence for three particularly under-represented classes (`Pneumonia`, `Pleural Other`, `Fracture`), while SLOG slightly underperforms with respect to pre-trained `R2Gen` in 2 classes only. It is worth noticing that SLOG is especially effective in improving recall without affecting precision on average, as shown by the two bottom lines reporting results averaged over classes (macro) and instances (micro) respectively. On `R2GenCMN`, SLOG is the best performing method on 5 classes, while the two baselines win on 3 classes each. Additionally, when SLOG is the winner, it does so by a sensible margin, with about 5% improvements in F_1 with respect `R2GenCMN` and 4% improvements with respect to `R2GenCMN` (fine-tuned). Conversely, when the baselines outperform SLOG they do so by a rather small margin (0.9% and 1.3% for `R2GenCMN` and `R2GenCMN` (fine-tuned) respectively. In short, SLOG applied to both `R2Gen` and `R2GenCMN` showcases an overall improvement in both micro and macro F_1.

Q4: SLOG has the potential to help doctors make better decisions. The guidance quality of SLOG was evaluated against a fine-tuned `R2GenCMN` via expert review by a pulmonologist with three years of clinical experience. In this setup, the clinician assessed 25 guidances from SLOG and fine-tuned `R2GenCMN` (total 50). We decided to focus on fine-tuned `R2GenCMN`, the runner-up according to the previous experiment, to avoid overloading the clinician with too many assessments. It is important to examine whether SLOG helps the physician identify the presence of a symptom. Therefore, to ensure representative coverage while maintaining balanced class counts, we implemented a sampling technique that guaranties the presence of a symptom with at least one positive mention in the sample set. Thus, the samples cover all classes with at least one positive mention. The actual algorithm has been provided in the Appendix A. Table 8 shows the results in terms of decision quality, measured by the F_1 score of the positive label for the 14 classes (prediction with multiple labels). The results confirm the advantage of SLOG in improving the overall performance of the decision-making process, both in terms of micro and macro F_1.

Ablation experiment. To investigate the influence of σ_{quality} in the finetuning process, we conduct an ablation study based on the percentage of training data used to train HUMANPROXY and σ_{quality}. In our primary experiments, we take 10% of training examples to train the surrogate models. However, for the ablation, we finetune the baseline model σ_{quality} with only 1% and 5% of the training examples. The results can be viewed in Fig. 7. While the surrogates trained with 10% of the data understandably yield the best F_1, an improvement can still be observed when reducing the amount of training data further. In particular, macro F_1 (right) shows that even with 1% surrogate training data, SLOG still manages to outperform both baselines (dashed lines), while micro F_1 (left) is on-par with them. This indicates that even the less informed surrogates can help improving prediction quality for the rarer – but possibly significant for decision making – classes. In Table 9, we notice that SLOG trained with 10% of training examples outperforms the former two in both micro and macro metrics. Detailed results can be obtained in Appendix A.

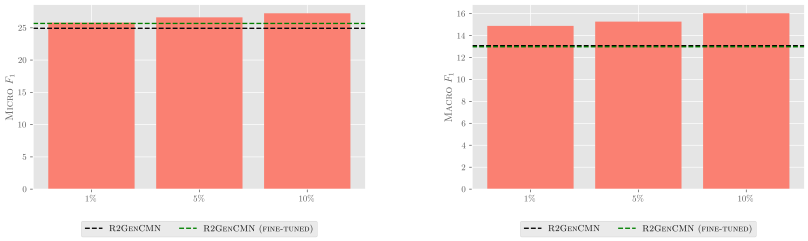
Table 8: Performance comparison between baseline (fine-tuned) and SLOG when a subset of samples was presented to a doctor. Results show per-class, macro and micro averaged precision, recall and F_1. Best F_1 results between baseline and SLOG are boldfaced.

	Baseline (fine-tuned)			SLOG		
PATHOLOGY	*Pr*	*Rc*	F_1	*Pr*	*Rc*	F_1
No Findings	11.11	33.33	**16.67**	0.00	0.00	0.00
Enlarged Cardiomediastinum	0.00	0.00	0.00	0.00	0.00	0.00
Cardiomegaly	11.11	33.33	16.67	25.00	50.00	**33.33**
Lung Lesion	0.00	0.00	0.00	16.67	33.33	**22.22**
Lung Opacity	0.00	0.00	0.00	0.00	0.00	0.00
Edema	28.57	50.00	**36.36**	14.29	20.00	16.67
Consolidation	0.00	0.00	0.00	0.00	0.00	0.00
Pneumonia	0.00	0.00	0.00	0.00	0.00	0.00
Atelectasis	9.09	33.33	14.29	31.25	100.00	**47.62**
Pneumothorax	0.00	0.00	0.00	0.00	0.00	0.00
Pleural Effusion	57.14	57.14	**57.14**	55.56	50.00	52.63
Pleural Other	0.00	0.00	0.00	0.00	0.00	0.00
Fracture	0.00	0.00	0.00	0.00	0.00	0.00
Support Devices	22.22	100.00	**36.36**	15.38	66.67	25.00
MACRO	9.95	21.94	12.68	11.30	22.86	**14.11**
MICRO	15.94	28.95	20.56	20.48	29.82	**24.29**

5 Related Work

Aligning LLMs. The standard approach for aligning large language models to human interests is reinforcement learning with human feedback (RLHF) (Ziegler et al., 2020; Ouyang et al., 2022). Several RLHF-based approaches that target medical tasks exist. Yunxiang et al. (2023) and Wang et al. (2023a) proposed medical chat models obtained by fine-tuning existing architectures, while Bazi et al. (2023) introduced a specially designed vision transformer. Seo et al. (2020) presented a method for improving the performance of an image caption generator with offline human feedback. SLOG can be viewed as a variant of RLHF that foregoes reinforcement learning in favor of a fully end-to-end fine-tuning strategy, for efficiency. It also differs in aim, in that it optimizes the model's guidance for *a specific human decision making task*, rather than for factuality and fairness in general (Ouyang et al., 2022).

Pathological report generation. Several approaches (Hou et al., 2021; Chen et al., 2020; Wang et al., 2022; 2023b) have been developed for machine-driven pathological report generation from chest X-ray images



(a) Micro F_1 on the test data (b) Macro F_1 on the test data

Figure 7: Performance of the finetuned models involving σ_{quality} trained with varying percentage of train examples. The black and green lines are the F_1 scores obtained with the pretrained and finetuned models ($\lambda = 0$).

Table 9: Ablation study on dataset size for SLOG. Performance improves consistently as training data increases from 1% to 10%.

Type	Metric	SLOG 1%	SLOG 5%	SLOG 10%
Macro	Precision	19.11	15.65	19.28
	Recall	17.58	18.38	19.24
	F_1	14.88	15.27	**16.03**
Micro	Precision	24.40	24.93	25.56
	Recall	27.57	28.57	29.32
	F_1	25.89	26.62	**27.41**

using the `Mimic-CXR-IV` (Johnson et al., 2019) and the Indiana University chest X-ray data sets (Demner-Fushman et al., 2016). (Lovelace & Mortazavi, 2020b) designed a model with a similar objective but they proposed to leverage the `CheXpert` dataset to enhance the coherence of their model. (Tanida et al., 2023) introduced a region-guided model to generate pathological reports, thus opening the window of interactive human-guided report generation. (Srivastav et al., 2024) used a large language model based on Vicuna-7B to generate radiology reports out of CXR images. A slightly different approach was used by (Woźnicki et al., 2024) where the authors used a large language model to extract the structured information out of the *findings* section of a report. However, these models are not concerned with optimizing the utility of the generated reports for the follow-up decision making. Our approach builds on top of these methods, enriching them with the ability to incorporate surrogate quality information (we use (Chen et al., 2020) in our evaluation, but any of these models can be adapted for SLOG).

Other approaches. LTG is related to *explain then predict* (ETP) (Camburu et al., 2018; Kumar & Talukdar, 2020; Zhao & Vydiswaran, 2021), a framework for building explainable (Guidotti et al., 2018) models in which a machine first outputs a full-fledged explanation – playing the role of "guidance" – and then derives a prediction from the explanation itself. In LTG, however, the prediction step is carried out by a human expert, and as such it is not differentiable. Also, ETP requires direct supervision on the explanations themselves, which is seldom available. In contrast, SLOG improves guidance quality using indirect scoring feedback, which is comparatively easier to acquire.

Finally, LTG is not restricted to textual guidance. One option is, for instance, to implement guidance in terms of explanations extracted from (or output by) an underlying image classifier to guide human decision making (Guidotti et al., 2018). From this perspective, LTG is tied to explanatory interactive learning (XIL) (Schramowski et al., 2020; Teso et al., 2023), in which the goal is to improve the quality of explanations output by a machine learning model by interactively acquiring corrections to the explanations themselves. The key difference is that LTG focuses on down-stream decision quality and SLOG supports textual guidance, while XIL aims at more generally improving explanation quality and implementations do no support textual explanations.

6 Conclusion

We introduced *learning to guide* as an alternative setup for high-stakes hybrid decision making that ensures the human expert is always in the loop, as well as SLOG, an end-to-end approach for turning pre-trained VLMs into high-quality textual guidance using human feedback. Our results suggest that SLOG is effective at steering VLMs towards generating more informative guidance, leading to improved accuracy in downstream decisions.

In follow-up work, we plan to extend SLOG by integrating ideas from active learning (Settles, 2012) to acquire the quality rankings, and to explore connections with explainable AI (Guidotti et al., 2018), explanatory interactive learning (Schramowski et al., 2020; Teso et al., 2023) and skeptical learning (Zeni et al., 2019) to facilitate the identification and correction of potential issues with the generated guidance.

References

Gagan Bansal, Besmira Nushi, Ece Kamar, Eric Horvitz, and Daniel S Weld. Is the most accurate ai the best teammate? optimizing ai for teamwork. In *AAAI*, 2021.

Yakoub Bazi et al. Visionlanguage model for visual question answering in medical imagery. *Bioengineering*, 2023.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-SNLI: Natural language inference with natural language explanations. *NeurIPS*, 31, 2018.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. Generating radiology reports via memory-driven transformer. In *EMNLP*, 2020.

Zhihong Chen, Yaling Shen, Yan Song, and Xiang Wan. Cross-modal memory networks for radiology report generation. In *ACL-IJCNLP*, 2021.

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

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *Algorithmic Learning Theory*, 2016.

Mary Cummings. Automation bias in intelligent time critical decision support systems. In *Collection of Technical Papers - AIAA 1st Intelligent Systems Technical Conference*, 2012.

Abir De, Paramita Koley, Niloy Ganguly, and Manuel Gomez-Rodriguez. Regression under human assistance. In *AAAI*, 2020.

Abir De et al. Classification under human assistance. In *AAAI*, 2021.

Dina Demner-Fushman et al. Preparing a collection of radiology examinations for distribution and retrieval. *J Am Med Inform Assoc*, 2016.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pp. 4171–4186, 2019.

Kate Donahue, Alexandra Chouldechova, and Krishnaram Kenthapadi. Human-algorithm collaboration: Achieving complementarity and avoiding unfairness. In *FAT*, 2022.

Ignat Drozdov, Daniel Forbes, Benjamin Szubert, Mark Hall, Chris Carlin, and David J. Lowe. Supervised and unsupervised language modelling in chest x-ray radiological reports. *PLOS ONE*, 15, 2020.

Eva Eigner and Thorsten Händler. Determinants of llm-assisted decision-making. *arXiv:2402.17385*, 2024.

European Commission. Proposal for a regulation laying down harmonised rules on artificial intelligence (artificial intelligence act). *eur-lex.europa.eu*, 2021.

Ruijiang Gao, Maytal Saar-Tsechansky, Maria De-Arteaga, Ligong Han, Min Kyung Lee, and Matthew Lease. Human-AI collaboration with bandit feedback. In *IJCAI*, 2021.

Government of Canada. Directive on automated decision-making, 2019.

Ben Green. The flaws of policies requiring human oversight of government algorithms. *Computer Law & Security Review*, 2022.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Marek Herde, Denis Huseljic, Bernhard Sick, and Adrian Calma. A survey on cost types, interaction schemes, and annotator performance models in selection algorithms for active learning in classification. *IEEE Access*, 2021.

Benjamin Hou et al. Ratchet: Medical transformer for chest x-ray diagnosis and reporting. In *MICCAI*, 2021.

Jeremy Irvin et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *AAAI*, 2019.

Alistair Johnson et al. MIMIC-CXR-JPG-chest Radiographs with Structured Labels (version 2.0.0). *PhysioNet*, 2019.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Shalmali Joshi et al. Pre-emptive learning-to-defer for sequential medical decision-making under uncertainty. *arXiv:2109.06312*, 2021.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv:2207.05221*, 2022.

Maxime Kayser, Cornelius Emde, Oana-Maria Camburu, Guy Parsons, Bartlomiej Papiez, and Thomas Lukasiewicz. Explaining chest x-ray pathologies in natural language. In *MICCAI*, 2022.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *NeurIPS*, 2017.

Vijay Keswani et al. Towards unbiased and accurate deferral to multiple experts. In *AIES*, 2021.

Vijay Keswani et al. Designing closed human-in-the-loop deferral pipelines. *arXiv:2202.04718*, 2022.

Sawan Kumar and Partha Talukdar. Nile: Natural language inference with faithful natural language explanations. In *ACL*, 2020.

Jessie Liu et al. Incorporating uncertainty in learning to defer algorithms for safe computer-aided diagnosis. *Scientific Reports*, 2022.

Justin Lovelace and Bobak Mortazavi. Learning to generate clinically coherent chest X-ray reports. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1235–1243, Online, November 2020a. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.findings-emnlp.110.

Justin Lovelace and Bobak Mortazavi. Learning to generate clinically coherent chest x-ray reports. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1235–1243, 2020b.

David Madras et al. Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer. *NeurIPS*, 2018.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *ICML*, 2020.

Nastaran Okati et al. Differentiable learning under triage. *NeurIPS*, 2021.

Long Ouyang, , et al. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv:1903.12220*, 2019.

Charvi Rastogi et al. Deciding fast and slow: The role of cognitive biases in ai-assisted decision-making. *Proc. ACM Hum.-Comput. Interact.*, 2022.

Patrick Schramowski, Wolfgang Stammer, Stefano Teso, Anna Brugger, Franziska Herbert, Xiaoting Shao, Hans-Georg Luigs, Anne-Katrin Mahlein, and Kristian Kersting. Making deep neural networks right for the right scientific reasons by interacting with their explanations. *Nature Machine Intelligence*, 2(8): 476–486, 2020.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*, 2020.

Paul Hongsuck Seo et al. Reinforcing an image caption generator using off-line human feedback. In *AAAI*, 2020.

Burr Settles. *Active Learning*. Morgan & Claypool Publishers, 2012.

Fahad Shamshad, Salman Khan, Syed Waqas Zamir, Muhammad Haris Khan, Munawar Hayat, Fahad Shahbaz Khan, and Huazhu Fu. Transformers in medical imaging: A survey. *Medical Image Analysis*, 2023.

Dhruv Sharma, Sanjay Purushotham, and Chandan K Reddy. Medfusenet: An attention-based multimodal deep learning model for visual question answering in the medical domain. *Scientific Reports*, 2021.

Shaury Srivastav, Mercy Ranjit, Fernando Pérez-García, Kenza Bouzid, Shruthi Bannur, Daniel C Castro, Anton Schwaighofer, Harshita Sharma, Maximilian Ilse, Valentina Salvatelli, et al. Maira at rrg24: A specialised large multimodal model for radiology report generation. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pp. 597–602, 2024.

Tim Tanida, Philip Müller, Georgios Kaissis, and Daniel Rueckert. Interactive and explainable region-guided radiology report generation. In *CVPR*, 2023.

Stefano Teso, Öznur Alkan, Wolfang Stammer, and Elizabeth Daly. Leveraging explanations in interactive machine learning: An overview. *Frontiers in Artificial Intelligence*, 2023.

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

Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In *ICML*, 2022.

Jun Wang, Abhir Bhalerao, and Yulan He. Cross-modal prototype driven network for radiology report generation. In *European Conference on Computer Vision*, pp. 563–579. Springer, 2022.

Sheng Wang et al. Chatcad: Interactive computer-aided diagnosis on medical image using large language models. *arXiv:2302.07257*, 2023a.

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

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *TMLR*, 2022.

Bryan Wilder et al. Learning to complement humans. In *IJCAI*, 2021.

Piotr Woźnicki, Caroline Laqua, Ina Fiku, Amar Hekalo, Daniel Truhn, Sandy Engelhardt, Jakob Kather, Sebastian Foersch, Tugba Akinci DAntonoli, Daniel Pinto dos Santos, et al. Automatic structuring of radiology reports with on-premise open-source large language models. *European Radiology*, pp. 1–12, 2024.

Bin Yan and Mingtao Pei. Clinical-BERT: Vision-language pre-training for radiograph diagnosis and reports generation. In *AAAI*, 2022.

Li Yunxiang et al. Chatdoctor: A medical chat model fine-tuned on llama model using medical domain knowledge. *arXiv:2303.14070*, 2023.

Mattia Zeni, Wanyi Zhang, Enrico Bignotti, Andrea Passerini, and Fausto Giunchiglia. Fixing mislabeling by human annotators leveraging conflict resolution and prior knowledge. *IMWUT*, 2019.

Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. Effect of confidence and explanation on accuracy and trust calibration in ai-assisted decision making. In *FAT*, 2020.

Xinyan Zhao and VG Vinod Vydiswaran. Lirex: Augmenting language inference with relevant explanations. In *AAAI*, 2021.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv:1909.08593*, 2020.

A Appendix

Table 10: Ablation study of SLOG based on `R2GenCMN` with different proportions of labeled data (SLOG 1%, 5%, and 10%).

Pathology	SLOG 1%			SLOG 5%			SLOG 10%		
	Pr	Rc	F_1	Pr	Rc	F_1	Pr	Rc	F_1
No Finding	35.21	45.25	39.60	37.07	46.38	41.21	36.11	47.06	40.86
Cardiomediastinum	0.00	0.00	0.00	0.00	0.00	0.00	1.27	5.56	2.06
Cardiomegaly	15.87	49.47	24.03	16.67	51.60	25.19	17.20	51.60	25.80
Lung Lesion	50.00	1.67	3.23	0.00	0.00	0.00	33.33	1.67	3.17
Lung Opacity	26.79	21.03	23.56	26.49	21.76	23.89	28.41	14.91	26.28
Edema	41.98	17.68	24.89	41.61	18.33	25.45	43.55	17.36	24.83
Consolidation	8.43	9.86	9.09	10.75	14.08	12.20	10.11	12.68	11.25
Pneumonia	7.14	0.65	1.20	4.17	0.65	1.13	14.29	1.96	3.45
Atelectasis	17.24	26.55	20.91	19.17	30.53	23.55	19.58	32.74	24.50
Pneumothorax	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Pleural Effusion	47.41	30.22	36.91	46.03	30.22	36.48	48.55	29.95	37.01
Pleural Other	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fracture	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Support Devices	17.38	43.75	24.88	17.15	43.75	24.64	17.61	44.32	25.20
Macro Avg	19.11	17.58	14.88	15.65	18.38	15.27	19.28	19.24	16.03
Micro Avg	24.40	27.57	25.89	24.93	28.57	26.62	25.56	29.32	27.41

Algorithm 1: Sampling with Minimum Class Coverage

Input: Dataset D (rows = reports),
label set $\mathcal{C}$ (e.g., CheXpert classes),
target sample size N (e.g., 30),
minimum positives per class m (e.g., 2)
Output: Sampled subset $S \subseteq D$ of size N

```
S ← ∅ ;                                                    // selected indices/reports
foreach c ∈ C do
    P_c ← {i ∈ D | label(i, c) = 1};
    if |P_c| ≥ m then
        choose m elements uniformly from P_c without replacement and add to S;
    else
        add all of P_c to S
if |S| > N then
    uniformly sub-sample S down to size N; return S;
R ← D \ S ;                                                // remaining pool
while |S| < N and R ≠ ∅ do
    pick x ∈ R uniformly at random; add x to S; remove x from R;
return S
```