A Closer Look at the Intervention Procedure of Concept Bottleneck Models

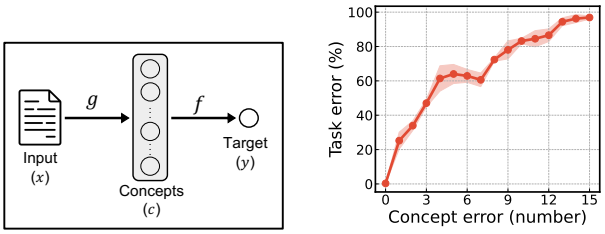
Sungbin Shin [1] **Yohan Jo** [2 *] **Sungsoo Ahn** [1] **Namhoon Lee** [1]

Abstract

Concept bottleneck models (CBMs) are a class of interpretable neural network models that predict the target response of a given input based on its high-level concepts. Unlike the standard end-to-end models, CBMs enable domain experts to intervene on the predicted concepts and rectify any mistakes at test time, so that more accurate task predictions can be made at the end. While such intervenability provides a powerful avenue of control, many aspects of the intervention procedure remain rather unexplored. In this work, we develop various ways of selecting intervening concepts to improve the intervention effectiveness and conduct an array of in-depth analyses as to how they evolve under different circumstances. Specifically, we find that an informed intervention strategy can reduce the task error more than ten times compared to the current baseline under the same amount of intervention counts in realistic settings, and yet, this can vary quite significantly when taking into account different intervention granularity. We verify our findings through comprehensive evaluations, not only on the standard real datasets, but also on synthetic datasets that we generate based on a set of different causal graphs. We further discover some major pitfalls of the current practices which, without a proper addressing, raise concerns on reliability and fairness of the intervention procedure.

1. Introduction

While deep learning has made rapid strides in recent years (LeCun et al., 2015; Jordan & Mitchell, 2015), the standard neural network models are not quite explainable, in that their decision-making process is neither straightforward to account for nor easy to control. To tackle this issue, various interpretable models have been proposed including, for example, those using concept activation vectors (Kim et al., 2018; Ghorbani et al., 2019), relating pixel contributions to image classification (Zhou et al., 2016; Selvaraju et al., 2017), or building intrinsically interpretable architectures (Alvarez Melis & Jaakkola, 2018).



(a) Diagram of CBMs (b) Task vs. Concept errors

Figure 1: (a) Given input data CBMs first predict its concepts ($g : x \rightarrow c$), and then based on which it makes a subsequent prediction for the target response ($f : c \rightarrow y$). (b) Average task error (mis-classification rate) vs. the number of incorrectly predicted concepts (on the CUB dataset). The task error increases rapidly as more mistakes are made in concept prediction; *e.g.*, making a single mistake yields 25% increase in task error.

Concept bottleneck models (CBMs) are among these to empower interpretability (Koh et al., 2020; Bahadori & Heckerman, 2021; Margeloiu et al., 2021; Mahinpei et al., 2021; Sawada & Nakamura, 2022; Zarlenga et al., 2022). Unlike standard end-to-end models, CBMs work in two steps: they first predict human-interpretable properties of a given input called *concepts*, and based on which, they subsequently make the final prediction for the given task. For instance, CBMs may classify the species of a bird based on its wing pattern or leg color rather than straight from the raw pixel values (see Figure 1a).

Revisited recently by Koh et al. (2020), this classic idea further facilitates human-model interaction in addition to plain interpretability, in that it allows one to *intervene* on the predicted concepts at test time, such that the subsequent prediction is made based on the rectified concept values. Notably, such intervention must be treated attentively as we find that correcting only a small number of mistakes on mis-predicted concepts can lead to a significant increase in

*This work is not associated with Amazon. [1]POSTECH, South Korea [2]Amazon – Alexa AI, USA. Correspondence to: Sungbin Shin <ssbin4@postech.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

Work	Selection	Cost	Level	Imp.	Data	Rel.
Koh et al. (2020)	✗	✗	△	△	✗	✗
Chauhan et al. (2022)	✓	△	△	△	✗	✗
Sheth et al. (2022)	✓	✗	△	✗	✗	✗
Ours	✓	✓	✓	✓	✓	✓

Table 1: Comparison between the studies on intervention strategy of CBMs. △ represents that the corresponding work provides only partial evaluations. *Selection* and *Cost* represent concept selection criteria and their analysis in terms of theoretical cost as will be discussed in Section 4.2. We study the effects of *Level, Implementation* and *Data* on intervention effectiveness in Sections 4.3, 4.4 and 5. *Reliability* of intervention practice is discussed in Section 6.

the task performance (see Figure 1b). Considering the high cost of intervention, *i.e.*, having domain experts go over each concept requires tremendous effort, this result further indicates the necessity of efficient intervention procedures to ensure the utility of CBMs.

Despite the great potential, the intervention procedure of CBMs has not been studied much in the literature, quite surprisingly. For example, previous works tend to focus on increasing task performance (Sawada & Nakamura, 2022; Zarlenga et al., 2022) and addressing the problem of confounding factors (Bahadori & Heckerman, 2021) or information leakage (Margeloiu et al., 2021; Mahinpei et al., 2021; Havasi et al., 2022; Marconato et al., 2022). While a few concurrent works suggest new intervention methods (Chauhan et al., 2022; Sheth et al., 2022), we find that many critical aspects of the intervention procedure still remain unexplored (see Table 1).

Our contributions are summarized as follows. First of all, we develop various concept selection criteria as new intervention strategies, improving the intervention performance of CBMs quite dramatically given the same amount of intervention counts. We also provide extensive evaluations to analyze these criteria under a wide variety of experimental settings considering the theoretical cost of each criterion, levels of intervention related to test-time environments, and how to train these models or conceptualize the concept predictions. We further develop a new framework to generate synthetic data using diverse causal graphs and conduct fully controlled experiments to verify the effectiveness of intervention on varying data. These results reveal that data characteristics as well as intervention granularity can affect the intervention procedure quite significantly. Finally, we identify some pitfalls of the current intervention practices, which helps to take a step toward building trustworthy and responsible interpretable models.

2. Related Work

Since the seminal work of Koh et al. (2020), CBMs have evolved in many different ways. Bahadori & Heckerman (2021) develop a debiased CBM to remove the impact of confounding information to secure causality. Sawada & Nakamura (2022) augment CBMs with unsupervised concepts to improve task performance. Mahinpei et al. (2021); Margeloiu et al. (2021) suggest addressing the information leakage problem in CBMs to improve interpretability of learned concepts, while Marconato et al. (2022); Havasi et al. (2022) design new CBMs based on disentangled representations or autoregressive models. Zarlenga et al. (2022) proposes to learn semantically meaningful concepts using concept embedding models to push the accuracy-interpretability trade-off. Both Chauhan et al. (2022) and Sheth et al. (2022) present uncertainty based intervention methods to determine which concepts to intervene on. We remark that previous work is mostly focused on developing CBM variants for high task performance from model-centric perspectives, whereas our work provides in-depth analyses and comprehensive evaluations on the intervention procedure of the standard CBMs in greater granularity.

3. Intervention Strategies

3.1. Preliminary

Let $x \in \mathbb{R}^d$, $c \in \{0,1\}^k$, $y \in \mathcal{Y}$ be input data, binary concepts, and target responses, respectively; here, d and k denote the dimensionality of input data and cardinality of concepts, and we assume $\mathcal{Y}$ encodes categorical distribution for classification tasks. Given some input data (*e.g.*, an image), a CBM first predicts its concepts (*e.g.*, existing attributes in the given image) using a concept predictor g and subsequently target response (*e.g.*, class of the image) using a target predictor f: *i.e.*, first $\hat{c} = g(x)$ then $\hat{y} = f(\hat{c})$, where $\hat{c}$ and $\hat{y}$ are predictions of concepts and target response.

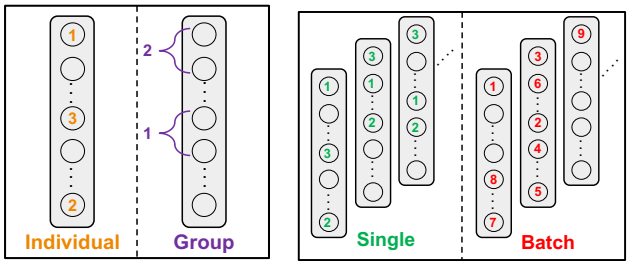
In this process, one can intervene on a set of concepts $\mathcal{S} \subseteq \{1, \cdots, k\}$ so that the final prediction can be made based on rectified concept values, *i.e.*, $\hat{y} = f(\tilde{c})$ where $\tilde{c} = \{\hat{c}_{\setminus\mathcal{S}}, c_{\mathcal{S}}\}$ denotes the updated concept values partly rectified on $\mathcal{S}$ with $\hat{c}_{\setminus\mathcal{S}}$ referring to the predicted concept values excluding $\mathcal{S}$.

3.2. Concept Selection Criteria

How should one select which concepts to intervene on? This is a fundamental question to be answered in order to legitimize CBMs in practice since intervention incurs the cost of employing experts, which would increase as with the number of intervening concepts $|\mathcal{S}|$. In principle, one would select a concept by which it leads to the largest increase in the task performance. To address this question and investigate the effectiveness of intervention procedure in current practice, we develop various concept selection

Criteria	N_g	N_f	Cost in complexity
RAND	1	1	$\mathcal{O}(\tau_g + \tau_f + n\tau_i)$
UCP	1	1	$\mathcal{O}(\tau_g + \tau_f + n\tau_i)$
LCP	1	1	$\mathcal{O}(\tau_g + \tau_f + n\tau_i)$
CCTP	1	3	$\mathcal{O}(\tau_g + 3\tau_f + n\tau_i)$
ECTP	1	$2k+2$	$\mathcal{O}(\tau_g + (2k+2)\tau_f + n\tau_i)$
EUDTP	1	$2k+2$	$\mathcal{O}(\tau_g + (2k+2)\tau_f + n\tau_i)$

Table 2: Theoretical cost of employing concept selection criteria to make final prediction with n number of intervened concepts. N_g and N_f refer to the number of forward/backward passes to run g and f, respectively.



(a) Individual vs. Group (b) Single vs. Batch

Figure 2: Different levels of intervention conducted on concepts. Each number represents the order of intervention.

criteria for which a selection score s_i for i-th concept is defined. Then, intervening concepts will be done based on the decreasing order of these scores.

Random (RAND) It selects concepts uniformly at random as in Koh et al. (2020). We can treat this method as assigning a random score for each concept, *i.e.*, $s_i \sim \mathcal{U}_{[0,1]}$. It will serve as a baseline to study the effectiveness of concept selection criteria.

Uncertainty of concept prediction (UCP) It selects concepts with the highest uncertainty of concept prediction. Specifically, it defines $s_i = \mathcal{H}(\hat{c}_i)$ where $\mathcal{H}$ is the entropy function. When the concepts are binary, it follows that $s_i = 1/|\hat{c}_i - 0.5|$ as in Lewis & Catlett (1994); Lewis (1995). Intuitively, uncertain concepts may have an adverse influence on making the correct target prediction, and thus, they are fixed first by this criterion.

Loss on concept prediction (LCP) It selects concepts with the largest loss on concept prediction compared to the ground-truth. Specifically, it defines $s_i = |\hat{c}_i - c_i|$. This scheme can be advantageous to increasing task performance since a low concept prediction error is likely to lead to a correct target prediction. Nonetheless, this score is unavailable in practice as the ground-truth is unknown at test time.

Contribution of concept on target prediction (CCTP) It selects concepts with the highest contribution on target prediction. Specifically, it sums up the contribution as $s_i = \sum_{j=1}^{M} \left|\hat{c}_i \frac{\partial f_j}{\partial \hat{c}_i}\right|$ where f_j is the output related to j-th target class and M is the number of classes. This scheme is inspired by methods to explain neural network predictions (Selvaraju et al., 2017).

Expected change in target prediction (ECTP) It selects concepts with the highest expected change in the target predictive distribution with respect to intervention. Specifically, it defines $s_i = (1-\hat{c}_i) D_{\text{KL}}(\hat{y}_{\hat{c}_i=0} \| \hat{y}) + \hat{c}_i D_{\text{KL}}(\hat{y}_{\hat{c}_i=1} \| \hat{y})$ where D_{KL} refers to the Kullback-Leibler divergence, and $\hat{y}_{\hat{c}_i=0}$ and $\hat{y}_{\hat{c}_i=1}$ refer to the new target prediction with $\hat{c}_i$ being intervened to be 0 and 1, respectively. The intuition behind this scheme is that it would be better to intervene on those concepts whose rectification leads to a large expected change in target prediction (Settles et al., 2007).

Expected uncertainty decrease in target prediction (EUDTP) It selects concepts with the largest expected entropy decrease in target predictive distribution with respect to intervention. Specifically, it defines $s_i = (1-\hat{c}_i)\mathcal{H}(\hat{y}_{\hat{c}_i=0}) + \hat{c}_i \mathcal{H}(\hat{y}_{\hat{c}_i=1}) - \mathcal{H}(\hat{y})$. Intuitively, it penalizes the concepts whose expected decrease in the target prediction entropy is low when intervened (Guo & Greiner, 2007).

3.2.1. Cost of Intervention

Note that the cost of intervention may differ by the choice of concept selection criteria. Specifically, let the theoretical cost of intervening on a concept be τ_i (*e.g.*, the time for an expert to look at the input and fix its attribute), and the theoretical cost of making inference on g and f be τ_g and τ_f, respectively. Then, the total cost of utilizing CCTP needed up to making the final prediction with n number of intervened concepts, for example, would be $\mathcal{O}(\tau_g + 3\tau_f + n\tau_i)$; here we assume that the cost of the backward pass on f is the same as τ_f. We summarize the cost of all concept selection criteria in Table 2.

3.3. Levels of Intervention

We find that intervention can be done at different levels given some auxiliary information about the structure of concepts or economic constraints put on practitioners. For example, it is often the case that datasets used to train CBMs have the grouping information for related concepts (Wah et al., 2011). Another situation worth consideration is where one has access to a batch of data to process with a budget constraint, and the goal is to maximize the overall task performance while minimizing the intervention effort (*e.g.*, examining medical images in a hospital). Taking into account these scenarios, we extend the intervention procedure at various levels to study the effectiveness of concept selection criteria.

Individual vs. Group intervention Intervention can be done depending on concept association (see Figure 2a):

- Individual (I): Concepts are assumed to be independent of each other and thus selected individually one at a time.
- Group (G): A group of related concepts is selected at once whose association information is subject to datasets. The selection score is computed by taking the average of selection scores of individual concepts within group.

Single vs. Batch intervention Intervention can be done depending on data accessibility (see Figure 2b):

- Single (S): Every test case is allocated with the same amount of intervention budget (*e.g.*, intervention counts). This could be useful for online systems where each test data comes in sequentially, and experts need to process as many cases as possible under a budget constraint.
- Batch (B): A batch of test cases shares a total intervention budget. This scheme could be particularly useful when the concept prediction is imbalanced toward easy cases, and one wants to focus on intervening on hard cases so as to maximize the overall task performance.

4. Evaluating Intervention Strategies

4.1. Experiment Settings

Dataset We experiment with three datasets: (1) CUB (Wah et al., 2011) – the standard dataset used to study CBMs, (2) SkinCon (Daneshjou et al., 2022b) – a medical dataset used to build interpretable models, and (3) Synthetic – the synthetic datasets we generate based on different causal graphs to conduct a wide range of controlled experiments. Extensive details of these datasets including preprocessing, label characteristics, data splits, and the generation process are provided in Appendix A.

Implementation We follow the standard implementation protocols as in previous works. The full details including model architectures and optimization hyperparameters are provided in Appendix B. Our code is available at `https://github.com/ssbin4/Closer-Intervention-CBM`.

Training

We consider the following training strategies similarly to Koh et al. (2020):

- IND: g and f are trained independently of each other. f always takes ground-truth concept values as input.
- SEQ: g and f are trained sequentially, g first and f next. f takes predicted concept values as input from trained g.
- JNT: g and f are trained jointly at the same time as a multi-objective. This results in increased initial task accuracy but comes with the price of decreased intervention effectiveness (Koh et al., 2020).
- JNT+P: similar to JNT but the input to f is sigmoid-activated probability distribution rather than logits.

Conceptualization We consider different forms of concept predictions as input to the target predictor at inference:

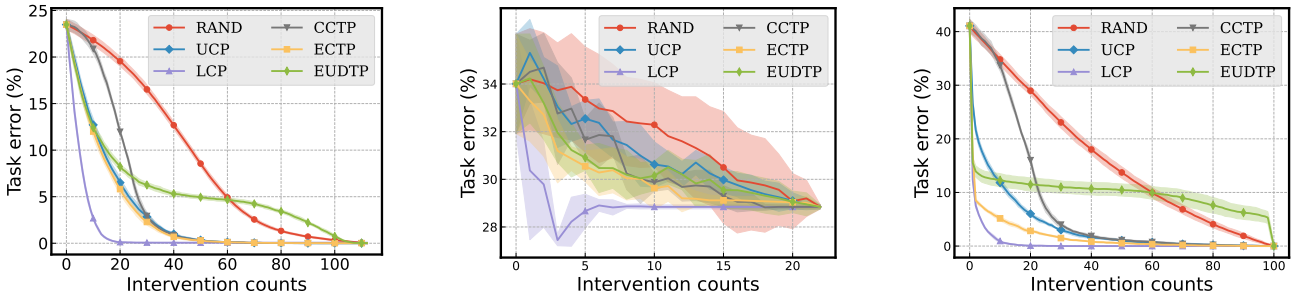
- SOFT: f takes real values of $\hat{c} \in [0,1]^k$ as soft representation of concepts (Koh et al., 2020).
- HARD: f takes binary values of $\hat{c} \in \{0,1\}^k$ as hard representation of concepts based on $\mathbb{1}[\hat{c} \geq 0.5]$ (Mahinpei et al., 2021). This prevents information leakage (Havasi et al., 2022) in exchange for decreased prediction performance.
- SAMP: m random samples are drawn by treating the soft concept prediction scores as a probability distribution, and the target prediction is made as an ensemble, *i.e.*, $\hat{y} = \frac{1}{m}\sum_{i=1}^{m} f(\hat{c})$ where $\hat{c}$ is binarized concept prediction (Havasi et al., 2022). We use $m = 5$ for the experiments.

4.2. Evaluating Concept Selection Criteria

We first evaluate the intervention effectiveness of concept selection criteria and present the results in Figure 3. Across all datasets, we find that the current practice of random intervention (RAND) is easily outperformed by the other alternatives in almost all cases with a significant margin. Specifically, in the CUB experiment, correcting 20 concepts by random intervention reduces the task error less than 4% whereas correcting the same amount based on the uncertainty of concept predictions (UCP) leads to more than 16% error reduction. To put it differently, RAND requires to intervene on 43 concepts in order to reduce the error by half, while it is only 12 concepts to fix for UCP to achieve the same reduction. In the SkinCon experiment, selecting concepts based on the expected change in target prediction (ECTP) leads the way among others, and yet, the scale of improvements over RAND is not as large. Note also that the strategy of fixing concepts with the largest loss first (LCP) performs exceptionally well in all cases. This is however due to the help of the ground-truth knowledge on concepts which is unavailable in practice. Nonetheless, we believe this can serve as an indicator to guide a better intervention strategy which we defer to future work.

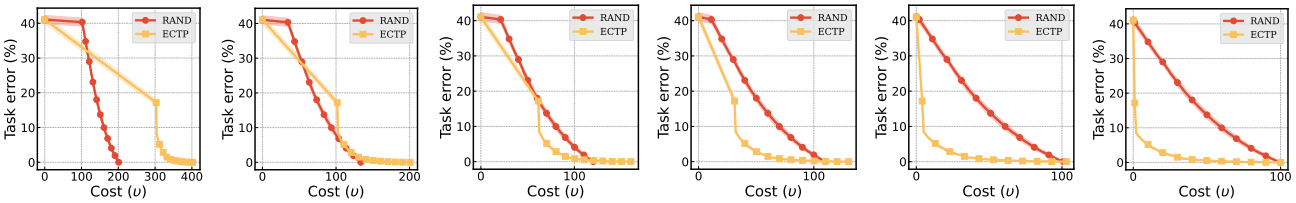
4.2.1. Reflecting cost of intervention

As we discussed in Section 3.2.1, the cost of intervention may differ by concept selection criteria. Taking into account this aspect, we set up experiments where we can evaluate the intervention effectiveness in terms of the theoretical cost. Specifically, we model the relationships between τ_i, τ_g, τ_f as $\tau_i = \alpha\tau_g$ and $\tau_g = \beta\tau_f$, which means that the cost of intervention (*e.g.*, time to fix a concept) is α-proportional to the cost of making inference on g, and likewise, τ_g is β-proportional to τ_f. Then we can evaluate the cost-reflected intervention effectiveness with respect to arbitrary unit (υ), and from which, we can further show how it transforms by controlling α and β.



(a) CUB (b) SkinCon (c) Synthetic

Figure 3: Intervention effectiveness of concept selection criteria (task error vs. number of concepts corrected by intervention) measured on I+S level. A more effective method would reduce the error more for the same number of concepts intervened.



(a) $\alpha = 0.01$ (b) $\alpha = 0.03$ (c) $\alpha = 0.05$ (d) $\alpha = 0.1$ (e) $\alpha = 1.0$ (f) $\alpha = 100.0$

Figure 4: Effect of α on intervention (on Synthetic). We fix $\tau_i = 1, \beta = 100, k = 100$. ECTP, the intervention method strongly evaluated previously, becomes less effective as α decreases. Here, kinked shapes are due to the relatively high initial cost on the first intervention before n becomes large.

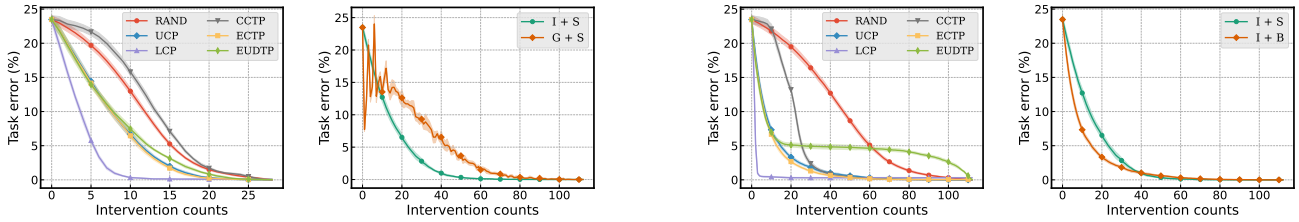
First, the result of changing α is plotted in Figure 4. As α becomes smaller RAND becomes very effective compared to ECTP. This makes sense because with small α, τ_i becomes relatively small and the other terms related to τ_g or τ_f dominate the cost of ECTP which is $\mathcal{O}\big(\tau_g + (2k+2)\tau_f + n\tau_i\big)$ as seen in Table 2. ECTP thus becomes penalized when it comes to the intervention effectiveness in the small α regime. In contrast, when α becomes larger, τ_i dominates the cost of ECTP as with increasing n, which in turn recovers the effectiveness of ECTP. The former can happen in extreme circumstances, for example, when using very large models (*i.e.*, large τ_g) or in places with a tight labor marker (*i.e.*, small τ_i in terms of monetized value). We clearly remark, however, that this can be seen as a hypothetical case and α will be much greater than 1 in realistic settings as summoning a domain expert for intervention would require more cost than a forward pass of neural networks.

We also experiment on changing β to control the relative cost between τ_g and τ_f. As a result, we find that when β is small ECTP can perform poorly while RAND can be effective as it only requires a single forward pass of f to make the final prediction. Furthermore, we extend this analysis to the CUB experiment with more realistic settings where τ_g and τ_f are set based on the wall-clock times of running each model, and τ_i is set based on the actual concept annotation time provided in the dataset. All of these results are put in Appendix C with detailed analysis for space reasons.

4.3. Analyzing Intervention Levels

As seen in Figure 5a, most criteria still remain more effective than RAND in group-wise single (G + S) intervention. Specifically, RAND needs 39.3% (11 out of 28), while UCP needs 25.0% (7 out of 28) of the groups to be intervened to decrease the task error by half. However, CCTP does not outperform RAND this time. We also find a similar pattern for the batch case G + B (see Figure 14 in Appendix D). We suspect that calculating the mean of the scores loses some discriminative information in some selection criteria and perhaps a different surrogate needs to be designed.

In addition, we find that group-wise intervention is in general less effective than individual counterpart with the same budget of intervention expense (see Figure 5b). Intuitively, correcting concepts within the same group may not provide rich information as opposed to selecting concepts across different groups with the same intervention counts. Nonetheless, we remark that group-wise intervention can potentially be cost-effective when concepts within the same group are mutually exclusive, which depends on how the concepts are annotated during the creation of datasets.



(a) G+S level (b) (I+S) vs. (G+S) (c) I+B level (d) (I+S) vs. (I+B)

Figure 5: Comparing the effects of different intervention levels using the CUB dataset. Here, intervention counts denote the number of intervened groups and average number of intervened concepts for G and B, respectively. We fix the selection criterion to be UCP in (b) and (d) while all other cases are provided in Appendix D.

The proposed concept selection criteria also remain effective for batch intervention (B) as seen in Figure 5c. Interestingly, batch intervention turns out to be more effective when compared to single (S) as well as seen in Figure 5d. This trend holds true for other criteria besides UCP except for CCTP and extends to group-wise batch (G+B) intervention (see Appendix D for full results).

4.4. Considering Training and Conceptualization

Effect of training scheme As seen in Figure 6a, intervention is in general the most effective under the IND training scheme. We believe that this is because f is not trained with the ground-truth concept labels in the case of SEQ and JNT(+P), and fixing concept predictions for these schemes may not work as well. We also find that EUDTP becomes much less effective under SEQ or JNT than other alternatives and actually underperforms RAND (see Appendix E). Hence, the effectiveness of a criterion can depend on which training strategy to use, implying the need of comprehensive evaluations for newly developed criteria.

For the SkinCon dataset, however, intervening on the concepts under SEQ, JNT, JNT + P strategies rather increases the average task error regardless of the concept selection criteria. Specifically, training under JNT already achieves low task error and applying intervention does not help reduce it further (see Figure 6b). We hypothesize that this is due to some inherent characteristics of the dataset as well as limited concepts provided in the bottleneck, resulting in the negative influence on making correct task predictions with binarized concepts. This can potentially correspond to the known issue of information leakage in CBMs (Mahinpei et al., 2021; Havasi et al., 2022).

Effect of conceptualization We find that HARD and SAMP may begin with high task error compared to SOFT as expected. However, when making use of the developed concept selection criteria such as UCP, the gap between these conceptualization methods decreases much faster with more intervention compared to RAND as seen in Figures 6c and 6d. This result is consistent across different training strategies and datasets (see Appendix F).

5. Analyzing Intervention with Synthetic Data

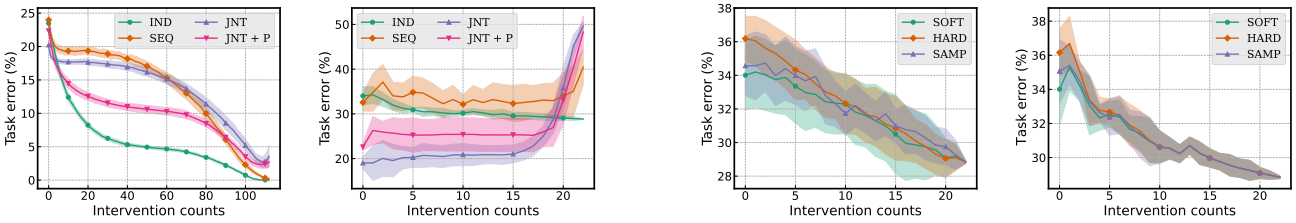
We have observed that intervention can often yield different results over datasets. Precisely, intervening on all concepts decreases the task error down to 0% on CUB, whereas the amount of decrease is much less and the average task error remains still high around 29% on SkinCon. Also, the relative order of effectiveness between concept selection criteria can vary. We find that it is difficult to unravel these findings if only experimenting on real datasets as in previous work (Koh et al., 2020; Chauhan et al., 2022; Sheth et al., 2022; Zarlenga et al., 2022). To provide an in-depth analysis, we develop a framework to generate synthetic datasets based on three different causal graphs that control the followings: input noise, hidden concepts, and concept diversity.

5.1. Generating Synthetic Data

CASE 1: Noisy input Real-world data contains a lot of random noise coming from various sources (*e.g.*, lighting). We construct a causal graph to consider this case where the Gaussian noise is added on input data (see Figure 7a).

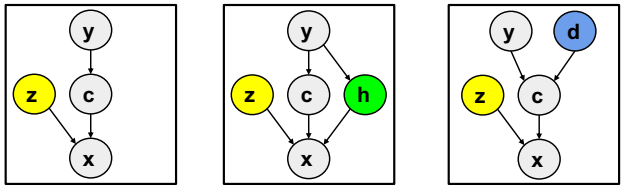
CASE 2: Hidden concept When a subset of concepts is unknown or hidden, the target prediction is made incomplete with only available concepts as deep representations are not fully captured in the bottleneck layer. We design a causal graph for this case and generate synthetic data for which some concepts that are necessary to make correct target predictions are hidden on purpose (see Figure 7b).

CASE 3: Diverse concept Examples within the same class can have different values for the same concept in realistic settings. For instance, simple concept-level noise or fine-grained sub-classes (*e.g.*, `'black swan'` and `'white swan'` for `'swan'` class) can make such diverse concept values. We construct a causal graph to generate such data for which concept values can vary probabilistically and inputs



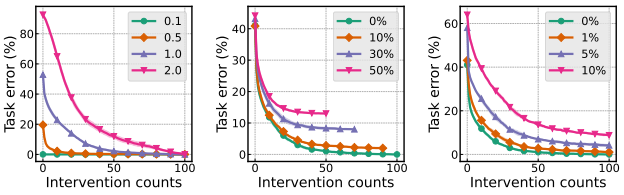
(a) Training on CUB (b) Training on SkinCon (c) Conceptualization: RAND (d) Conceptualization: UCP

Figure 6: Comparing the effects of different training strategies (a,b) and conceptualization methods (c, d). We choose EUDTP as the concept criterion for (a,b) and SkinCon as the dataset for (c, d). We provide all other results in Appendices E and F.



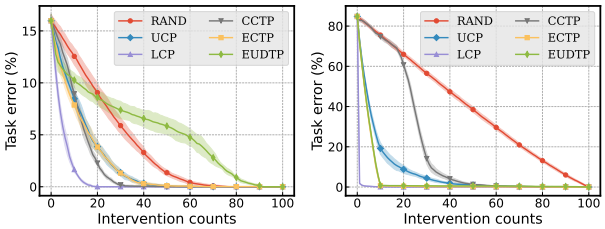
(a) Noisy input (b) Hidden concept (c) Diverse concept

Figure 7: Causal graphs for generating synthetic datasets. z, h, and d represent factors of input noise, hidden concepts, and concept diversity, respectively. The full details of the data generation process are provided in Appendix A.3.



(a) Noisy input (b) Hidden concept (c) Diverse concept

Figure 8: Effects of data on intervention with UCP. Each plot is with different values of the variance of noise (z), the ratio of hidden concepts (h), and the probability to perturb the concept values (d), respectively.



(a) $\gamma = 1$ (b) $\gamma = 10$

Figure 9: Intervention effectiveness with different sub-group size γ. The relative order of effectiveness between selection criteria changes significantly according to γ.

are produced according to these concepts (see Figure 7c).

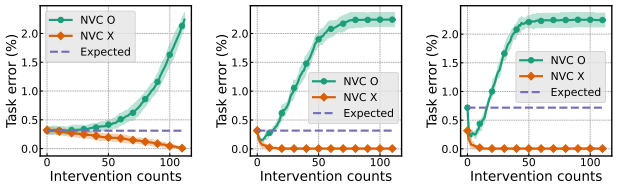
5.2. Results

First, we display the effect of input noise in Figure 8a. The initial task error increases with a level of noise (z) due to the poor performance on concept prediction. Specifically, we need 17 intervention counts to decrease the task error by half with extremely noisy data ($z = 2.0$) while correcting only 2 concepts yields the same effect for a moderate level of noise case ($z = 0.5$). In contrast, the initial task error is already near 0% with an extremely small level of noise ($z = 0.1$) where we do not need intervention at all.

Next, we evaluate the effect of hidden concepts in Figure 8b. The final task error increases with more hidden concepts, and thus, intervention becomes less effective. Specifically, the error is still high around 13% when half of the concepts are hidden ($h = 50\%$) while it reaches zero error without hidden concepts ($h = 0\%$). This is due to the fact that the target prediction cannot be made with complete information when there exist hidden concepts, which is often the case for constructing CBMs in realistic settings.

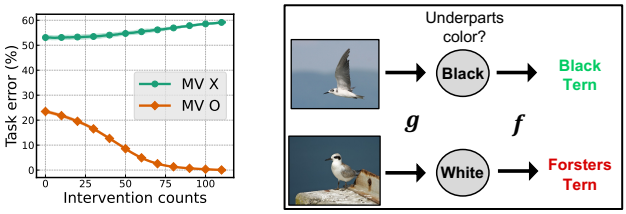
We also find that generating more diverse concept values within the same class increases both initial and final task errors, making intervention less effective (see Figure 8c). This is because learning discriminative representations for target prediction would be a lot more difficult. To circumvent this issue, many previous works (Koh et al., 2020; Zarlenga et al., 2022; Havasi et al., 2022) attempt to preprocess the data so as to force concepts within the same class have the same value. However, this may have an adverse effect on model fairness as we discuss in Section 6.

Furthermore, we discover that different sub-group sizes can change the relative ordering of intervention effectiveness between concept selection criteria. Here, we define a sub-group as classes with similar concept values and denote its size as γ. Interestingly, EUDTP becomes less effective with a small group size ($\gamma = 1$) even compared to RAND whereas it becomes the most effective when $\gamma = 10$ except for LCP as seen in Figure 9. We believe that it is because classes within the same sub-group are classified more easily by decreasing uncertainty in target prediction using EUDTP when γ is large.



(a) RAND (b) UCP (c) ECTP

Figure 10: Effect of NVC on task error. Intervention is done on the CUB images for which concept prediction is 100% accurate, and yet, NVC keeps on increasing the task error. NVC O and NVC X each correspond to the result with and without NVC.



(a) Advantage of MV (b) Disadvantage of MV

Figure 11: Effects of majority voting (MV) on target prediction. MV O and MV X each correspond to the result with and without MV. (a) While it helps decrease task error on intervention, (b) it yields biased predictions against minorities.

The result indicates that the behavior of a criterion can vary significantly across different datasets and again demonstrate the necessity of a comprehensive evaluation of the newly developed criteria. We refer to Appendix G for results on the effect of some other factors on intervention.

6. Pitfalls of Intervention Practices

So far we have focused on analyzing the effectiveness of intervention procedure in many aspects. In this section, we add another dimension, namely, reliability and fairness of the current intervention practices, to help advance toward trustworthy and responsible machine learning models.

6.1. Nullifying Void Concepts Increases Task Error

Does intervention always help target prediction? Contrary to expectation, we find that the answer is no, and in fact, intervention can rather increase the task error. To verify this, we set up an ablation experiment using the CUB dataset where intervention is conducted only on the cases for which all concepts are predicted correctly with zero error; ideally intervention should have no effect in this case. The results are quite the opposite as presented in Figure 10. The task error keeps on increasing as with more intervention, and the prediction error reaches to more than seven times as much as that with no intervention.

It turns out that it is due to nullifying void concepts (NVC), a common practice of treating unsure concepts by setting them to be simply zero, which leads to this catastrophic failure. For example, just because the wing part of a bird species is invisible does not necessarily mean that the concept '`wing color:black`' should be zero valued; this bird can fall in the class of '`Black_Tern`' whose wing color is actually black. We identify that this seemingly plausible tactic can in fact mistreat invalid concepts, and therefore, for invalid cases applying NVC intervention should be avoided.

6.2. Majority Voting Neglects Minorities

Another common practice often taken by the community (Koh et al., 2020; Zarlenga et al., 2022; Havasi et al., 2022) is to coalesce concept values among the same class by forcing them to have their majority votes (MV). As a preprocessing, this tactic can dramatically improve the task performance as we demonstrate in Figure 11a. This is quite obvious by now as with our Synthetic experiment results in Section 5.2 where we show that high concept diversity can deteriorate the target prediction performance.

However, it turns out that MV can have a negative impact on model fairness by ignoring minority samples. As a concrete example, consider the CUB dataset in which the majority of images of '`black tern`' class have black underparts while some minority samples have white underparts. When MV is used in this case, we find that the underparts color predictions for the minorities are mis-guided to be black, which correspond to the majority-voted values, so as to yield the correct target prediction; if the minorities follow their own concept values before MV otherwise, it can lead to an incorrect target prediction (see Figure 11b). Intervention can even aggravate the situation since it can decrease the task error for the minorities only when the predicted concept value is changed to the majority-voted value (black). In this sense, target predictions become biased toward the majority when MV is used.

This scenario can be problematic in the real world when the dataset contains sensitive concepts, e.g., gender or race. Consider the case where the target task is to predict the occupation of a person based on his/her look and 'race' is included in the concepts. When most 'physicians' are Caucasians and if we apply MV in this case, then an 'Asian physician' can be correctly classified only when he is predicted as a Caucasian; otherwise, it would lead to an incorrect target prediction. While this might be somewhat exaggerated, we remark that this kind of situation can happen in practice. Besides, MV also forces to misconduct intervention at test time with the majority votes, which is

neither available in practice nor considered fair. We defer addressing the trade-off between performance and fairness to future work.

7. Discussion and future work

In this section, we discuss our key findings, their potential implications to the community, and possible future research directions.

In-depth analysis of intervention procedure We design and conduct a wide variety of new experiments from scratch to investigate the effectiveness of the current intervention procedure of CBMs. In a nutshell, our results reveal that not only is it the specific way of selecting which concept to intervene, but also how to intervene on what data under which environments matters to the degree of drastically changing results. Future works can extend our analysis to theoretically investigate the intervention strategies in more detail.

Benchmark for evaluating concept selection methods Our evaluation protocol can serve as a way to evaluate any newly developed concept selection methods for their effectiveness. We also provide a framework to generate synthetic data based on which the effectiveness of proposed methods can be tested under various circumstances.

Analyzing the cost of intervention The effectiveness of concept selection criteria can change when reflecting the cost of intervention (see Section 4.2.1). Specifically, we find that a strongly evaluated criterion can become less effective in hypothetical cases considering the size of the models or the status of the labor markets. This indicates that choosing the concept selection criterion should reflect the available budgets and environments at test time, especially in some extreme environments.

Identifying the effect of data on intervention The effectiveness of the intervention procedure can vary quite significantly depending on some unknown characteristics of the real-world datasets (see Section 5). For example, intervention becomes less effective on datasets containing more hidden concepts or more diverse concept values within the same class. Practitioners should take into account this aspect when developing and deploying CBMs since intervention may not work effective as expected.

Reliability and fairness of intervention While the current trend is mostly focused on developing new intervention methods, we discovered somewhat unexpected and previously unknown issues, which can be critical for ensuring reliability and fairness of the intervention procedure (see Section 6). To be more specific, intervention can sometimes increase the task error contrary to the expectation and have a negative impact on model fairness by making the predictions biased toward the majority. We call for future work to address these problems before blindly adopting CBMs in practice.

Extension of our work to other settings We remark that we have only focused on the classification tasks, considering the characteristics of the real-world datasets used in the literature (Koh et al., 2020; Zarlenga et al., 2022; Havasi et al., 2022) [1]. Extension of the intervention strategies to the regression problems with real-valued concepts or targets can be a promising avenue for future works. Analyzing intervention under more diverse settings could also be interesting, such as introducing architectural variations with hard autoregressive models (Havasi et al., 2022) or concept embedding models (Zarlenga et al., 2022).

8. Conclusion

The intervention procedure of CBMs has been unattended in previous work despite its critical impact on practitioners. In this work, we study a wide range of aspects regarding the procedure and provide an in-depth analysis for the first time in the literature. Specifically, we develop various concept selection criteria that can be used for intervention and demonstrate that their behaviors can vary quite significantly based on an array of factors including intervention levels, cost, training, conceptualization, and data characteristics. We also find several pitfalls in the current practices that need a careful addressing to be deployed in realistic settings. We plan to investigate further on developing more effective and reliable intervention strategies in future work.

Acknowledgement

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH) and No.2022-0-00959, (part2) Few-Shot learning of Causal Inference in Vision and Language for Decision Making) and National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2022R1C1C1013366, 2022R1F1A1064569, RS-2023-00210466).

References

Alvarez Melis, D. and Jaakkola, T. Towards robust interpretability with self-explaining neural networks. *NeurIPS*, 2018.

Bahadori, M. T. and Heckerman, D. E. Debiasing concept-based explanations with causal analysis. *ICLR*, 2021.

[1] Concept and target variables in the OAI dataset (Nevitt et al., 2006) take 4 integer values and thus the tasks can be easily converted into the classification problem.

Chauhan, K., Tiwari, R., Freyberg, J., Shenoy, P., and Dvijotham, K. Interactive concept bottleneck models. *AAAI*, 2022.

Daneshjou, R., Vodrahalli, K., Novoa, R. A., Jenkins, M., Liang, W., Rotemberg, V., Ko, J., Swetter, S. M., Bailey, E. E., Gevaert, O., et al. Disparities in dermatology ai performance on a diverse, curated clinical image set. *Science advances*, 2022a.

Daneshjou, R., Yuksekgonul, M., Cai, Z. R., Novoa, R. A., and Zou, J. Skincon: A skin disease dataset densely annotated by domain experts for fine-grained debugging and analysis. *NeurIPS Datasets and Benchmarks Track*, 2022b.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. *CVPR*, 2009.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 2017.

Ghorbani, A., Wexler, J., Zou, J. Y., and Kim, B. Towards automatic concept-based explanations. *NeurIPS*, 2019.

Groh, M., Harris, C., Soenksen, L., Lau, F., Han, R., Kim, A., Koochek, A., and Badri, O. Evaluating deep neural networks trained on clinical images in dermatology with the fitzpatrick 17k dataset. *CVPR*, 2021.

Guo, Y. and Greiner, R. Optimistic active-learning using mutual information. *IJCAI*, 2007.

Havasi, M., Parbhoo, S., and Doshi-Velez, F. Addressing leakage in concept bottleneck models. *NeurIPS*, 2022.

Jordan, M. I. and Mitchell, T. M. Machine learning: Trends, perspectives, and prospects. *Science*, 2015.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *ICML*, 2018.

Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. Concept bottleneck models. *ICML*, 2020.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature*, 2015.

Lewis, D. D. A sequential algorithm for training text classifiers: Corrigendum and additional data. *Acm Sigir Forum*, 1995.

Lewis, D. D. and Catlett, J. Heterogeneous uncertainty sampling for supervised learning. *Machine learning proceedings*, 1994.

Mahinpei, A., Clark, J., Lage, I., Doshi-Velez, F., and Pan, W. Promises and pitfalls of black-box concept learning models. *Workshop on XAI, ICML*, 2021.

Marconato, E., Passerini, A., and Teso, S. Glancenets: Interpretabile, leak-proof concept-based models. *NeurIPS*, 2022.

Margeloiu, A., Ashman, M., Bhatt, U., Chen, Y., Jamnik, M., and Weller, A. Do concept bottleneck models learn as intended? *Workshop on Reponsible AI, ICLR*, 2021.

Nevitt, M., Felson, D., and Lester, G. The osteoarthritis initiative. *Protocol for the cohort study*, 2006.

Sawada, Y. and Nakamura, K. Concept bottleneck model with additional unsupervised concepts. *IEEE Access*, 2022.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. *CVPR*, 2017.

Settles, B., Craven, M., and Ray, S. Multiple-instance active learning. *NeurIPS*, 2007.

Sheth, I., Rahman, A. A., Sevyeri, L. R., Havaei, M., and Kahou, S. E. Learning from uncertain concepts via test time interventions. *Workshop on Trustworthy and Socially Responsible Machine Learning, NeurIPS*, 2022.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. *CVPR*, 2016.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Yuksekgonul, M., Wang, M., and Zou, J. Post-hoc concept bottleneck models. *ICLR*, 2023.

Zarlenga, M. E., Barbiero, P., Ciravegna, G., Marra, G., Giannini, F., Diligenti, M., Shams, Z., Precioso, F., Melacci, S., Weller, A., et al. Concept embedding models. *NeurIPS*, 2022.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Learning deep features for discriminative localization. *CVPR*, 2016.

A. Datasets

A.1. CUB

CUB (Wah et al., 2011) is the standard dataset used to study CBMs in the previous works (Koh et al., 2020; Zarlenga et al., 2022; Havasi et al., 2022; Sawada & Nakamura, 2022). There are 5994 and 5794 examples for train and test sets in total, in which each example consists of the triplet of (image x, concepts c, label y) of a bird species. All the concepts have binary values; for example, the '`wing color:black`' for a given bird image can be either 1 (for true) or 0 (for false). Following previous works (Koh et al., 2020; Sawada & Nakamura, 2022; Zarlenga et al., 2022), we perform so-called majority voting as pre-processing so that images of the same class always have the same concept values; for example, if more than half of the crow images have true value for the concept '`wing color:black`' then this process converts all concept labels for images belonging to the crow class to have the same true value. Since the original concept labels are too noisy, this procedure helps to increase the overall performance. However, it can be potentially harmful to model fairness in some cases as we address in Section 6.2. We also remove concepts that are too sparse (*i.e.*, concepts that are present in less than 10 classes) which results in 112 out of 312 concepts remaining. It is suggested in Koh et al. (2020) that including these sparse concepts in the concept layer makes it hard to predict their values as the positive training examples are too scarce.

A.2. SkinCon

SkinCon (Daneshjou et al., 2022b) is a medical dataset which can be used to build interpretable machine learning models. The dataset provides densely annotated concepts for 3230 images from Fitzpatrick 17k skin disease dataset (Groh et al., 2021), which makes a triplet of (image x, concepts c, disease label y) of a skin lesion for each example. Since training and test sets are not specified in the SkinCon dataset, we randomly split the dataset into $70\%, 15\%, 15\%$ of training, validation, and test sets respectively. The dataset provides various levels of class labels ranging from individual disease labels with 114 classes to binary labels representing if the skin is benign or malignant. Following the experiments with Post-hoc CBM (Yuksekgonul et al., 2023) introduced in Daneshjou et al. (2022b), we use the binary labels for the target task and only use 22 concepts which are present in at least 50 images. Since the binary class labels are highly imbalanced (87% vs. 13%), we train the target predictor f with weighted loss and use the average of per-class error as the metric instead of overall error for a fair comparison.

A.3. Synthetic dataset

Algorithm 1 Generating synthetic data

1: Sample $p_i \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha)$ for $i = \{1, 2, \cdots, k\}$
2: **for** group $\ell = 0, 1, \cdots, k/\gamma - 1$ **do**
3: Sample $\zeta_i \sim \mathcal{U}_{[0,1]}$ and set $\ell_i = \mathbb{1}[\zeta_i \geq p_i]$ for $i = \{1, 2, \cdots, k\}$
4: **for** $y = 1, \cdots, \gamma$ **do**
5: Sample $i_y \in \{1, 2, \cdots, k\}$ uniformly at random without replacement
6: Set $c_i^j = \neg \ell_i$ if $i = i_y$ and $c_i^j = \ell_i$ otherwise (class index $j = \gamma * \ell + y$)
7: **end for**
8: **end for**
9: Generate $W_x \in \mathbb{R}^{k \times k}$ with each element distributed according to the unit normal distribution $\mathcal{N}(0, \sigma_w)$
10: **for** class $j = 1, \cdots, k$ **do**
11: Generate ν samples for class j as $x = W_x \cdot c^j + z$ where $z \sim \mathcal{N}(0, \sigma_z)$
12: **end for**

We generate the synthetic data following Algorithm 1 to test the effect of dataset characteristics on intervention. Here, we first assume that all examples within the same class share the same concept values and denote the i-th concept value of j-th class as c_i^j. We also assume for simplicity that the dimensionality of inputs and the number of target classes are the same as the number of concepts k, following Bahadori & Heckerman (2021). In line 1, μ_α and $p_i = P(c_i = 0)$ each represent the overall sparsity level of the concepts (proportion of concepts with value 0) and the probability of i-th concept taking value 0, respectively. We set μ_α to be 0.8 considering that 80% of the concepts have value 0 in the CUB dataset. We then divide classes into k/γ sub-groups of size γ to make those within the same group have similar concept values. Note that the classes within each sub-group only differ by two concept values as seen in line 6. We set

$\gamma = 2, k = 100, \nu = 100, \sigma_\alpha = 0.1, \sigma_w = 0.1, z_\alpha = 0.8$ unless stated otherwise. We randomly divide the generated examples into 70% of training sets, 15% of validation sets, and 15% of test sets.

To generate the data with hidden concepts, we randomly pick $h\%$ of the concepts and remove them from the concept layer of CBMs. For training the models and intervention experiments, we only consider the remaining concepts. In addition, a new dataset with diverse concepts can be easily produced by introducing a single variable d and reversing the value of each concept from the previously generated dataset with probability d. In other words, d stands for a factor to give variations to concept-target pairs that can exist in real world datasets, and it differs from the role of z which controls the noise level to the input.

B. Architectures and Training

CUB For the CUB dataset, we use Inception-v3 (Szegedy et al., 2016) pretrained on Imagenet (Deng et al., 2009) for the concept predictor g and 1-layer MLP for the target predictor f respectively following the standard setup as in Koh et al. (2020). Here, both g and f are trained with the same training hyperparameters as in Koh et al. (2020). We used $\lambda = 0.01$ for JNT and JNT+P whose values were directly taken from Koh et al. (2020). For the experiments without majority voting (Figure 30 in Appendix H), we use Inceptionv3 pretrained on the Imagenet for g and 2-layer MLP for f with the dimensionality of 200 so that it can describe more complex functions. We searched the best hyperparameters for both g and f over the same sets of values as in Koh et al. (2020). Specifically, we tried initial learning rates of $[0.01, 0.001]$, constant learning rate and decaying the learning rate by 0.1 every $[10, 15, 20]$ epoch, and the weight decay of $[0.0004, 0.00004]$. After finding the optimal values of hyperparameters whose validation accuracy is the best, we trained the networks with the same values again over 5 different random seeds on both training and validation sets.

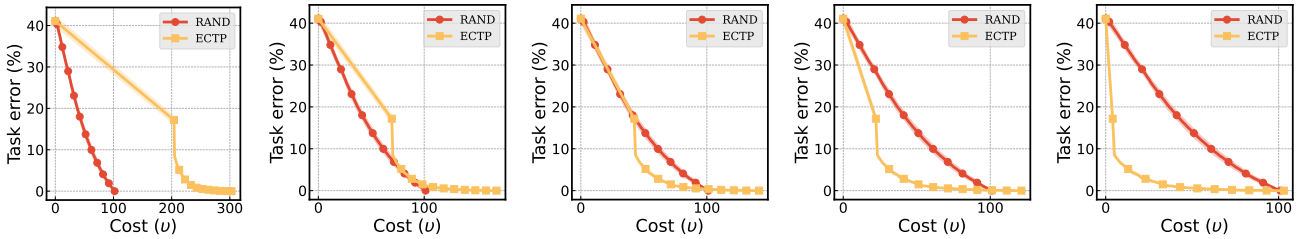
SkinCon For the SkinCon dataset, we fine-tune Deepderm (Daneshjou et al., 2022a) for the concept predictor g, which is the Inception-v3 network trained on the data in Esteva et al. (2017), and train 1-layer MLP for the target predictor f. We select hyperparameters that achieve the best performance (in terms of overall accuracy and average per-class accuracy for g and f respectively) in the validation set. Specifically, we tried initial learning rates of $[0.0005, 0.001, 0.005]$, and constant learning rate and decaying the learning rate by 0.1 every 50 epoch. Here, we did not use the weight decay factor. For JNT and JNT+P training strategies, we tried concept loss weight λ of $[0.01, 0.1, 1.0, 5.0]$, but all of the values failed to decrease the task error at intervention. As in the CUB dataset, we trained the networks with the best hyperparameters over 5 different random seeds on the both training and validation sets.

Synthetic For the synthetic datasets, we use 3-layer MLP of hidden layer size $\{100, 100\}$ for g and a single linear layer for f, as similar to Zarlenga et al. (2022). For all the experiments, we tried constant learning rates of $[0.01, 0.1, 1.0]$ without learning rate decay or weight decay factor and trained the networks with the best hyperparameters over 5 different random seeds on the training sets. We used $\lambda = 0.1$ for JNT and JNT+P whose values were determined by grid search over $[0.01, 0.1, 1.0]$.

C. More on Reflecting Cost of Intervention

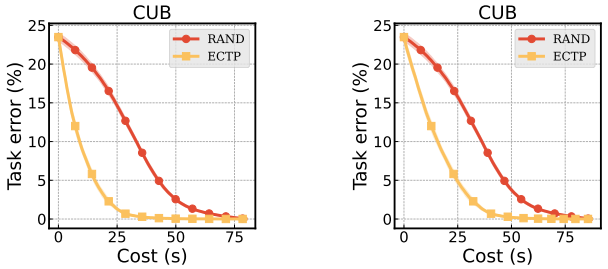
As β becomes smaller RAND becomes more effective compared to ECTP (see Figure 12). This is because with small β, τ_g becomes marginalized in the cost of ECTP which is $\mathcal{O}(\tau_g + (2k+2)\tau_f + n\tau_i)$, and therefore, the intervention effectiveness of ECTP is penalized as with increasing k compared to RAND which only requires a single forward pass of f.

In addition, we experiment with more realistic settings for the CUB where we set τ_i as the concept annotation time (seconds) provided in the dataset and τ_g, τ_f as the wall-clock times for the inference. Specifically, we set $\tau_i \approx 0.7$ by dividing the annotation time into the number of concepts within the group and taking the average. In addition, $\tau_g \approx 18.7 * 10^{-3}$ and $\tau_f \approx 0.03 * 10^{-3}$ are acquired by measuring the inference time with RTX 3090 GPU and taking the average of 300 repetitions. In this setting, τ_i dominates the others, *i.e.*, α is large, and the relative effectiveness between the criteria remains the same as seen in Figure 13. Nonetheless, we remark that the result can change with different model sizes or GPU environments in extreme cases. We also considered a more detailed case where we do not directly take the average of τ_i's (concept annotation time) at once but rather take the average per intervention step, reflecting differences of intervention costs between different concepts. The relative rankings between RAND and ECTP do not change but interestingly we have



(a) $\beta = 1$ (b) $\beta = 3$ (c) $\beta = 5$ (d) $\beta = 10$ (e) $\beta = 100$

Figure 12: Effect of β on intervention. We fix $\tau_i = 1, \alpha = 1, k = 100$. ECTP, the concept selection criteria strongly evaluated previously, becomes less effective as β decreases.



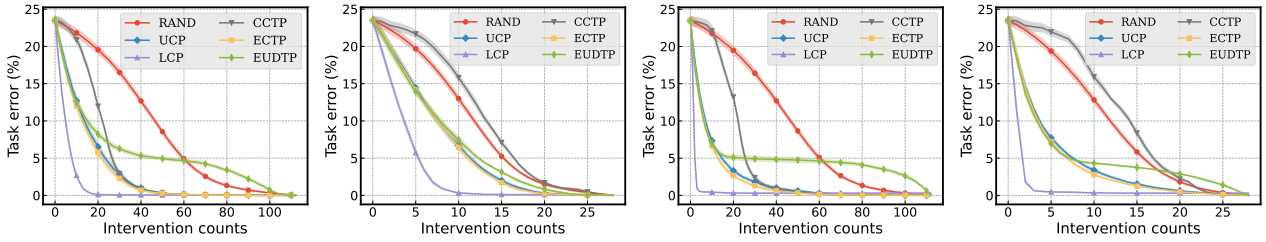
(a) τ_i set as the average among all concepts

(b) τ_i set as the average per each intervention step

Figure 13: Comparison between concept selection criteria in terms of the intervention cost for the CUB. Here, cost represents the seconds for concept annotation time and inference times for g, f.

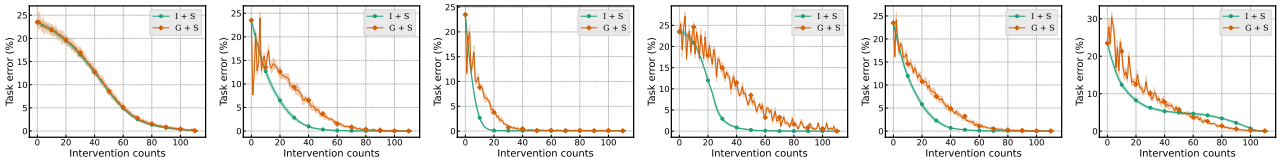
found that ECTP first selects the concepts which require more intervention costs (*i.e.*, more concept annotation time).

D. More Results on the Effect of Intervention Levels on Intervention



(a) I+S (b) G+S (c) I+B (d) G+B

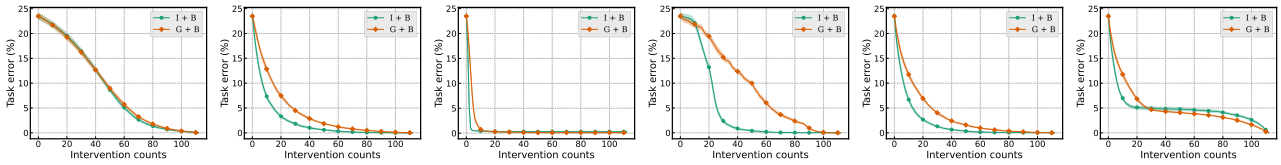
Figure 14: Comparison between intervention criteria under different levels for the CUB.



(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

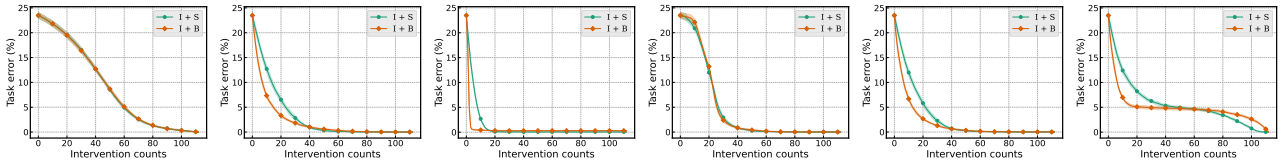
Figure 15: Comparison between I+S vs. G+S for the CUB.

The comparison between I+S and G+S using different concept selection criteria is presented in Figure 15. Individual



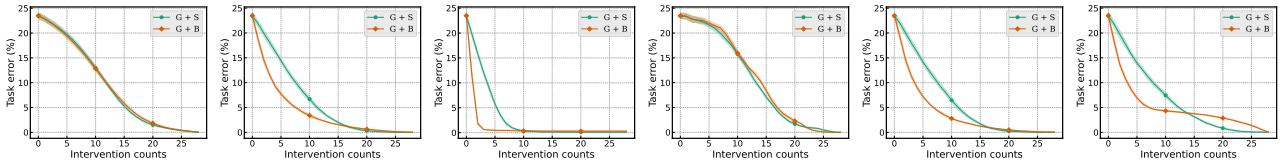
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 16: Comparison between I+B vs. G+B for the CUB.



(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 17: Comparison between I+S vs. I+B for the CUB.



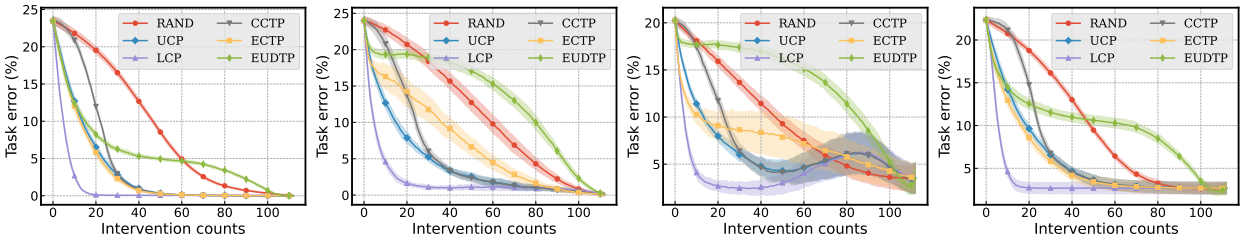
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 18: Comparison between G+S vs. G+B for the CUB. For G+B, each point is plotted when the average number of intervened concepts per image first exceeds each integer value.

intervention is in general more effective than group-wise intervention except for RAND criterion. We find similar results for the comparison between I+B and G+B (see Figure 16). We also note that CCTP becomes less effective in G levels as seen in Figure 14.

Batch intervention is either more effective or at least as competitive as single intervention across different concept selection criteria as seen in Figure 17. In Figure 18, we observe that G+B are also more effective than G+S level. CCTP does not show much difference between S and B. It is because the target predictor f is a simple linear layer for our experiments and thus $\frac{\partial f_j}{\partial \hat{c}_i} = w_{ij}$ is fixed for all examples where w_{ij} is the weight of i-th concept to j-th class in f.

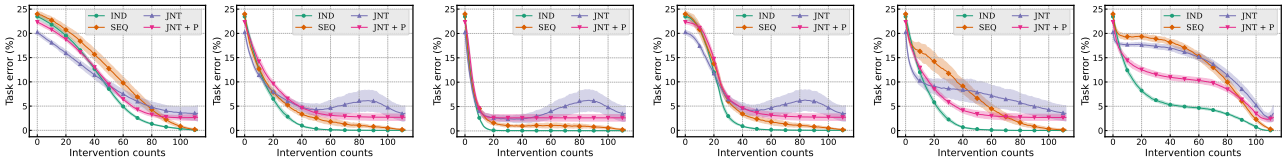
E. More Results on the Effect of Training Strategies on Intervention



(a) IND (b) SEQ (c) JNT (d) JNT + P

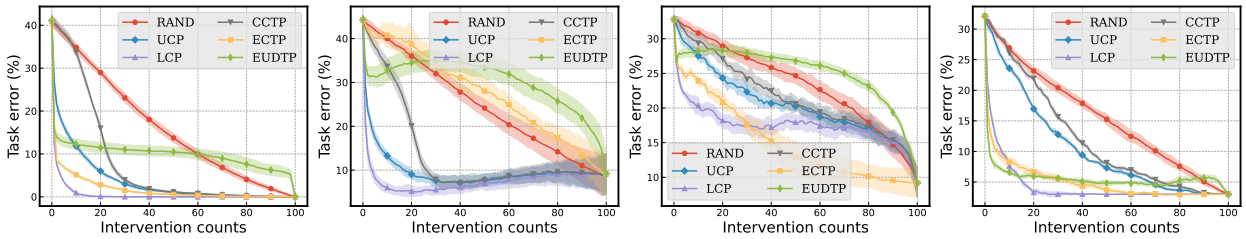
Figure 19: Comparison between concept selection criteria using different training strategies for the CUB. For JNT, JNT + P, we present the results when $\lambda = 0.01$.

The results for the CUB dataset are presented in Figure 19. Note that EUDTP becomes even less effective than RAND in SEQ



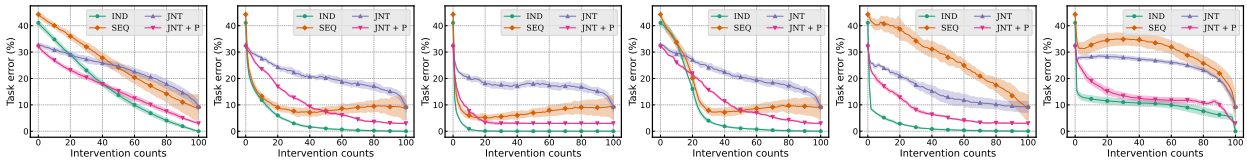
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 20: Comparison between different training strategies for a fixed concept selection criterion for the CUB.



(a) IND (b) SEQ (c) JNT (d) JNT + P

Figure 21: Comparison between concept selection criteria using different training strategies for the Synthetic. For JNT, JNT + P, we present the results when $\lambda = 0.1$.

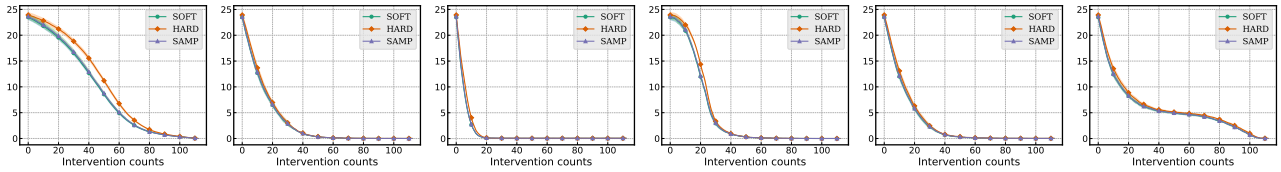


(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 22: Comparison between different training strategies for a fixed concept selection criterion for the Synthetic.

and JNT. For the synthetic datasets, EUDTP also becomes much less effective as in the CUB dataset (see Figure 21). Note that when using JNT or JNT+P training schemes, LCP may not be the best choice as the target predictor f is not trained with the ground-truth concept values and thus rectifying the concept with the highest prediction loss does not always guarantee the decrease in the task error. Comparisons between different training strategies for a fixed concept selection criterion in the CUB and Synthetic are presented in Figures 20 and 22.

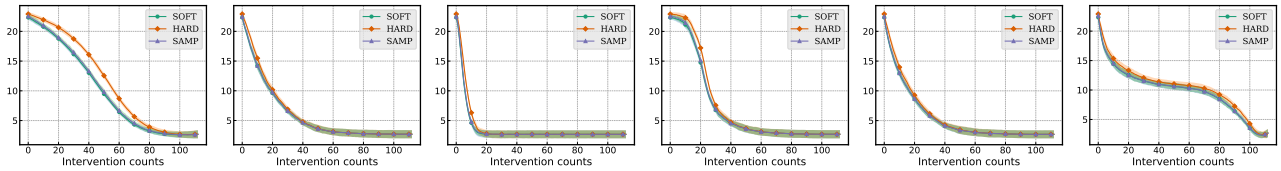
F. More Results on the Effect of Conceptualization Methods on Intervention



(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

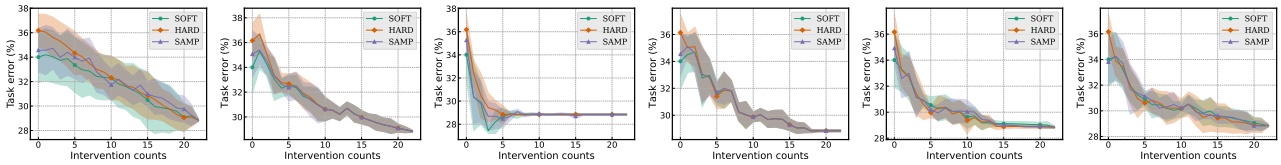
Figure 23: Intervention results under different conceptualization methods using various concept selection criteria. Here, we used IND training strategy for the CUB.

Across all the datasets and concept selection criteria, utilizing effective criteria can reduce the gap between different conceptualization strategies much faster than RAND criterion as seen in Figures 23 to 27.



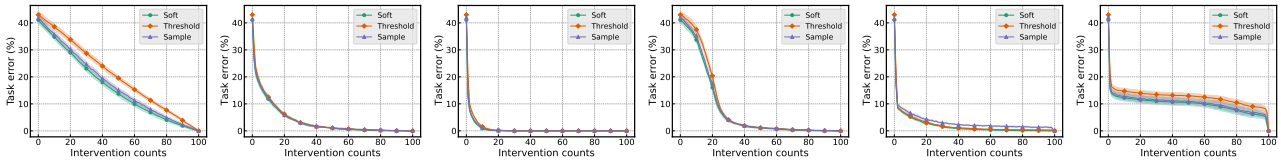
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 24: Intervention results under different conceptualization methods using various concept selection criteria. Here, we used JNT + P training strategy for the CUB.



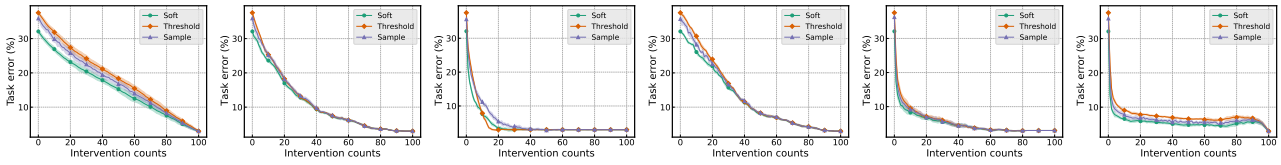
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 25: Intervention results under different conceptualization methods using other concept selection criteria. Here, we used IND training strategy for the SkinCon.



(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 26: Intervention results under different conceptualization methods using various concept selection criteria. Here, we used IND training strategy for the synthetic dataset.



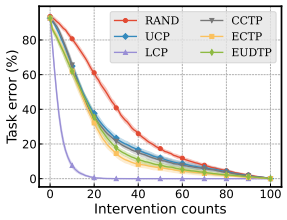
(a) RAND (b) UCP (c) LCP (d) CCTP (e) ECTP (f) EUDTP

Figure 27: Intervention results under different conceptualization methods using various concept selection criteria. Here, we used JNT + P training strategy for the synthetic dataset.

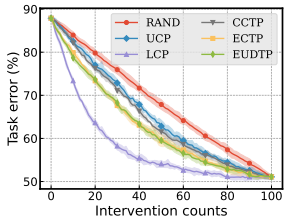
G. More Results on the Effect of Data on Intervention

We find that intervention on data with extremely high input noise or extremely high diversity makes developed concept selection criteria less effective in general with a larger gap from LCP (see Figure 28). Specifically, UCP becomes less effective than other criteria in these cases. We assume that concept prediction uncertainty is rather uncorrelated with concept prediction loss when the concept predictor g achieves very low accuracy.

We also evaluate the effect of concept sparsity levels, *i.e.*, probability of each concept having value 0, using CCTP criterion. Note that intervention becomes less effective as the sparsity level gets closer to 50% as seen in Figure 29a. To understand why, recall that this criterion aggregates the contribution of each concept to the target label prediction. When the sparsity level is high and most concepts have value 0, target prediction is determined by only a few concepts and CCTP can work effectively by first intervening on the concept with the highest contribution. In contrast, as the level gets closer to 50%, target prediction is determined by almost half of the concepts and contribution on target prediction becomes no longer a

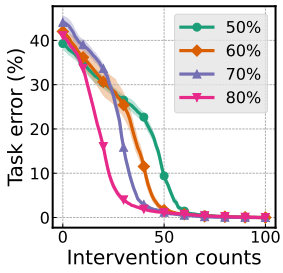


(a) Data with extremely high input noise

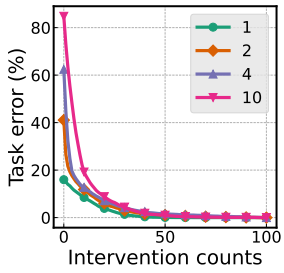


(b) Data with extremely high concept diversity

Figure 28: Intervention results on the data with extremely high input noise (variance of 2.0) or concept diversity (perturbation probability of 30%) respectively. In these cases, the proposed concept selection criteria work less effectively.



(a) CCTP with different concept sparsity levels

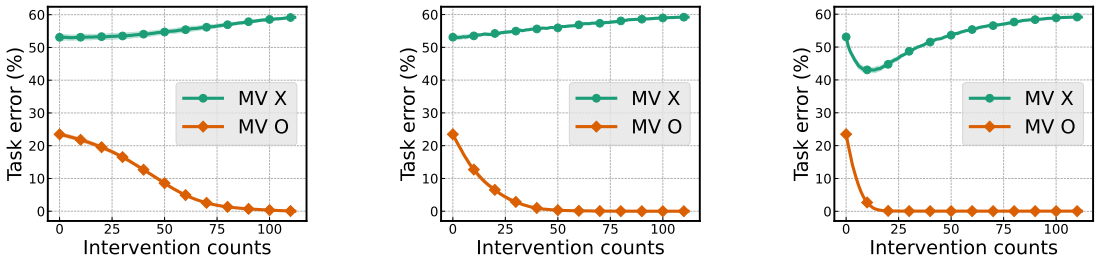


(b) UCP with different sub-group sizes

Figure 29: (a) CCTP becomes more effective with a higher concept sparsity level. (b) Final task error increases, but intervention becomes more effective with larger sub-group sizes.

discriminative feature of the concepts, thus decreasing the effectiveness of the criterion. Furthermore, we observe that the final task error increases but intervention becomes more effective with a large sub-group size γ (see Figure 29b). Specifically, we need 12 intervention counts to decrease the task error by half for the data with $\gamma = 1$, but correcting 5 concepts achieve the same effect for $\gamma = 10$. This is because intervention can decrease the task error much faster for mis-classified examples by distinguishing from similar classes when γ is large.

H. More Results on Fairness of Majority Voting



(a) RAND (b) UCP (c) LCP

Figure 30: Comparison of test-time intervention results with and without using majority voting.

When we do not use majority voting on the CUB dataset, intervention rather increases the task error as seen in Figure 30. Specifically, intervention does not decrease task error at all with RAND, UCP. Even with LCP criterion, intervention does not reduce the task error as much as when we use majority voting, and the error rather starts to increase after about 10 concepts intervened. See Appendix B for the training details.