Human Uncertainty in Concept-Based AI Systems

Katherine M. Collins
University of Cambridge
United Kingdom
kmc61@cam.ac.uk

Matthew Barker*
University of Cambridge
United Kingdom

Mateo Espinosa Zarlenga*
University of Cambridge
United Kingdom

Naveen Raman*
University of Cambridge
United Kingdom

Umang Bhatt
University of Cambridge
Alan Turing Institute
United Kingdom

Mateja Jamnik
University of Cambridge
United Kingdom

Ilia Sucholutsky
Princeton University
United States

Adrian Weller
University of Cambridge
Alan Turing Institute
United Kingdom

Krishnamurthy (Dj) Dvijotham
Google DeepMind
United States

ABSTRACT

Placing a human in the loop may help abate the risks of deploying AI systems in safety-critical settings (e.g., a clinician working with a medical AI system). However, mitigating risks arising from human error and uncertainty within such human-AI interactions is an important and understudied issue. In this work, we study human uncertainty in the context of concept-based models, a family of AI systems that enable human feedback via concept interventions where an expert intervenes on human-interpretable concepts relevant to the task. Prior work in this space often assumes that humans are oracles who are always certain and correct. Yet, real-world decision-making by humans is prone to occasional mistakes and uncertainty. We study how existing concept-based models deal with uncertain interventions from humans using two novel datasets: UMNIST, a visual dataset with controlled simulated uncertainty based on the MNIST dataset, and CUB-S, a relabeling of the popular CUB concept dataset with rich, densely-annotated soft labels from humans. We show that training with uncertain concept labels may help mitigate weaknesses of concept-based systems when handling uncertain interventions. These results allow us to identify several open challenges, which we argue can be tackled through future multidisciplinary research on building interactive uncertainty-aware systems. To facilitate further research, we release a new elicitation platform, UElic, to collect uncertain feedback from humans in collaborative prediction tasks.

KEYWORDS

human-in-the-loop, interactive, uncertainty, concept learning, XAI

ACM Reference Format:
Katherine M. Collins, Matthew Barker, Mateo Espinosa Zarlenga, Naveen Raman, Umang Bhatt, Mateja Jamnik, Ilia Sucholutsky, Adrian Weller, and Krishnamurthy (Dj) Dvijotham. 2023. Human Uncertainty in Concept-Based AI Systems. In *AAAI/ACM Conference on AI, Ethics, and Society (AIES '23), August 08–10, 2023, Montréal, QC, Canada.* ACM, New York, NY, USA, 21 pages. https://doi.org/10.1145/3600211.3604692

1 INTRODUCTION

Human-in-the-loop machine learning (ML) systems are often framed as a promising way to reduce risks in settings where automated models cannot be solely relied upon to make decisions [54]. However, what if the humans themselves are unsure? Can such systems robustly rely on human interventions which may be inaccurate or uncertain? Concept-based models (e.g., Concept Bottleneck Models (CBMs) [31] and Concept Embedding Models (CEMs) [14]), are ML models that enable users to improve their predictions via feedback in the form of human-interpretable "concepts", as opposed to feedback in the original feature space (e.g., pixels of an image). For instance, a radiologist can identify concepts like lung lesions or a fracture to aid a model which uses chest X-rays to predict diseases. Such human-in-the-loop systems typically assume that the intervening human is always correct and confident about their interventions; a so-called "oracle" whose predictions should always override those of the model (see Figure 1A). Yet, uncertainty is an integral component of the way humans reason about the world [5, 16, 33, 41]. If a doctor is unsure of whether a lung lesion is present, or a human cannot observe a feature in a bird due to occlusion (e.g. the tail of a bird is hidden from view), it may be safer to permit them to express this uncertainty [19, 32, 50]. Human-in-the-loop systems, which can take uncertainty into account when responding to human interventions, may help mitigate the risks of both end-to-end automation and human error (see Figure 1B).

Just as machines "knowing when they don't know" has been emphasized for reliability [2, 21, 35, 43], we emphasize empowering humans to express when they do not know as a way to improve trustworthy deployment and outcomes. Recent works have demonstrated the benefits of incorporating uncertainty over label spaces on predictive performance [9, 10, 22, 39, 46, 49, 56], including by

[*] Contributed equally (ordered alphabetically by last name).

AIES '23, August 08–10, 2023, Montréal, QC, Canada

ACM ISBN 979-8-4007-0231-0/23/08.
https://doi.org/10.1145/3600211.3604692

combining human and model predictions [28, 55]; we continue this tradition in the space of concept-based *feedback*. Specifically, our contributions can be summarized as follows:

- We introduce the safety-critical problem of human uncertainty in interactive, concept-based models.
- We reveal failure modes of existing concept-based models when handling user uncertainty over concepts.
- We empirically demonstrate the value of training with uncertainty as a mitigation strategy for better handling test-time uncertainty.
- We develop `UElic`, an extensible platform to facilitate collection of rich, real-world human uncertainty over concepts.
- We use `UElic` to curate a novel relabeling of `CUB` (called `CUB-S`) designed to address limitations in the present dataset. Furthermore, we illustrate how `CUB-S` can serve as a challenge dataset to study uncertain human interventions.

2 PRIMER ON CONCEPT-BASED SYSTEMS

In this section, we introduce concept-based models and discuss how their design enables concept interventions. Concept-based models use human-interpretable values (concepts) as intermediate representations when predicting a task label [31]. An aim of such models is to improve the interpretability of the outputs and facilitate human interventions which correct model mistakes [1, 6, 7, 12, 14, 31].

2.1 Notation

We consider the supervised case where each datapoint consists of (input $\mathbf{x} \in \mathcal{X}$, concepts $\mathbf{c} \in \mathcal{C}$, output $\mathbf{y} \in \mathcal{Y}$). Typically, concepts $\mathbf{c} = [c_1, c_2, \cdots, c_k]^T$ are binary (indicating that concept is "on" or "off"; e.g., oedema is or is not present), or categorical (e.g., different wing colors). Notice, however, that categorical concepts can be converted into binary concepts (e.g., wing color is or is not blue). Typically, concept presence is annotated as being "on" or "off" ($c_i \in \{0, 1\}$); however, there may be *uncertainty* over a concept's presence, which necessitates a continuous value. For that reason, in this work, we let concepts live $\in [0, 1]$, representing $p(c_i|x)$.

2.2 Models

Concept-based models predict the concepts from an intermediate layer in a neural network. Although a plethora of such systems have been developed [14, 31, 36, 42, 60, 62], in this work we focus on Concept Embedding Models (CEMs) [14] as they represent a recent extension of the popular Concept Bottleneck Models (CBMs) [31].

CBMs learn two mappings, one from the input to the concepts $g : \mathcal{X} \rightarrow \mathcal{C}$, and another from the concepts to the outputs $f : \mathcal{C} \rightarrow \mathcal{Y}$. The overall prediction is given by:

$$\hat{y} = f(\hat{\mathbf{c}}) = f(g(\mathbf{x})) \tag{1}$$

There are many ways of learning g and f; here, we focus on the joint bottleneck, which learns g and f at the same time, simultaneously minimizing the concept prediction loss and the output prediction loss. In this work we focus on CBMs with sigmoidal activations in their concept layers whose output can be interpreted as a concept's probability of activation. CEMs further extend CBMs by learning supervised embeddings for each concept, representing concepts as high dimensional vectors while still learning to predict their values as an intermediate step [14]. This allows CEMs to better leverage their capacity when trained on datasets with an "incomplete" set of concept annotations [14, 61]. We use **training with uncertainty** to refer to models trained with concepts represented as probabilities $\in [0, 1]$, rather than as binary concepts. The target y is left unchanged in this work.

2.3 Interventions

A prime motivation for employing concept-based systems is the ease of intervenability. If a user notices that the model is predicting a concept incorrectly (e.g., the X-ray scan shows bone spurs, yet the model predicted no bone spurs), a user (e.g., a medical professional) can directly edit said concept to (potentially) update the prediction. This involves updating a predicted concept $\hat{c}_i$ with the concept value returned by the human $\hat{c}_i \leftarrow c_i$ and recalculating our prediction $\hat{y} = f(\hat{\mathbf{c}})$. Because these interventions edit the *model's predicted probability of a given concept*, we can readily permit the user to edit *with their own predicted probability of that concept.* When we refer to **testing with uncertainty**, we let the human-edited concept be a probability, analogous to our "training with uncertainty" setting.

Coupled with the ease of intervenability is the notion of an **intervention policy**, an algorithm that selects the next concept to query a human user given a set of previously provided concept labels. Such policies are sensible to employ in practice when it is costly to query a human to intervene and when one wishes to maximise the impact that a single intervention may have on the model's performance [6]. In this work, we consider two policies to select the concept to intervene on: 1) *Random*: selecting the next concept to query randomly of concepts, and 2) *Skyline:* an approximate oracle policy following Chauhan et al., which selects the next concept to query that will best impact performance (as if it were possible to know, simulating an upper bound on intervention efficacy; see Supplement for further details). While other works have been developed with more advanced policies [6, 52, 53], we select Random and Skyline because they illustrate the *bounds* on achievable performance; Random being the most naive policy and Skyline being the optimal policy. Unless otherwise noted, concepts are chosen via Random.

2.4 Critiques and Common Assumptions

Concept-based models, and the broader ecosystem in which they are deployed, have been shown to exhibit information leakage [37] or impurities distributed across concept representations [15], spurious input saliency maps [38], bloated, hard-to-learn concept definitions [47], and propensity to be influenced by correlations amongst concepts [20]. To our knowledge, we are the first work which directly considers *uncertainty in the human user* with concept-based models.

3 RESEARCH QUESTIONS

In this work, we address the following research questions:

- **RQ1**: How do existing concept-based systems handle the introduction of human uncertainty at test time?
- **RQ2**: How can systems be bolstered to better support human uncertainty at test time?

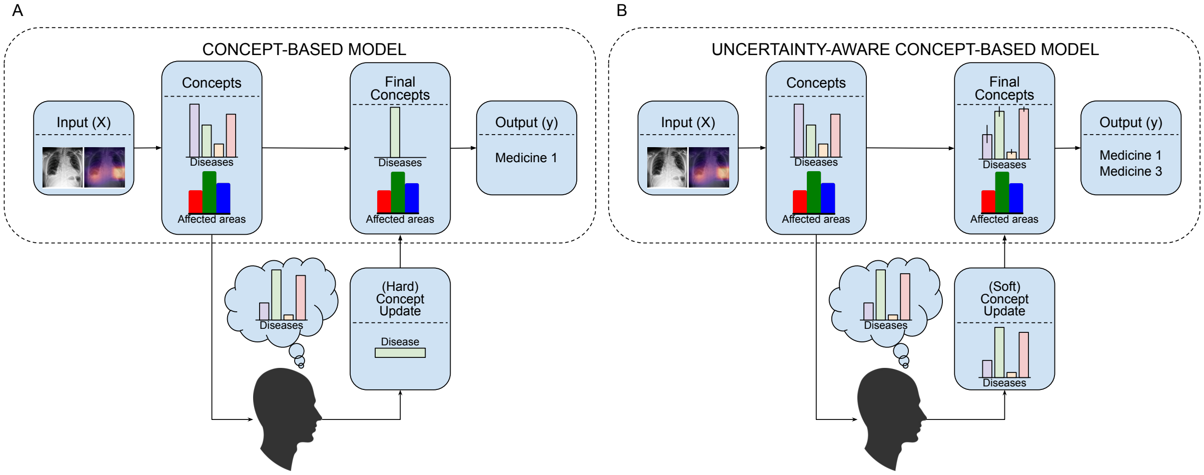


Figure 1: Schematic of uncertainty in test-time interventions in concept-based models. When presented with features and a concept to annotate, a human user may be uncertain. We empower the user to express this uncertainty when intervening on concepts: to make the concept-based systems *aware of their uncertainty*. We demonstrate the set-up in a hypothesized safety-critical setting of medical diagnosis; X-ray images are depicted from CheXpert [24].

- **RQ3**: How does the level and form of uncertainty (e.g., whether the uncertainty is expressed through discrete labels, or rich, continuous probabilities) impact performance?

These questions are important when assessing the receptiveness of concept-based systems to interventions from humans in the real-world who may not be oracles, and may wish to express some uncertainty over their concept edits. We investigate such questions across a *spectrum of degrees and forms of uncertainty*. First, we study controlled, simulated uncertainty in UMNIST, our newly proposed addition dataset based on the MNIST handwritten digit recognition, as well as over the popular medical dataset CheXpert [24]. We then depart from considering simulated uncertainty – moving to the real, human-elicited in-the-wild uncertainty; first, coarse-grained uncertainty scores collected in CUB [58], and then richer uncertainty which we collect in our new real-world dataset of human uncertainty: CUB-S.

For each dataset, we study the test-time performance of models trained on binary, certain concepts, but faced with uncertainty at test-time. Then, we explore how this performance is affected when the same models are trained with uncertainty estimations in concept labels.

4 SIMULATED UNCERTAINTY

We first investigate concept-based models on simulated uncertainty.

4.1 Experimental Set-Up

4.1.1 Data. We consider two datasets with varying degrees of simulated uncertainty: CheXpert and a newly constructed, controllable dataset of uncertainty, UMNIST. CheXpert is a visual dataset containing chest X-rays that are annotated with a set of 14 concepts. In this task, we aim to predict the "No Finding" concept based on the other 13 concepts. We incorporate simulated uncertainty by each concept's label by setting uncertain values to 0.5 and unknown values to 0 (CheXpert comes with annotations indicating which concepts are uncertain/unknown). In contrast, UMNIST's samples are formed by a mixture of MNIST digits where the task is to compute the sum of all digits and each sample is given the number represented by each digit as a concept annotation. For simplicity, in this work we use zeros or ones only as the possible numbers each digit may take even though this dataset can be easily generalised to more options per digit (see Supplement for more details). UMNIST is parameterized with parameter $\delta \in [0, 1]$ which controls the amount of uncertainty/noise in each sample's concept annotations. Intuitively, $\delta = 0$ represents fully certain concept labels and no mixing of each sample's digits while $\delta = 1$ represents a dataset with random concept annotations. We apply such uncertainty level in the UMNIST dataset by performing a random mixture of a digit in correspondence to its assigned uncertainty label. For example, if a concept's label is set to 0.75, then the digit it represents may be an image whose 75% of its pixels come from an image of a "one" digit while the reaming pixels come from an arbitrary image of a "zero". The same δ-smoothing is used in CheXpert, without the sample image mixing, to produce noisy concept annotations by mixing *concept labels* only (as it is unclear how to mix sample images based on a given concept).

4.1.2 Evaluation. We study the performance of the concept-based systems on the task of interest (e.g., abnormality detection in chest X-rays and predicting the sum of digits in an image) as a function of the number of concepts intervened. For CheXpert, following Chauhan et al., we evaluate the Area under the ROC curve (*AUC*). For UMNIST, given its multi-class setting, we evaluate accuracy instead. Finally, as we are interested in how uncertain interventions affect concept-based models rather than how to best take into account uncertainty at intervention time, an interesting yet different research question, in our evaluation we randomly choose which concepts to intervene on rather than deploying more principled intervention policies.

4.2 Intervening with Uncertainty

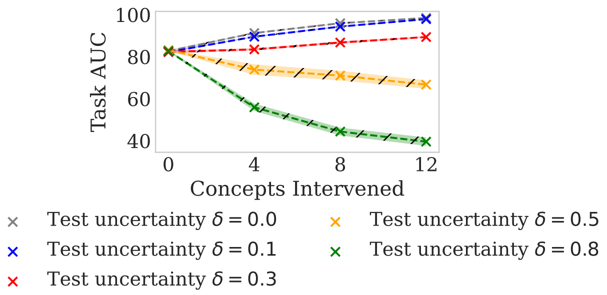


Figure 2: Effect of simulated uncertainty in `CheXpert` on test-time efficacy (AUC) in CME as the number of concepts intervened on increases. Standard error depicted over three random seeds.

We first benchmark how well models *not trained with uncertain concepts* cope with uncertainty at intervention during testing. This setting best captures what a user facing uncertainty may experience when deploying pre-trained concept-based models, which are rarely trained on uncertainty. Specifically, we explore varying the total amount of concept uncertainty in the testing data and observe that, even with low simulated uncertainty, both CEMs and CBMs suffer from significant drops in intervention performance when dealing with uncertain samples (see Figures 2 and 3; see Supplement). We can see that this drop is particularly sharp as the amount of uncertainty grows, as seen in the performance of CEMs when $\delta \in \{0.4, 0.6\}$. This suggests that these models, although accurate and high performing when receiving fully certain interventions, cannot generalize to settings where the intervening user is uncertain of the nature of some of the concepts, showing the surprising brittleness of these models in the face of uncertainty. Finally, we note that although one would expect a model's performance to drop when intervening with uncertainty, the observed drops in Figures 2 and 3, and later also seen in the bottom row of Figure 3 (right), are significantly sharper than what one would intuitively expect, bringing attention to the need to further explore this phenomenon. This can be seen by noticing that even slightly uncertain interventions (e.g., when the test uncertainty is set to $\delta = 0.2$) result in significant drops in performance.

4.3 Training with Uncertainty Can Improve Robustness

While we observe that exposing models not trained with concept uncertainty to uncertain concepts leads to the breakdown of intervention efficacy — we hypothesize that training with uncertainty can boost the ability of these models to cope with uncertain interventions. This hypothesis spans from previous results in knowledge distillation [22] and adversarial training [18] suggesting that, by injecting perturbations to the model's target labels during training, a model's robustness to small changes in its inputs (in this case in the concepts being intervened) may be improved. In Figures 3 (Right) and 2 we indeed observe that by *training* with uncertainty, we can salvage the efficacy of interventions – particularly under *distribution shift* (see, in particular, Figure 3 (Right) when test uncertainty level is set to $\delta = 0.4$). These results suggest that, if we train on an uncertainty level that differs from the level expressed by a user, we may be better equipped to handle that user's uncertainty than if we did not train with uncertainty at all. Notably, however, we observe a "sweet spot" in the level of uncertainty that is helpful to the model.

4.4 Implications

Even in controlled settings, existing concept-based systems struggle to handle concept uncertainty at inference-time adequately. Training with concept uncertainty may prove a reasonable salve for capturing value from the uncertain interventions, particularly affording robustness under distribution shifts. However, our results suggest that training with too much concept "softness" can be harmful.

5 REAL HUMAN UNCERTAINTY

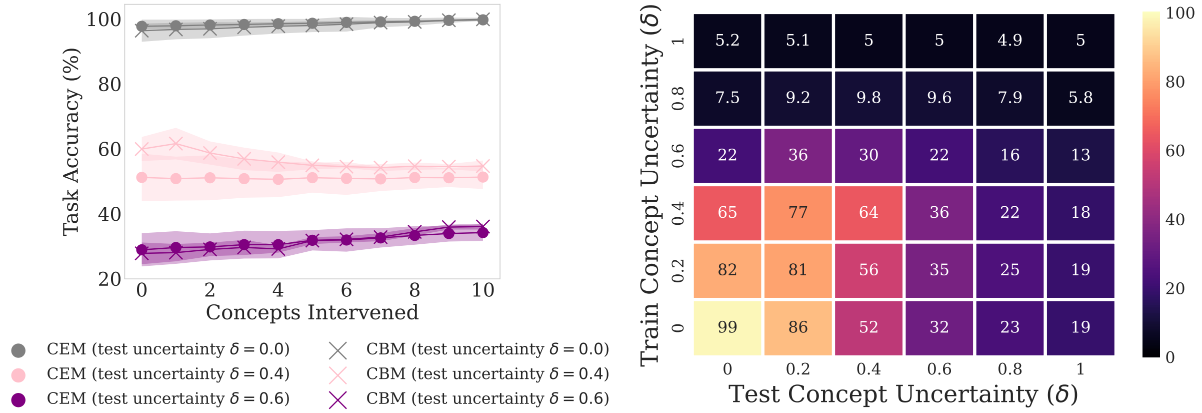
We see in our simulations that exposing models to test-time uncertainty can impact performance and training with uncertainty offers a potential remediation strategy to handle such test-time uncertainty. However, these investigations are on contrived uncertainty: how do existing systems fare with real-world uncertainty?

5.1 Taxonomy of Forms of Uncertainty

Real human uncertainty can come in many forms. This uncertainty may be **epistemic**, representing lack of knowledge, or **aleotoric**, due to (potentially) inevitable randomness [23]. Further, this uncertainty can either be **heteroschedastic**, i.e., dependent on the input, or **homeoschedastic**, independent of the input [48]. Thus far, we have focused on *regular* uncertainty – simulating the same level of uncertainty δ across all concepts.

However, in-the-wild uncertainty, elicited from humans, is not so simple. The method by which uncertainty is elicited can have a sizeable impact on the quality of the elicitation [17, 27, 41, 44]. As researchers may use a variety of elicitation practices, we believe it is **important to understand how concept-based systems handle different forms of elicited human uncertainty**.

We focus on two flavors of uncertainty **coarse-grained** (elicited from a few-option discrete scale) and **fine-grained** (probabilities extracted over each possible attribute in a concept group). In the coarse-grained setting, humans provide both binary concept annotations, $c_i \in \{0, 1\}$, and a discrete measure of confidence ω, e.g., $\omega \in \{$"Guessing", "Probably", "Definitely"$\}$. In this setup, we need to construct a map from $c_i \times \omega$ to the probability distribution of interest $p(c_i|x)$. In contrast, in the fine-grained setting, humans directly provide $p(c_i|x)$. While we do not consider *all* forms of uncertainty expression, e.g., humans may prefer to express uncertainty flexibly through language [11, 64], we see our study as a promising first step into a deeper investigation of the impact of different forms of *real human uncertainty* on concept-based system performance.



	0	0.2	0.4	0.6	0.8	1
1	5.2	5.1	5	5	4.9	5
0.8	7.5	9.2	9.8	9.6	7.9	5.8
0.6	22	36	30	22	16	13
0.4	65	77	64	36	22	18
0.2	82	81	56	35	25	19
0	99	86	52	32	23	19

Figure 3: Left: Mean test accuracies of random interventions on CBMs and CEMs, together with their standard error computed across five different random initializations, as we increase the number of concepts we intervene on. Concept-based systems (CBMs and CEMs) that have not been trained on uncertainty struggle to handle test-time uncertainty, even when both models achieved similarly high concept accuracies. We note that as opposed to our results in Figure 2, we observe different accuracies when no concepts are intervened when we vary δ. This is because the sample images in this dataset, and not just the concept labels, are mixed as a function of δ. Right: Heatmap showing the task accuracy (%) of a CEM trained in UMNIST (with train-time δ varying across the y-axis) after intervening in 50% of its concepts with possibly uncertain test-time concept labels (controlled by the test dataset's δ value in the x-axis). Training with uncertainty in UMNIST improves robustness under distribution shift at intervention time (compare bottom row when test time $\delta \in \{0.4, 0.6\}$ vs CEMs trained with samples generated with $\delta \in \{0.2, 0.4\}$), provided the training level of uncertainty is not too high.

5.2 Coarse-Grained Uncertainty

We first consider these questions over *coarse-grained* human uncertainty; i.e., a single discrete annotation indicating user uncertainty. The limited, discrete nature of the uncertainty variable ω raises important design considerations when considering how to use the score. For instance, if a user marks that they are uncertain, how can we know *how* uncertain are they? And are they only uncertain over parts or the entirety of the concept space? We next study how design choices to impute these ambiguities at train- and test-time can impact the intervention efficacy. We address these questions through the uncertainty annotations provided in [58].[1]

5.2.1 *Experimental Set-Up.*

Data. CUB is a highly popular benchmark dataset for concept prediction that includes images of birds, annotated with 28 different concept groups (e.g., wing color, beak shape) [58]. Each concept can take on many different values. The task is to predict one of two hundred different bird species. Wah et al. elicited humans' uncertainty when collecting the original annotations; however, these annotations are highly coarse (a simple: “Guessing,” “Probably”, or “Definitely” mark over each concept group's annotations). There are 311 total binary concepts that can be extracted from the 28 categorical concepts; we follow the common practice proposed in Koh et al. by filtering these down to 112 concepts. We study how intervening with, and learning with, these coarse-grained annotations impacts performance.

Evaluation. We follow similar evaluation protocols to our Simulated Uncertainty experiments, focusing on the measure of task accuracy (where the task is bird species classification). We include Skyline interventions to help demonstrate the best possible intervention policy that can be achieved to further highlight the impact of the types of uncertainty on performance.

5.2.2 How to Use Discrete Uncertainty Scores? The first question raised with the real-world uncertainty of the form elicited in CUB is how to leverage the scores at intervention time. Uncertain annotations are only provided in the form of a single, discrete measure of uncertainty: CUB annotators provided *coarse-grained*, discretized approximations of their confidence in said annotations (i.e., specifying whether they were Guessing, Probably Sure, or Definitely Sure in their annotations).

However, concept-based systems typically necessitate interventions to be specified in continuous space; as such, we need to define a custom mapping from discretely expressed uncertainty to continuous values. The choice of such a mapping impacts downstream performance. Second, for categorical concepts like those in CUB, a single measure of uncertainty does not permit a nuanced assignment of uncertainty to individual concepts. There is ambiguity around what the human user intended to express; i.e., if the user says they are “Probably” we do not know over which concepts and *how* unsure they are. We highlight the ramification of this ambiguity in two ways. First, we demonstrate that imputing the coarse-grained uncertainty with different continuous values can – at times

[1] We include analyses over the “real” coarse-grained uncertainty annotations in CheXpert in the Supplement. In contrast to CUB, which has uncertainty annotations for each image and attribute, only some concepts are labeled with an uncertainty score in CheXpert. Moreover, the score obfuscates whether the label is deemed uncertain due to human uncertainty versus annotation-scraping uncertainty [24].

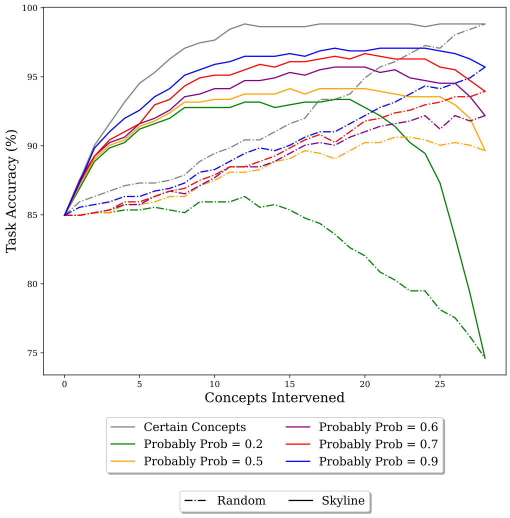


Figure 4: Impact of different levels of uncertainty on intervention efficacy (task accuracy) in CEMs as the number of concepts intervened on increases, across both Random and Skyline policies. Colors correspond to different intervention-time imputations of the probability someone may intend when they say they are "Probably" sure. Mean performance when intervening over all test set examples in CUB.

dramatically – impact performance. Second, we demonstrate that the degree of softness assumed when leveraging uncertainty over *categorical* concept spaces matters.

Imputing the "Probably" Probability. We focus on the concept annotations where humans expressed they were "Probably" sure of the annotations. Here, we do not know *how* certain the annotators were in their labeling. We vary the level of uncertainty we assume annotators were in such annotations when intervening and apply the same imputed probably to the "on" (e.g., blue wing present) and "off" concepts (e.g., wing color is not yellow); for the latter, we flip the assigned probability. We observe in Figure 4 that the imputed probability can have a dramatic impact. The imputation matters – demonstrating both limitations of insufficient richness in annotation (we do not know what the original annotators intended) and further brittleness of these systems to test-time uncertainty when they have been trained exclusively on deterministic concepts.

Distribution of Uncertainty over Categorical Concepts. Not only does insufficient richness in uncertain annotations pose a challenge when determining what level of certainty to assign: it is also ambiguous *which* concepts the annotator was uncertain in when they said they were "probably" sure. We refer to this phenomenon as whether the annotator's uncertainty is **broad** (over all possible concept values) or **narrow** (just over a few of the possible concept values). For instance, when annotating beak shape, the annotator may be very certain the shape is not rounded – but unsure whether to classify the shape as dagger-like or pointed: "narrow" uncertainty. In that case, the intervention on rounded should be left fully "off" (i.e., 0%), but the mass should be spread on the possible "on" values (perhaps 70% dagger, 30% pointed). We demonstrate in Figures 5 and in the Supplement that these choices also matter. Assuming that an annotator's uncertainty is broad, and only over aspects of the concept space, can substantially impair intervention quality (likely because the converse was oversmoothing – i.e., falsely miscalibrating to be underconfident). The sensitivity of the models and policies to these varied degrees of uncertainty highlights the brittleness of systems to such design choices and possible spectra of human uncertainty expression.

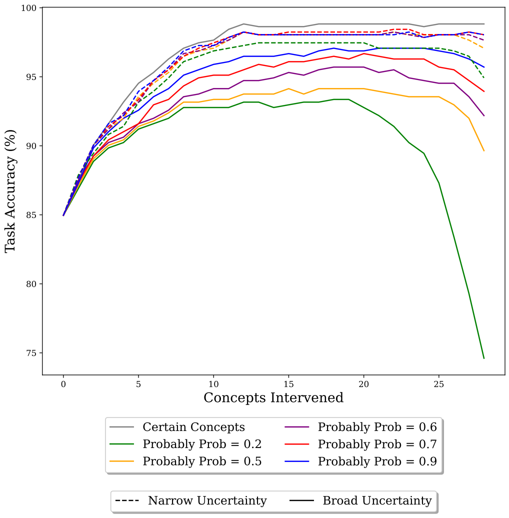


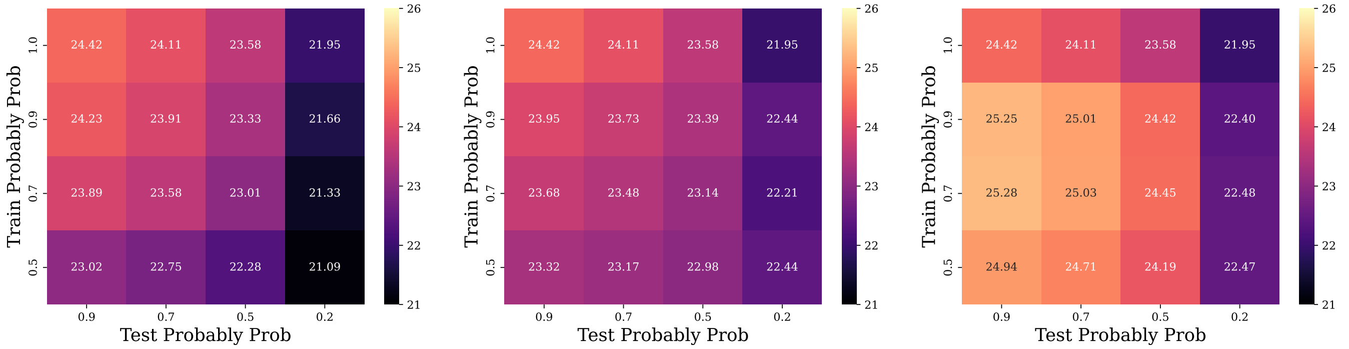
Figure 5: Impact on task accuracy of different ways of distributing the discrete uncertainty over categorical concept groups selected using Skyline.

5.2.3 *Instance- vs. Population-Level Uncertainty?*

Another question raised by in-the-wild human uncertainty is how to handle individual differences versus group-level uncertainty [10, 46]. This question is particularly pertinent in CUB, as the annotations are both sparse and noisy. Several concepts have few annotations, and many annotations may be low-quality. As such, it may make sense to consider *population-level* uncertainty rather than individual uncertainty. Here, we refer to population-level uncertainty as the class-level labels used by Koh et al.. We form soft labels by aggregating all annotators' individual-level soft labels for a given category. To "upper bound" the differences in population vs. individual-level uncertainty, we consider the possibility that annotators are unsure over *both* "on" and "off" annotations (i.e., that they possess broad uncertainty). We see that whether or not to intervene with population-level uncertainty matters — test-time performance is markedly higher when using population-level labels (see Supplement).

5.2.4 *Training with Uncertainty.*

Likewise, the question of the form of uncertainty and whether or not to leverage aggregate uncertainty matters at train time. Training on aggregated uncertainty not only performance on similarly population-level uncertainty (see Supplement), but also over softer, potentially noisier individual-level



(a) Training on instance-level (broad) uncertain concept annotations.

(b) Training on instance-level (narrow) uncertain concept annotations.

(c) Training on population-level (broad) uncertain concept annotations.

Figure 6: Training on uncertain concept labels improves generalization to instance-level (broad) uncertainty at test-time – the most challenging of the in-the-wild coarse-grained varieties. Heatmap colors depict generalization efficacy operationalized as the AUC between the intervention-accuracy curve. Uncertainty here is expressed by varying the imputed "Probably" probability at train and test time; decreasing probability (e.g., left-to-right on the x-axis) corresponds to increasing uncertainty.

uncertainty – across gradations of uncertainty (see Figure 6). Further, whether uncertainty is assumed to be broad or narrow at an individual-level can also impact training label efficacy (see Left and Middle panels of Figure 6).

5.2.5 Implications. While we focus here on CUB – as the dataset is a highly popular concept benchmark, and therefore necessary to deeply understand – the elicitation of discrete uncertainty is lightweight and popular in crowdsourcing [41] (see further investigations with CheXpert in the Supplement); as such, our investigations may be broadly applicable to researchers leveraging elicited discrete uncertainty. The wide impact design choices can have serves as a caution – if we want safe systems which are robust, we ought to be able to handle the array of intended meaning expressed by humans through discrete uncertainty. Decisions around how to treat discrete uncertainty over concepts persist across train- and test-time.

5.3 Fine-Grained Uncertainty

Next, we turn to more fine-grained uncertainty. When faced with many options (e.g., multiple different possible colors for the wing, or different gradations of severity in a medical phenotype), a human may have different levels of uncertainty over each option. We now consider this form of categorical uncertainty *explicitly*, rather than inferring from an ambiguous single measure of "uncertainty."

However, there is a paucity of datasets available with such richly annotated labelings over concept space. As such, to facilitate this research, **we build a new platform for uncertainty elicitation over concepts**, which we call UElic and offer a first application of UElic to relabel a subset of CUB with human soft labels over all concepts. **We release our dataset as CUB-S, replete with nearly 5,000 rich uncertainty-labeled concept groups**.

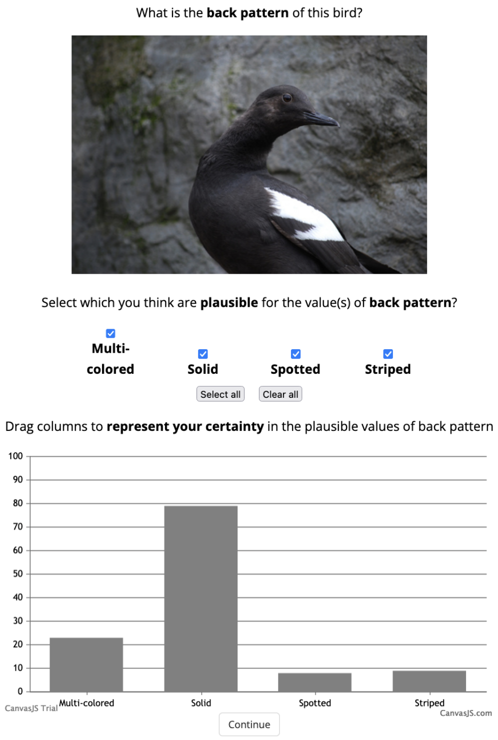
In this Section, we begin by introducing our new elicitation interface for rich human uncertainty and offer several insights into the character of the elicitations. We then highlight how concept-based systems crumble under the nuances of the fine-grained uncertainty we elicit. We believe CUB-S can serve as a formative dataset to further study human uncertainty in concept-based models.

5.3.1 Eliciting Human Uncertainty. We offer a new platform to streamline the elicitation of human uncertainty. Our interface, UElic, offers a lightweight paradigm for users to express uncertainty. Users are presented with the features of interest (e.g., an image), the concept to be annotated, and all available options. To reduce the cognitive load of expressing uncertainty over *all* options per concept, we request users select only the attributes they think are plausible and express their uncertainty over these attributes by dragging an interactive bar chart to represent their perceived probability, inspired by [17]. An example interface screen is depicted in Figure 7.

5.3.2 Collecting CUB-S. We recruit 89 participants from the crowdsourcing platform, Prolific [45]. Participants annotate *all 28 concepts* for two different bird images: totalling **4984 soft categorical concept group annotations**[2]; concept order is shuffled for each participant to control for order effects. Within each soft concept group annotation, participants provide their uncertainty over each of the possible attributes for that concept (e.g., possible wing colors, beak shapes, etc). Stimuli are selected from the CUB test set. While two images is a small sample size per individual, we selected the number to avoid cognitive fatigue, as we wanted participants to annotate all concept groups for a given bird image, permitting rich exploration at inference-time mimicking real-world cases where a single user would likely interact with the concept-based model. Additional details are included in the Supplement.

5.3.3 Richness in CUB-S. Our elicited soft labels demonstrate that humans indeed can starkly depart from a uniform distribution of uncertainty over concepts (see Figure 8). Humans possess rich approximations of uncertainty. Eliciting this uncertainty directly can resolve some of the mentioned ambiguity with discrete uncertainty.

[2] All data is included at our repository.



What is the **back pattern** of this bird?

Select which you think are **plausible** for the value(s) of **back pattern**?

☑ **Multi-colored** ☑ **Solid** ☑ **Spotted** ☑ **Striped**

Drag columns to **represent your certainty** in the plausible values of back pattern

Figure 7: Example screen of UElic for CUB. Participants select the concept attributes they think are plausible, and drag bars to express said uncertainty. Here, the back is not visible; users must be uncertain in their annotation. We empower annotators to *richly* express this belief distribution, in contrast to the original CUB dataset.

Further, by tracking which concepts were annotated by particular individuals (information which is not stored in the original CUB annotations), we identify a wide spectrum in the calibration of annotators. This is not entirely unexpected, given different levels of uncertainty calibration in humans broadly [25, 30]. We use the Expected Calibration Error (ECE) [40] as a metric to evaluate the accuracy of annotators when estimating their confidence. Intuitively, the metric is the expected absolute difference between the fraction of correct predictions (accuracy), and the probabilities provided by the annotators (confidence). The "correct" concepts for a given bird are determined from the original CUB annotations averaged over all birds of the same species. These "correct" concepts are a suitable approximation to ground truth, and are significantly less noisy than the CUB-S annotations; however, we emphasize that they are *not definitive ground truth.*

Figure 9 shows that the majority of annotators are reasonably calibrated, although this value is positively skewed by the large number of (correct) zero probabilities provided for rare concepts (such as the color "purple"). There are some annotators who are poorly calibrated and it mitigating this issue remains an open question. Some calibration "error" is a result of the additional richness in the CUB-S annotations not present in CUB. However, there are also genuine annotation errors which we observe when manually checking the annotations. Illustrative examples comparing soft CUB-S annotations to hard CUB annotations are shown in Figure 8. Humans who intervene at test time will also suffer from calibration errors, challenging the common assumption that human experts are perfect "oracles". On average, we observe that annotators consistently underestimate small probabilities but overestimate large probabilities (Figure 10). When several concepts are possible, it is likely that annotators attempt to reduce their cognitive load by only selecting a few to have a nonzero probability. Conversely, when a concept is highly probable, annotators may incorrectly round an annotation to 100 (i.e. absolute certainty). Figure 17 shows that 0 and 100 are the most popular uncertain annotation values, due to the presence of these two effects. We emphasize that some errors are predictable, and thus have the potential to be corrected when training an uncertainty-aware model.

It is unclear whether the poor calibration is a result of our interface, or an unavoidable issue when eliciting uncertainties in a crowdsourcing setting; humans can have limited cognitive resources – they may not be willing to endorse several related concepts (e.g., orange and red), while providing detailed uncertainty over each. However, the fact that we *do* encounter such challenges is an important consideration in the deployment of systems in which *receive* such uncertainty estimates. It is essential that **systems be robust to these nuances and peculiarities in elicited human uncertainty, or else they may fail at deployment time**.

5.3.4 *Intervening at Test-Time with* CUB-S.

We next apply the same computational investigations as in our prior experiments to CUB-S; now, only varying the labels used at test time. We use models trained on population-level broad uncertainty derived from coarse-grained CUB as in the prior section. We find in Figure 11 that the richness of CUB-S poses a substantial challenge for concept-based models. While we find that using models trained on the coarse-grained uncertainty in CUB can mitigate some of the failures under test-time uncertainty, they are not a perfect salve.

The development of better mitigation strategies to handle the nuances of in-the-wild categorical uncertainty over concepts is exciting ground for future work. We observe that some concepts are preferable to elicit interventions over; sometimes human uncertainty is helpful, other times it *harms* model performance (see Supplement). Further, these differences persist across methods of training the models (i.e., the level of uncertainty in the training data, see Supplement), underscoring the need for adaptive, query procedures personalized to individual- and model uncertainty. We argue multi-disciplinary methodological advances to handle in-the-wild, rich human uncertainty over concept annotations are essential.

5.3.5 *Implications.*

Humans interpret and reason in the world with richly structured uncertainty. Our CUB-S elicitation demonstrates this richness. However, we find that concept-based systems struggle to handle this level of richness. Given humans *are capable and do* express fine-grained uncertainty, it is sensible that our systems ought to be equipped to handle the nuances of in-the-wild uncertainty.

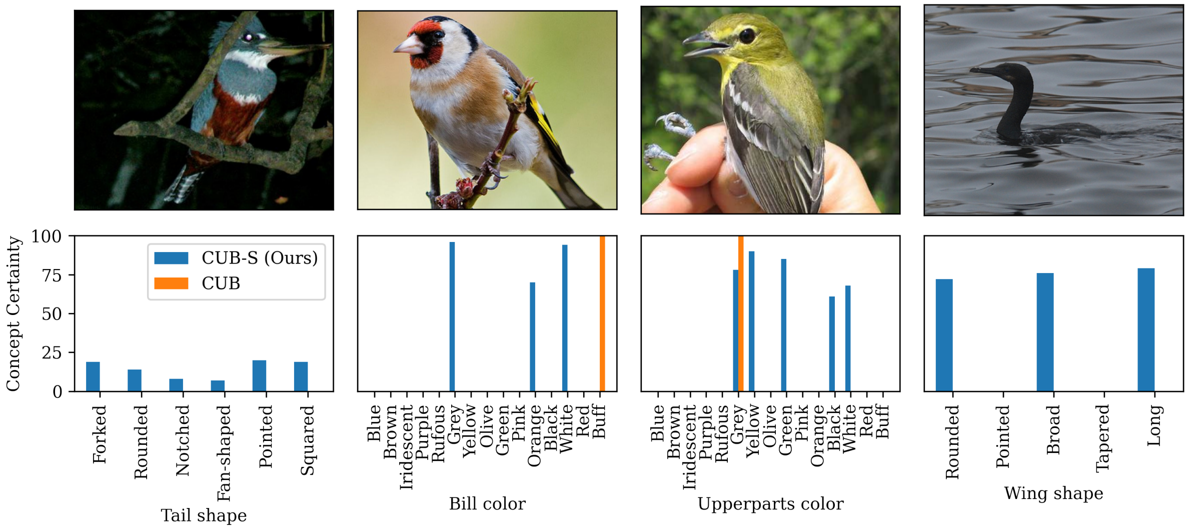


Figure 8: Example soft concept annotations elicited in CUB-S compared to CUB class labels. Far left: well-calibrated annotations for the "tail shape" concept, expressing appropriate uncertainties which sum to 100. Center left: annotations rarely included the obscure "buff" color, even when it was appropriate. Center right: richer annotations for the "upperparts color" provide more information than the certain CUB annotations. Far right: uncalibrated uncertainty of the "wing shape" concept under occlusion.

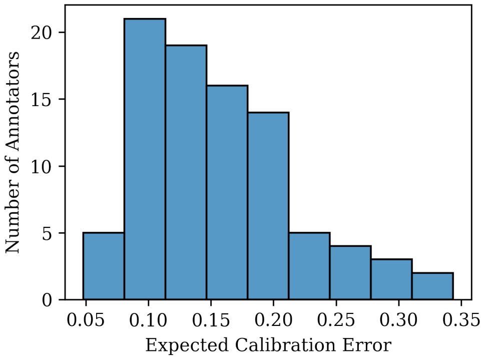


Figure 9: Distribution of Expected Calibration Error for annotators in CUB-S. The positive skew shows most annotators are well-calibrated, with a few who are very poorly calibrated.

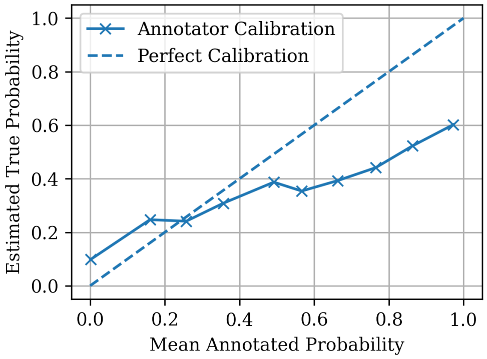


Figure 10: Calibration curve for CUB-S annotators, showing consistent underestimates of small probabilities and overestimates of large probabilities.

6 OPEN CHALLENGES

We emphasize the importance of considering human uncertainty in concept-based models, and the need for richer datasets of human uncertainty to study these challenges. CUB-S is a promising initial playground to study the nuances faced with real human uncertainty[3]. Our work raises several open challenges.

6.1 Complementarity of Human and Machine Uncertainty

Considering human uncertainty in interventions opens up exciting opportunities in the study of human-machine complementarity [3, 4, 28, 55, 59]. When we break the assumption that humans are confident oracles, it becomes especially important to consider whether cases which are hard for the model to annotate are also those that a human struggles with; in that case, selecting such a concept is not ideal. Learning models and intervention policies which complement humans' strengths and weaknesses, accounting

[3] All code and data will be hosted at our repository.

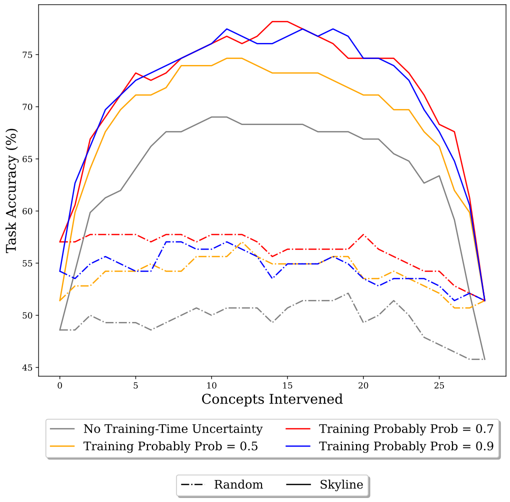


Figure 11: CEMs struggle to handle real human uncertainty. While Skyline is able to leverage *some* signal in the data, not all incorporated concept annotations help model performance: some may hurt. Using models trained on human uncertainty information may mitigate some of the drop.

for their expertise and confidence, are promising grounds for further study with CUB-S and beyond. We see that varied models may prefer different forms of uncertainty (see Supplement); further, even though we see that Random interventions fail disastrously, there is a signal that Skyline picks up on the uncertain concepts – how can we predict where and when to ask people for their uncertainty? And when we do receive their uncertainty, it is not immediately apparent whether we *should* take the human intervention as "truth." As we demonstrate, real humans are *not* consistent oracles – and in some settings (e.g., occlusion), *no* human may be an oracle, even if they are an expert. Models which can learn whether or not to trust human interventions, e.g. [13, 36], are promising grounds for future study.

6.2 Treating Human (Mis)Calibration

A core factor in whether or not to trust a *human's* intervention, and determinant of which concepts to query, may depend on the expected calibration of the user. We observe wide variation in individuals' level of calibration in their uncertainty expression, a finding that resonates with a wealth of cognitive science literature [25, 27, 30, 34, 41, 51, 57]. However, we emphasize that calibration need not be a turn-off from collecting uncertainty in the first place; not only are some humans highly calibrated – but forcing someone to express certainty when they are not (and when it is not possible to be certain; e.g., occlusion), we argue may be worse. Future work for post-hoc calibration in a *few-shot* manner, e.g., from limited individual-level user data, provided in an online fashion, is further promising ground for new methodological advances. Additionally, we encourage further experimentation with UElic to encourage better calibration from humans – perhaps through the use of a carefully designed teaching curriculum [26, 27]. We see calibration – particularly across real users with varying domain expertise [12] – as an exciting nexus for a multi-disciplinary study spanning ML, cognitive science, UX design, and psychology.

6.3 Scaling Uncertainty Elicitation

Further, we recognize that the annotation of large-scale datasets with human uncertainty may be practically challenging. It is costly to elicit human uncertainty: annotators take substantially more time [10]. There is a need for more scalable elicitation techniques, and better simulators of human uncertainty to permit the study of softness at train time. We observe substantial differences in model performance depending on the form of uncertainty used; more data is needed to further characterize these differences and determine when one form of uncertainty is better to elicit than another, such that when we deploy systems in the world – they can handle a variety of forms of uncertainty expression.

7 CONCLUSION

We highlight the importance of considering human uncertainty in concept-based models to improve reliable performance for safe applications in deployment across society. Humans in the real-world are not certain oracles. We make mistakes and may be unsure. Even though humans may be miscalibrated in their uncertainty, we believe the study of tools to elicit and work with human uncertainty has great potential to improve human-in-the-loop systems. Through a mixture of simulated and in-the-wild experiments with uncertainty, we demonstrate failure modes of popular concept-based systems to handle both coarse- and fine-grained uncertain feedback. We offer a new interface, UElic, and a new challenge dataset, CUB-S, to support further study into human uncertainty in interventions. Modeling human uncertainty at train- and test-time has the potential to greatly improve the reliability and trustworthiness of concept-based models when deployed safely in the wild.

ACKNOWLEDGMENTS

We thank (alphabetically) Allison Chen, Yanzhi Chen, Carl Henrik Ek, Kris Jensen, Isaac Reid, Deepak Ramachandran, Josh Tenenbaum, and Richard Turner for helpful conversations. We also thank Steve Branson, Catherine Wah, Kushal Chauhan, and Rishabh Tiwari for helpful clarifications on their work. Thank you to the participants from Prolific who took part in our annotation. We also thank the reviewers for their incredibly useful feedback.

KMC gratefully acknowledges support from the Marshall Commission and the Cambridge Trust. MEZ acknowledges support from the Gates Cambridge Trust via the Gates Cambridge Scholarship. NR is supported by a Churchill Scholarship. UB acknowledges support from DeepMind and the Leverhulme Trust via the Leverhulme Centre for the Future of Intelligence (CFI), and from the Mozilla Foundation. MJ is supported by the EPSRC grant EP/T019603/1. AW acknowledges support from a Turing AI Fellowship under grant EP/V025279/1, The Alan Turing Institute, and the Leverhulme Trust via CFI. IS is supported by an NSERC fellowship (567554-2022).

REFERENCES

[1] Pietro Barbiero, Gabriele Ciravegna, Francesco Giannini, Pietro Lió, Marco Gori, and Stefano Melacci. 2022. Entropy-based logic explanations of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 6046–6054.

[2] Umang Bhatt, Javier Antorán, Yunfeng Zhang, Q Vera Liao, Prasanna Sattigeri, Riccardo Fogliato, Gabrielle Melançon, Ranganath Krishnan, Jason Stanley, Omesh Tickoo, et al. 2021. Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 401–413.

[3] Elizabeth Bondi, Raphael Koster, Hannah Sheahan, Martin Chadwick, Yoram Bachrach, Taylan Cemgil, Ulrich Paquet, and Krishnamurthy Dvijotham. 2022. Role of Human-AI Interaction in Selective Prediction. In *AAAI*, Vol. 36. 5286–5294.

[4] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. 2010. Visual recognition with humans in the loop. In *European Conference on Computer Vision*. Springer, 438–451.

[5] Nick Chater, Jian-Qiao Zhu, Jake Spicer, Joakim Sundh, Pablo León-Villagrá, and Adam Sanborn. 2020. Probabilistic Biases Meet the Bayesian Brain. *Current Directions in Psychological Science* 29, 5 (2020), 506–512. https://doi.org/10.1177/0963721420954801 arXiv:https://doi.org/10.1177/0963721420954801

[6] Kushal Chauhan, Rishabh Tiwari, Jan Freyberg, Pradeep Shenoy, and Krishnamurthy Dvijotham. 2022. Interactive Concept Bottleneck Models. *arXiv preprint arXiv:2212.07430* (2022).

[7] Zhi Chen, Yijie Bei, and Cynthia Rudin. 2020. Concept whitening for interpretable image recognition. *Nature Machine Intelligence* 2, 12 (2020), 772–782.

[8] John Joon Young Chung, Jean Y. Song, Sindhu Kutty, Sungsoo (Ray) Hong, Juho Kim, and Walter S. Lasecki. 2019. Efficient Elicitation Approaches to Estimate Collective Crowd Answers. In *CSCW*.

[9] Katherine M. Collins, Umang Bhatt, Weiyang Liu, Vihari Piratla, Ilia Sucholutsky, Bradley Love, and Adrian Weller. 2023. Human-in-the-Loop Mixup. https://doi.org/10.48550/ARXIV.2211.01202

[10] Katherine M Collins, Umang Bhatt, and Adrian Weller. 2022. Eliciting and Learning with Soft Labels from Every Annotator. In *HCOMP*.

[11] Mandeep K. Dhami and David R. Mandel. 2022. Communicating uncertainty using words and numbers. *Trends in Cognitive Sciences* 26, 6 (2022), 514–526. https://doi.org/10.1016/j.tics.2022.03.002

[12] Finale Doshi-Velez and Been Kim. 2017. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv e-prints* (2017), arXiv–1702.

[13] Krishnamurthy Dvijotham, Jim Winkens, Melih Barsbey, Sumedh Ghaisas, Nick Pawlowski, Robert Stanforth, Patricia MacWilliams, Zahra Ahmed, Shekoofeh Azizi, Yoram Bachrach, et al. 2022. Enhancing the reliability and accuracy of AI-enabled diagnosis via complementarity-driven deferral to clinicians (CoDoC). (2022).

[14] Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, and Mateja Jamnik. 2022. Concept Embedding Models. https://doi.org/10.48550/ARXIV.2209.09056

[15] Mateo Espinosa Zarlenga, Pietro Barbiero, Zohreh Shams, Dmitry Kazhdan, Umang Bhatt, Adrian Weller, and Mateja Jamnik. 2023. Towards Robust Metrics for Concept Representation Evaluation. *arXiv preprint arXiv:2301.10367* (2023).

[16] Zoubin Ghahramani. 2015. Probabilistic machine learning and artificial intelligence. *Nature* 521, 7553 (2015), 452–459.

[17] Daniel G Goldstein and David Rothschild. 2014. Lay understanding of probability distributions. *Judgment and Decision making* 9, 1 (2014), 1.

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[19] Katherine H Hall. 2002. Reviewing intuitive decision-making and uncertainty: the implications for medical education. *Medical education* 36, 3 (2002), 216–224.

[20] Lena Heidemann, Maureen Monnet, and Karsten Roscher. 2023. Concept Correlation and Its Effects on Concept-Based Models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 4780–4788.

[21] Dan Hendrycks and Thomas Dietterich. 2019. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261* (2019).

[22] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* 2, 7 (2015).

[23] Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning* 110 (2021), 457–506.

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

[25] Gideon Keren. 1987. Facing uncertainty in the game of bridge: A calibration study. *Organizational Behavior and Human Decision Processes* 39, 1 (1987), 98–114.

[26] Gideon Keren. 1990. Cognitive aids and debiasing methods: can cognitive pills cure cognitive ills? In *Advances in psychology*. Vol. 68. Elsevier, 523–552.

[27] Gideon Keren. 1991. Calibration and probability judgements: Conceptual and methodological issues. *Acta psychologica* 77, 3 (1991), 217–273.

[28] Gavin Kerrigan, Padhraic Smyth, and Mark Steyvers. 2021. Combining Human Predictions with Model Probabilities via Confusion Matrices and Calibration. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 4421–4434. https://proceedings.neurips.cc/paper_files/paper/2021/file/234b941e88b755b7a72a1c1dd5022f30-Paper.pdf

[29] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[30] Joshua Klayman, Jack B. Soll, Claudia González-Vallejo, and Sema Barlas. 1999. Overconfidence: It Depends on How, What, and Whom You Ask. *Organizational Behavior and Human Decision Processes* 79, 3 (1999), 216–247. https://doi.org/10.1006/obhd.1999.2847

[31] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. In *ICML*. PMLR, 5338–5348.

[32] Cassidy Laidlaw and Stuart Russell. 2021. Uncertain Decisions Facilitate Better Preference Learning. *NeurIPS* 34 (2021), 15070–15083.

[33] Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. 2017. Building machines that learn and think like people. *Behavioral and Brain Sciences* 40 (2017), e253. https://doi.org/10.1017/S0140525X16001837

[34] Sarah Lichtenstein, Baruch Fischhoff, and Lawrence D Phillips. 1977. Calibration of probabilities: The state of the art. *Decision making and change in human affairs* (1977), 275–324.

[35] Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. 2020. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems* 33 (2020), 7498–7512.

[36] Joshua Lockhart, Daniele Magazzeni, and Manuela Veloso. 2022. Learn to explain yourself, when you can: Equipping Concept Bottleneck Models with the ability to abstain on their concept predictions. https://doi.org/10.48550/ARXIV.2211.11690

[37] Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. 2021. Promises and pitfalls of black-box concept learning models. *arXiv preprint arXiv:2106.13314* (2021).

[38] Andrei Margeloiu, Matthew Ashman, Umang Bhatt, Yanzhi Chen, Mateja Jamnik, and Adrian Weller. 2021. Do concept bottleneck models learn as intended? *arXiv preprint arXiv:2105.04289* (2021).

[39] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? *NeurIPS* 32 (2019).

[40] Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.

[41] A. O'Hagan, C. E. Buck, A. Daneshkhah, J. R. Eiser, P. H. Garthwaite, D. J. Jenkinson, J. E. Oakley, and T. Rakow. 2006. *Uncertain Judgements: Eliciting Expert Probabilities*. John Wiley, Chichester.

[42] Tuomas Oikarinen, Subhro Das, Lam M Nguyen, and Tsui-Wei Weng. 2023. Label-free Concept Bottleneck Models. In *ICLR*.

[43] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. *Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty under Dataset Shift*. Curran Associates Inc., Red Hook, NY, USA.

[44] Anthony O'Hagan. 2019. Expert Knowledge Elicitation: Subjective but Scientific. *The American Statistician* 73, sup1 (2019), 69–81. https://doi.org/10.1080/00031305.2018.1518265 arXiv:https://doi.org/10.1080/00031305.2018.1518265

[45] Stefan Palan and Christian Schitter. 2018. Prolific. ac—A subject pool for online experiments. *Journal of Behavioral and Experimental Finance* 17 (2018), 22–27.

[46] Joshua C Peterson, Ruairidh M Battleday, Thomas L Griffiths, and Olga Russakovsky. 2019. Human uncertainty makes classification more robust. In *ICCV*.

[47] Vikram V Ramaswamy, Sunnie SY Kim, Ruth Fong, and Olga Russakovsky. 2022. Overlooked factors in concept-based explanations: Dataset choice, concept salience, and human capability. *arXiv e-prints* (2022), arXiv–2207.

[48] Carl Edward Rasmussen and Christopher K. I. Williams. 2006. *Gaussian processes for machine learning*. MIT Press. I–XVIII, 1–248 pages.

[49] Kate Sanders, Reno Kriz, Anqi Liu, and Benjamin Van Durme. 2022. Ambiguous Images With Human Judgments for Robust Visual Event Classification. In *NeurIPS*.

[50] Claudia R Schneider, Alexandra LJ Freeman, David Spiegelhalter, and Sander van der Linden. 2022. The effects of communicating scientific uncertainty on trust and decision making in a public health context. *Judgment and Decision Making* 17, 4 (2022), 849–882.

[51] Tali Sharot. 2011. The optimism bias. *Current biology* 21, 23 (2011), R941–R945.

[52] Ivaxi Sheth, Aamer Abdul Rahman, Laya Rafiee Sevyeri, Mohammad Havaei, and Samira Ebrahimi Kahou. 2022. Learning from uncertain concepts via test time interventions. In *Workshop on Trustworthy and Socially Responsible Machine Learning, NeurIPS 2022*.

[53] Sungbin Shin, Yohan Jo, Sungsoo Ahn, and Namhoon Lee. 2023. A Closer Look at the Intervention Procedure of Concept Bottleneck Models. *arXiv preprint arXiv:2302.14260* (2023).

[54] Ben Shneiderman. 2022. *Human-centered AI*. Oxford University Press.

[55] Mark Steyvers, Heliodoro Tejeda, Gavin Kerrigan, and Padhraic Smyth. 2022. Bayesian modeling of human–AI complementarity. *Proceedings of the National*

Academy of Sciences 119, 11 (2022), e2111547119.
[56] Ilia Sucholutsky, Raja Marjieh, Nori Jacoby, and Thomas L. Griffiths. 2022. On the Informativeness of Supervision Signals. https://doi.org/10.48550/ARXIV.2211.01407
[57] Amos Tversky and Daniel Kahneman. 1996. On the reality of cognitive illusions. *Psychological Review* 103, 3 (1996), 582–591.
[58] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. 2011. *The Caltech-UCSD Birds-200-2011 Dataset.* Technical Report CNS-TR-2011-001. California Institute of Technology.
[59] Bryan Wilder, Eric Horvitz, and Ece Kamar. 2020. Learning to Complement Humans. arXiv:2005.00582 [cs.AI]
[60] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. 2022. Language in a Bottle: Language Model Guided Concept Bottlenecks for Interpretable Image Classification. https://doi.org/10.48550/ARXIV.2211.11158
[61] Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. 2020. On completeness-aware concept-based explanations in deep neural networks. *Advances in Neural Information Processing Systems* 33 (2020), 20554–20565.
[62] Mert Yuksekgonul, Maggie Wang, and James Zou. 2023. Post-hoc Concept Bottleneck Models. https://doi.org/10.48550/ARXIV.2205.15480
[63] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. In *ICLR*.
[64] Kaitlyn Zhou, Dan Jurafsky, and Tatsunori Hashimoto. 2023. Navigating the Grey Area: Expressions of Overconfidence and Uncertainty in Language Models. *arXiv preprint arXiv:2302.13439* (2023).

SUPPLEMENT

Constructing UMNIST

We provide further clarity on how we constructed UMNIST. Each sample of the UMNIST dataset is formed by p 28×28 grey-scale images of handwritten zeros or ones, given as a normalized sample with shape $\mathbf{x} \in [0,1]^{28\times 28\times p}$. We annotate each sample with p binary concept annotations $\mathbf{c} \in \{0,1\}^p$, where c_i indicates whether the i-th image is a one or a zero, and a task label $y \in \{0, \cdots, p\}$ corresponding to the number of ones in its digits, i.e., $y = \sum_i c_i$. To introduce uncertainty in this dataset's samples and concepts, we update concept c_i corresponding to the i-th image as follows:

$$c_i := \begin{cases} \text{Randomly sample from Unif}(0,\delta) & \text{if i-th digit is 0} \\ \text{Randomly sample from Unif}(1-\delta,1) & \text{if i-th digit is 1} \end{cases}$$

where $\delta \in [0,1]$ is a user-provided hyperparameter controlling the amount of dataset uncertainty. Furthermore, in order for this concept annotation uncertainty to be reflected as part of the input digits $\mathbf{x}$, we mix a concept's corresponding digit, akin to Zhang et al. [63], with a randomly selected MNIST training example of the opposite digit using c_i as the mixing ratio. In other words, after generating a sample's uncertain concept annotations $\mathbf{c}$ we update its i-th input digit $\mathbf{x}_{(:,:,i)}$ as follows:

$$\mathbf{x}_{(:,:,i)} := \begin{cases} (1-c_i)\mathbf{x}_{(:,:,i)} + c_i\mathbf{z} \text{ with } \mathbf{z} \sim p_{\text{M}}(\mathbf{x}|y=1) & \text{if } \mathbf{x}_{(:,:,i)} \text{ is 0} \\ c_i\mathbf{x}_{(:,:,i)} + (1-c_i)\mathbf{z} \text{ with } \mathbf{z} \sim p_{\text{M}}(\mathbf{x}|y=0) & \text{if } \mathbf{x}_{(:,:,i)} \text{ is 1} \end{cases}$$

where $p_{\text{M}}(\mathbf{x}|y)$ is the empirical training distribution of MNIST samples whose label is y. For this paper, we focus on using only $p = 10$ digits per sample. See Figure 12 for some examples of this dataset as we vary δ.

Computational experiment details

We next include additional details on how models were trained and run on the various probe datasets, as well as the intervention methods considered.

Training Details for UMNIST *Experiments.* For all UMNIST experiments, we train both CBMs and CEMs using a concept extractor whose architecture consisted of four 3-by-3 convolutional layers with filters $\{5, 10, 20, 40\}$ followed by a linear layer with 20 activations and an output layer with pm output activations, where m is the embedding size used for CEM (one can think of CBM as having $m = 1$). In practice, we set m to 8 following the recommendations from Espinosa Zarlenga et al. [14]. Between all non-output layers, we include leaky-ReLU nonlinear activations and we apply batch normalization after each nonlinearity that follows a convolutional layer. Similarly, for both CEMs and CBMs, we use a simple ReLU two-layer MLP as its concept-to-label map with layers sizes $\{20, p\}$ and train each end-to-end CBM/CEM by weighting the concept loss as much as the task loss (i.e., the joint training hyperparameter α was set to $\alpha = 1$ for both methods). Finally, to avoid each model learning to simply predict the most common class to minimize its error, we weight each sample's task loss according to the empirical label distribution of its corresponding label to encourage our models.

All models are trained by sampling a total of $20,000$ training UMNIST samples, of which 20% were used as a validation set, and

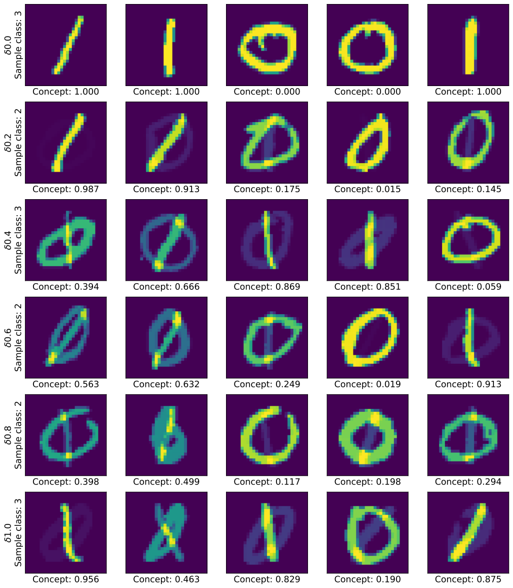


Figure 12: Example datapoints in UMNIST as we vary the value of δ (rows). Each row represents a single sample, with each column representing one of the $p = 5$ digits forming that sample. We include each concept's annotation, as well as the datapoint's label, underneath each digit and to the left of each datapoint, respectively.

tested by sampling 5,000 UMINST testing samples from MNIST's testing set (so no digit in the testing set is ever used to construct UMNIST's training set). We train all models using a standard Adam [29] optimizer with a learning rate 10^{-3} and a batch size of 256 for a maximum of 50 epochs, stopping earlier if the validation loss has not improved for 15 epochs. For each method in UMNIST, we run 5 models from different seeds.

Training Details for CheXpert *Experiments.* For the CheXpert dataset [24], we train all models for 25 epochs, subsampling the dataset to use only 25% of the training dataset when training due to the large size of the dataset. Because the test split for CheXpert does not have the "uncertain" concept label, we perform an 80-10-10 split of the train split into the train, validation, and test folds. Results for CheXpert are averaged over 5 trials, and we use a learning rate of 0.001 across all trials.

Training Details for CUB-Based Experiments. Models trained on CUB followed the same training settings as Espinosa Zarlenga et al. [14]; we employ a single model run for each seed due to computational complexity.

Details on Intervention Policies. The interaction policies we consider in this work (Random and Skyline) consider the setting where a user can be queried to intervene, or edit, a single concept (e.g., wing color) at a time. *Skyline* assumes access to the true label y and how the human would intervene (e.g., assumes access to the CUB-S elicited soft concept annotations), and "tests" intervening with each of the remaining concepts to see which yields the highest predicted probability of the model on the true label. In that way, this mimics an "Oracle" policy, which can greedily select the best of the available next concept interventions, following Chauhan et al.. However, the assumption of knowing the humans' interventions in advance, and the true label, are not realistic (and defeat the purpose of an intervention policy) in practice; hence, this method is meant to capture the "best possible" amount of information that can be gleaned by a single-step direct intervention policy alone. *Random* simply selects the next concept to query by randomly choosing amongst the available concepts which have not yet been queried.

Additional Results on Concept-Incomplete Variant of UMNIST

As discussed by Espinosa Zarlenga et al. [14], CBMs have a significant failure mode when the set of concept annotations available at training time is not fully predictive, or complete, with respect to the task of interest. Similarly to our UMNIST experiments summarized in Figure 3, this section explores how test-time uncertainty affects CBMs and CEMs when the dataset we are working with does not have a complete set of concept annotations. For this, we use our defined UMNIST dataset but only provide 50% of its concept annotations at training time. We train a CBM and CEM using the same configuration and architecture as that described for our UMNIST experiments, with the exception that the concept weight loss α was changed to 0.1. We apply such a change to improve CBM's performance, as otherwise, it was unable to achieve a moderately high task accuracy.

Our results in Figure 13 demonstrate that both CBMs and CEMs significantly drop their performance when test-time uncertainty increases (as we saw in Figure 3 before). Nevertheless, in contrast with Figure 3, we see that interventions in CBMs actually decrease their test accuracy, with uncertainty exacerbating this effect even further. Therefore, these experiments suggest that in concept-incomplete setups, which tend to be what we would expect in real-world datasets given the cost of acquiring all possible concept annotations, CEMs are relatively safer to use regardless of the user's uncertainty at intervention time.

Additional CheXpert Investigations

We include training with simulated uncertainty in Figure 14.

We also explore the original uncertain annotations in CheXpert [24], which contains concept annotations from chest x-rays. The dataset is marked with four labels: positive, negative, unknown, and uncertain. Unknown concepts have no information on their labels, while uncertain labels have information supporting both positive and negative labels. For our experiments, we vary the value taken by uncertain labels, both at train and test time, and investigate its impact on intervention performance. In Figures and 15 and 16, we find that test-time uncertainty improves intervention performance, while train-time uncertainty has minimal impact. This is partially because of the sparsity of uncertain labels in the dataset; only 5% of annotations are marked as uncertain, capping the total effect of train-time uncertainty. For test-time uncertainty, we find that non-zero values improve intervention accuracy, because models are able to distinguish between "uncertain" labels and "negative" labels.

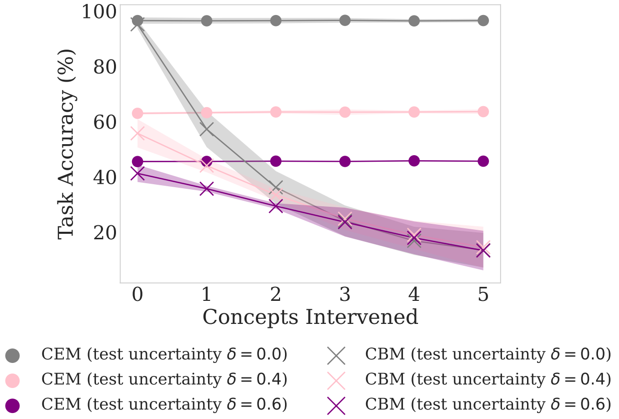


Figure 13: Mean test accuracies of random interventions on CBMs and CEMs, and standard errors across 5 different random initializations, as we increase the number of concepts we intervene on. These models are trained on a variant of UMNIST where we only provide 50% of its concepts at training time.

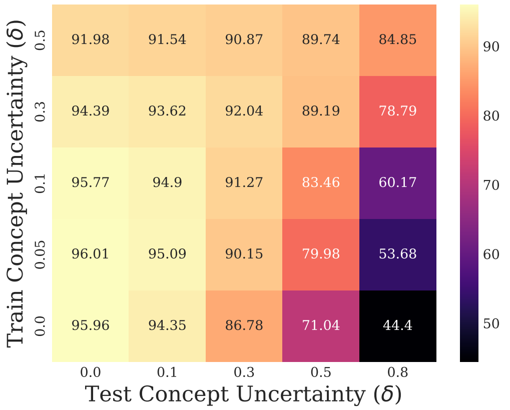


Figure 14: Comparing CEMs trained and tested on differing levels of uncertainty in CheXpert. Heatmap colors depict AUC of the different variants.

Additional Details on CUB-S

We next include additional details on the way we collected CUB-S, as well as further qualitative observations into the labels collected.

Additional Collection Details. Stimuli are preferentially subsampled from the CUB test set to include images which CEMs and CBMs both typically get wrong[4]. Participants are informed the study is intended to last approximately 30 minutes and are paid at a base rate of $9/hr, with an optional bonus paid up to $10/hr to encourage quality predictions; the bonus is applied to all participants.

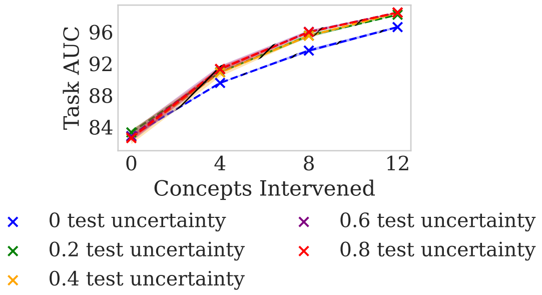


Figure 15: Test-time uncertainty values have a large impact on intervention performance, when using random concept interventions. Setting it to 0 prevents models from differentiating between negative concepts and uncertain concepts, leading to a decrease in performance. However, setting it to non-zero values allows models to pick up on this difference and improve intervention performance.

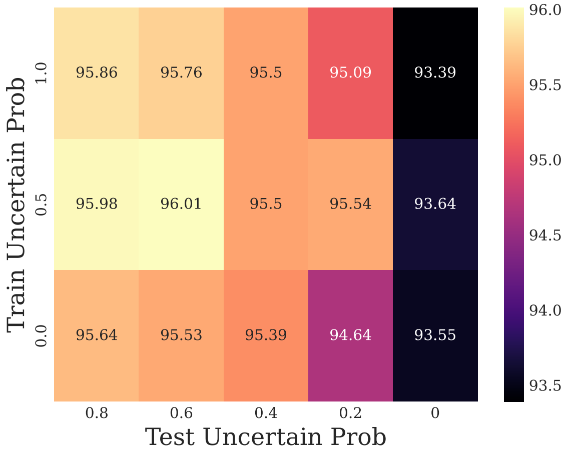


Figure 16: Intervention performance (i.e., random interventions) when using 8 out of 13 concepts to intervene across training and testing uncertainty values. Test uncertainty values have a much larger impact than train uncertainty values, and in general, training with uncertainty seems to have little impact on test-time uncertainty performance.

Additional Observations. We observe in Figure 17 that distribution of provided uncertain annotations is highly irregular, with heavy tails at 0 and 100, and a peak at 50. We hypothesize that heavy tails may be explained by humans rounding values to reduce their cognitive load; Collins et al. found similar rounding effects in free-form uncertainty expression. 50 is the default value provided by the interface, likely underlying the large number of annotations at 50. This suggests there is scope for improving the interface to extract a more accurate distribution of uncertainties, potentially striking a more naunced balance in granularity of information elicited, e.g., [8].

[4] Approximately 50% of the images shown to participants are those which four different seeds of both CEMs and CBMs got incorrect, rendering them more interesting - and challenging - to study at intervention-time

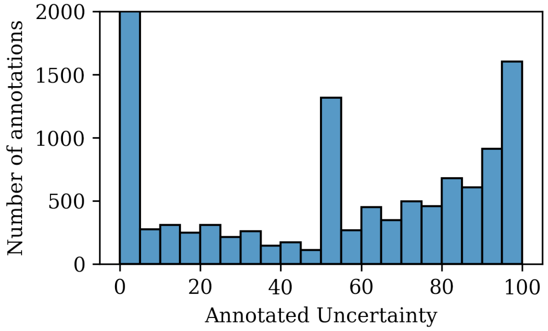


Figure 17: Distribution of uncertainty values for all annotations in CUB-S. Annotators favor certain annotations (0 or 100) and the default value of 50 provided by the interface.

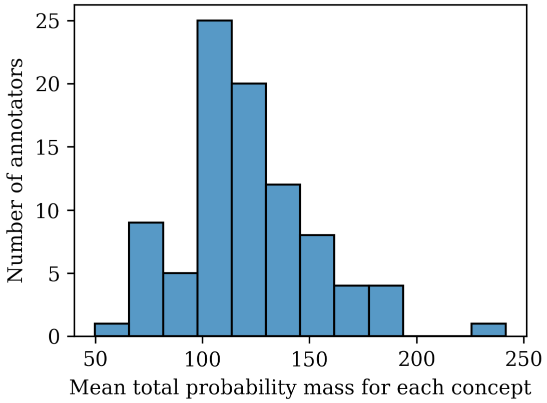


Figure 18: Histogram showing the distribution of mean total probability mass for each concept assigned by each annotator. Most annotators assign approximately 100 probability mass, although there are a significant number which over-assign probability mass.

As observed in Section 5.3.3, the calibration of individual annotators varies significantly. Figure 17 shows that most annotators consistently assign approximately 100 probability mass for each concept, as one would expect. However, the distribution is positively skewed, with a significant number of annotators consistently over-assigning probability mass acoss the concept groups (for any individual concept, the annotator can endorse at most 100 "probability units"). This is partly explained by concept groups where more than one concept is relevant (such as color), although it is also likely that annotators are overestimating their confidence.

Further, we investigate the variance in flavor of uncertainty expressed between different concepts. In Figure 19 we plot the distribution of probability mass assigned for each concept. We observe significant variations between concepts, in terms of their mean, variance and skew. Some concepts such as "eye color" have a very tight distribution around 100, suggesting those concepts are "easy" to annotate. In contrast, some concepts such as "upperparts color" show greater variation in the probability mass assigned. These concepts tend to either be color concepts, which can have several correct annotations, or ambiguous concepts like "size" which may be harder to annotate correctly.

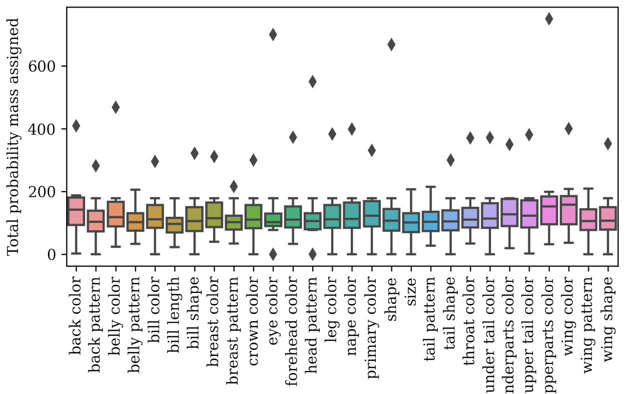


Figure 19: Distribution across images of total probability mass assigned for each concept. There is significant variation in the mean, skew and variance of distributions, showing that different concepts are annotated differently by human annotators.

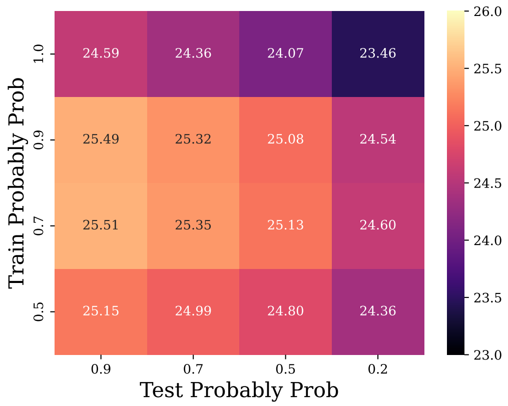


Figure 20: Training with a moderate level of (aggregate/population-level) uncertainty improves robustness under test-time uncertainty; as measured by AUC between intervention-accuracy curve. Higher is better.

These observations highlight nuances in the CUB-S dataset which aren't present in the original hard annotations. Soft annotations give insights into how humans interpret concepts when labeling and the variation in individual calibration of annotators [10]. We hope to encourage future work to design ML models and datasets which account for the idiosyncrasies of human uncertain annotations.

Additionally, our labels demonstrate potential issues with the concept filtering typically applied on CUB. Koh et al. propose a filtering scheme to avoid overly sparse annotations; however, we note that our annotators assign a substantial amount of probability

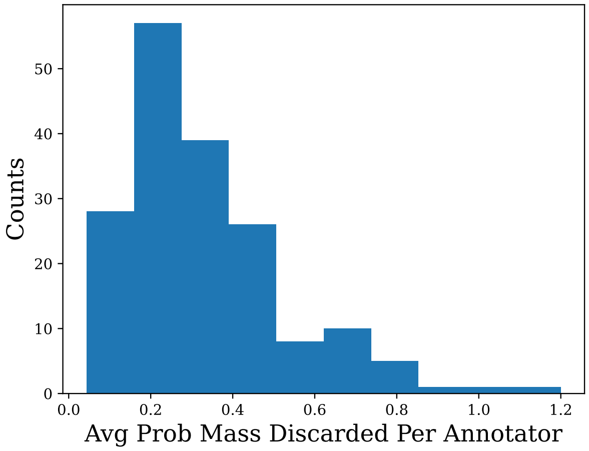


Figure 21: Amount of assigned probability mass discarded per individual when using the popular Koh et al. concept filtering (averaged over concept groups).

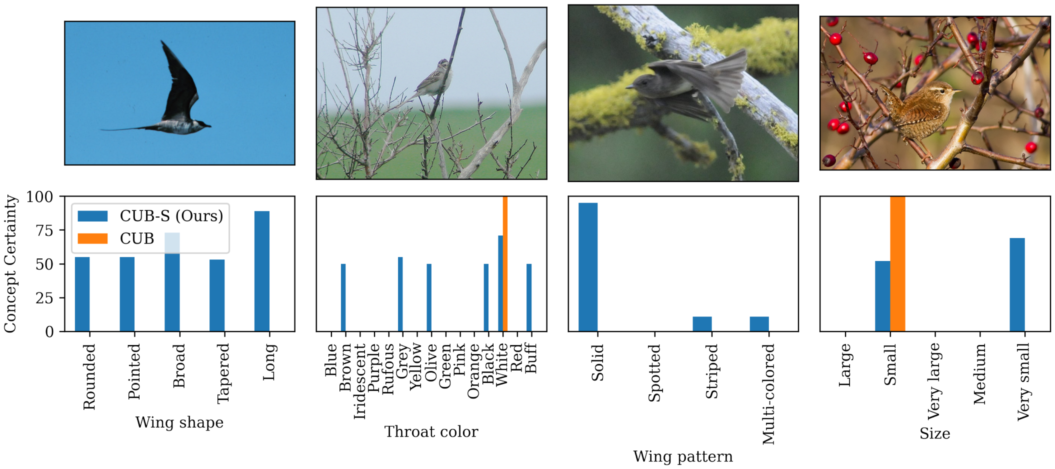


Figure 22: Additional examples showing rich annotations for CUB-S compared to hard assignments in CUB.

mass to concept attributes which are *filtered out* (see Figure 21). These data highlight that the filtered out attributes could indeed be missing critical information from people as to what is in the image.

Additional CUB Uncertainty Computational Experiments

We next include further observations from our computational experiments in CUB and CUB-S.

Broad vs. Narrow Uncertainty. We demonstrate the sensitivity of concept-based systems to broad versus narrow uncertainty under the Random policy (see Figure 25), further highlighting that the method of distributing uncertainty through discrete confidence scores matters impacts intervention efficacy.

Individual- vs Population-Level Uncertainty. As noted, whether or not we intervene with individual or population-level annotations matters (see Section 5.2.3), and we see in Figure 20 that training and then intervening with population-level annotations yields the best performance. These observations are relevant not only to ML practioners who work with CUB, but broadly in annotation-design and questions around who and how many annotators should we elicit from.

CBMs and Simulated Uncertainty. Further, we concretize why we focus on CEMs in the bulk of this work. CBMs severely struggle

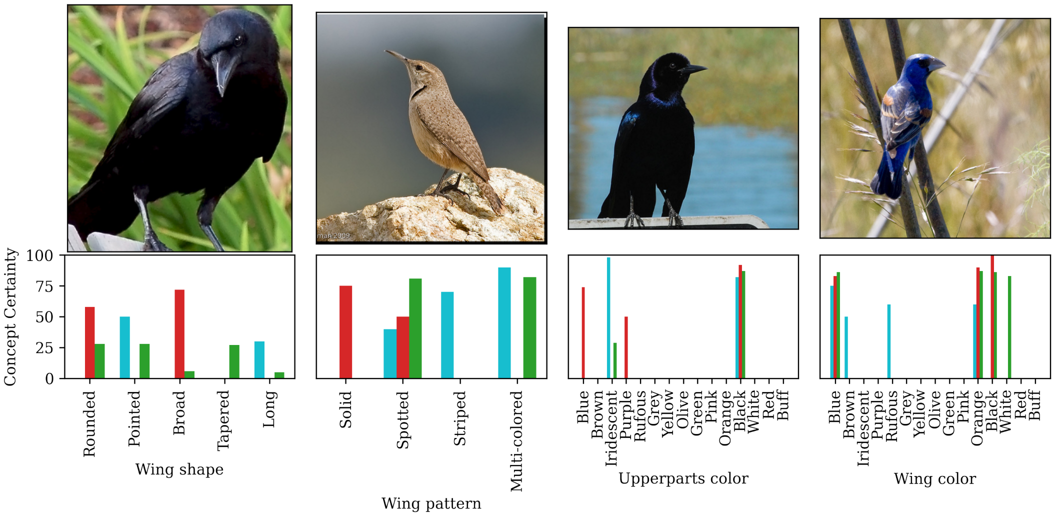


Figure 23: CUB-S Examples where multiple annotators labelled the same image. Each bar color represents a unique annotator for each image. The annotated concepts vary significantly between annotators, especially for challenging concepts such as "wing shape" and "wing pattern".

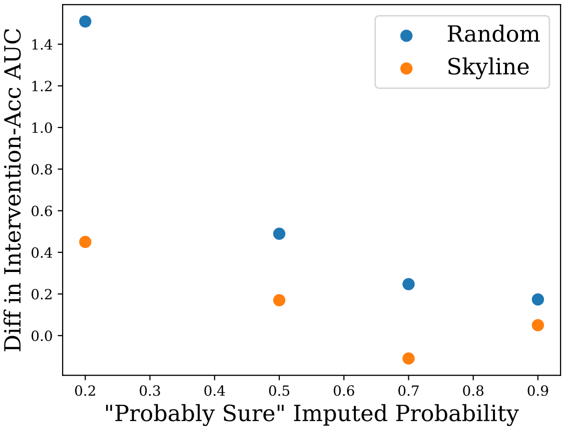


Figure 24: It matters whether or not we use *instance-level, individual* annotator uncertainty, or average over many individuals' uncertainty. Averaging improves the stability of interventions; but in practice, we may only have a single individual who can provide their uncertainty. We find sizeable differences in the intervention efficacy when using averaged uncertainty for both Skyline and Random.

under test-time uncertainty when dealing with concept-incomplete datasets (see the UMNIST section of this Supplement) and in-the-wild uncertainty (see Figure 26).

Skyline Selections Reveal "Helpful" and "Harmful" CUB-S *Annotations.* As seen in Figure 11, Skyline rapidly improves by selecting "good" uncertain annotations; however, the final selections hamper performance. We demonstrate how human selections can both help and hinder performance in Figure 27. We depict the proportion of selections for each concept being in the first or last 5 selections by Skyline. Avoiding selecting the examples in the last 5, e.g., "upperparts" color, offer promising directions for future policy design and investigation into when and why humans are good uncertain annotators. Interestingly, we observe differences in which concepts are preferred depending on whether the model was trained without (Figure 28) or with uncertainty in the concepts at training time (i.e., Figures 29, 30, 31).

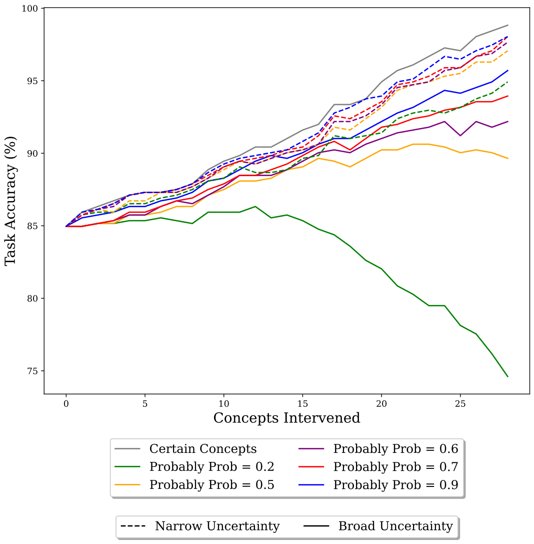


Figure 25: Impact of different ways of distributing the discrete uncertainty over categorical concept groups, selected using Random intervention policies on CEMs.

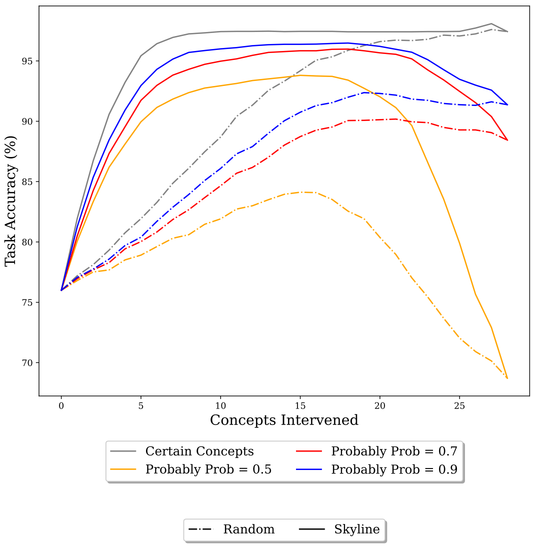


Figure 26: CBMs struggle to handle uncertainty in CUB as well and are comparatively worse than CEMs.

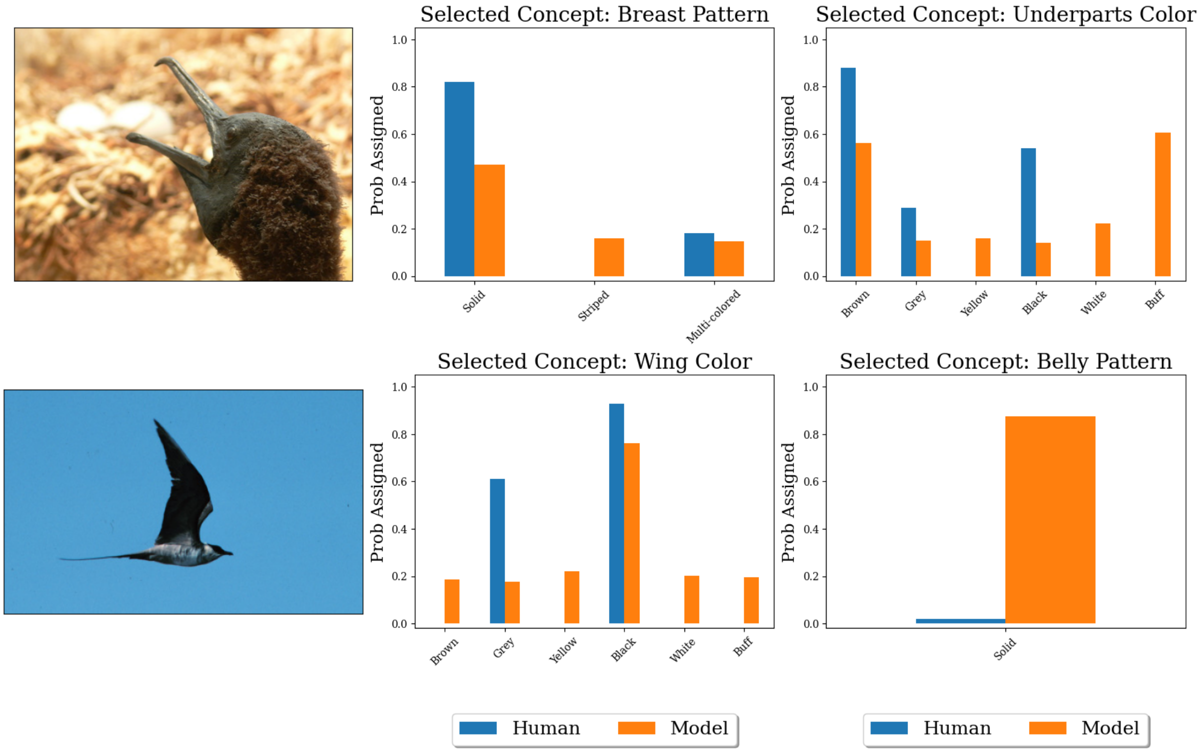


Figure 27: Model versus human distributions over concepts at the time of selection by Skyline. The first column of distributions are selections which boosted the model's classification (from incorrect to correct); humans' uncertainty was helpful to intervene with. The second column of distributions depicts the human uncertainty at intervention time which *hurt* model performance (the classification went from correct to incorrect). Model trained on uncertain concepts ("Probably" probability = 0.7).

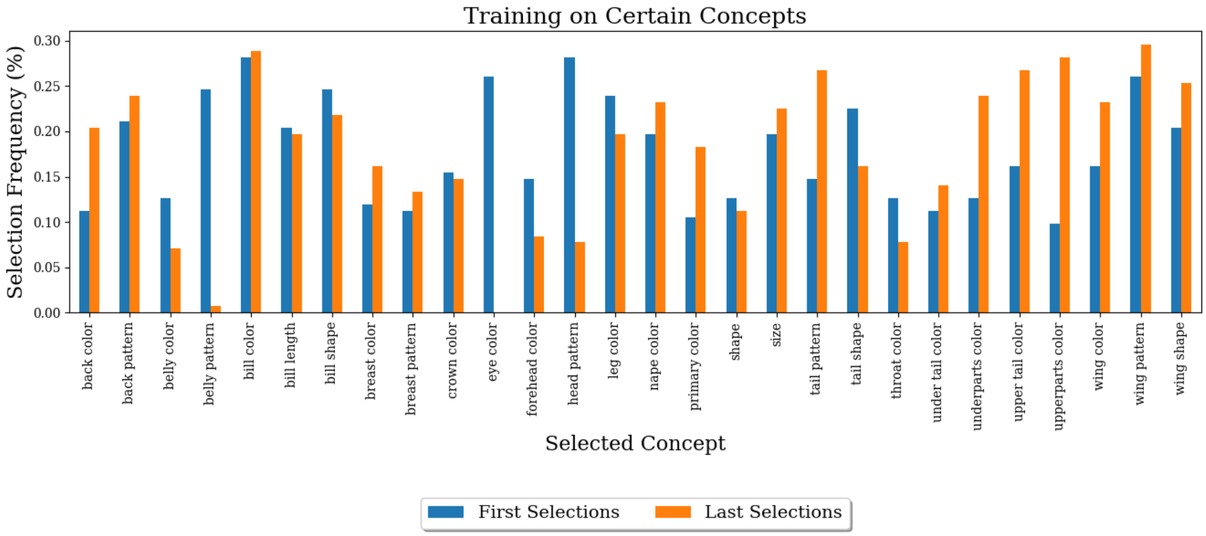


Figure 28: Skyline selections for CEM run on CUB-S reveal when human uncertainty elicitation is helpful (versus harmful). Proportion of selections for each concept being in the first or last 5 selections by Skyline. CEM trained on certain concepts.

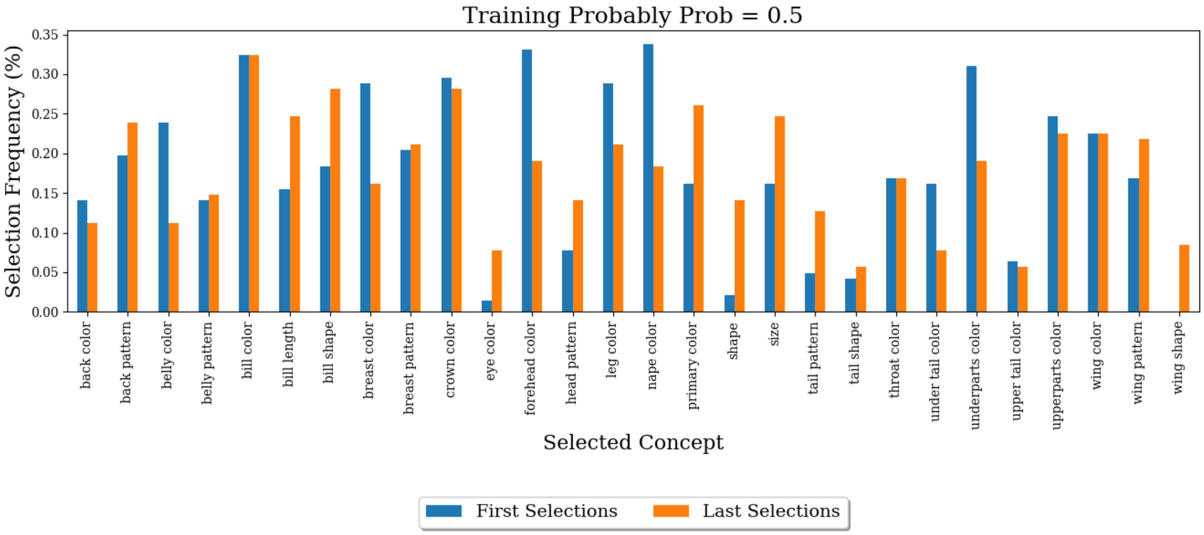


Figure 29: Skyline selections for CEM trained on uncertain concepts (where the imputed "Probably" probability is set to 0.5). Population-level broad uncertainty labels used.

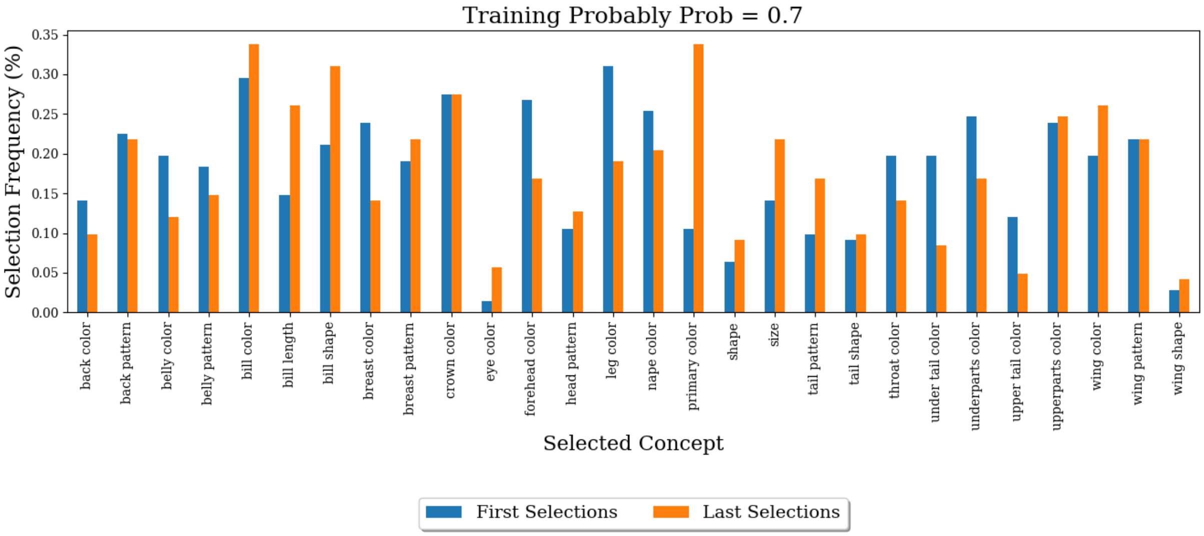


Figure 30: Skyline selections for CEM trained as in Figure 29, but with the imputed "Probably" probability set to 0.7.

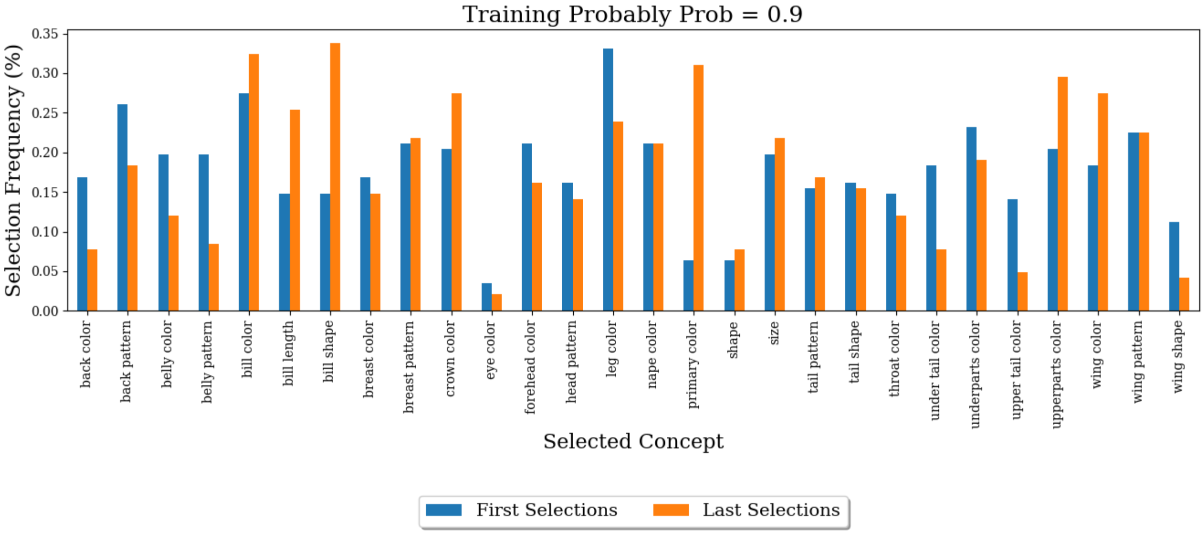


Figure 31: Skyline selections for CEM trained as in Figure 29, but with imputed "Probably" probability set to 0.9.