9. Appendix: Usage Scenarios

To demonstrate the effectiveness of INFICOND, we describe its usage by Sam on a KD analysis and fine-tuning workflow. Specifically, we demonstrate using the Q2L model trained on the PASCAL VOC 2012 dataset, which has 20 classes and contains 10,582 training and 1,449 validation images. Per the descriptions in Section 4, INFICOND will use an concept corpus of 584 text-aligned visual concepts.

As there are 20 classes in the Q2L model, INFICOND will distill 20 student models, each having one fully-connected layer (no activation) of 584 input nodes and 1 output node. We use L1 regularization with weight 10^{-4} to balance student accuracy and its weight sparsity, a batch size of 2,084, and an Adam optimizer with a learning rate of 0.2. We also use batch normalization during training to stabilize the gradients. Sam trains the student models to mimic the Q2L predictions (i.e., using the Q2L model's outputs as ground truth).

Figure 7: *A closeup of the sheep class in the student performance view, indicating how the student model (right circle) outperforms the teacher model (left circle). The horizontal bars provide more detailed performance insights on images with the class (top row) and without the class (bottom row).*

9.1. Initial Performance Review of High-Performing Classes

Sam begins by reviewing the performance of the student models and identifying high performing classes that do not need fine-tuning. Upon review, he discovers several classes in which the student model outperforms the teacher model, including *sheep*, *bird*, and *horse*. Specifically, for the *sheep* class, the student model achieves an AP of 96.25%, while the teacher model only achieves 95.31% (as shown in Figure 7). Similarly, for the *horse* class, the student model attains an AP of 98.18%, compared to the teacher model's 98.03%. For the *bird* class, the student model has an AP of 99.84%, surpassing the teacher model's 99.74%. These results indicate that the student and teacher models had small capacity gaps, and Sam is able to use the panels in INFICOND to review how the visual concepts contributing to these classes are highly relevant, indicating that the student models have learned to predict on the "correct" concepts (i.e., that align with Sam's mental model) present in images. Encouraged by this, he proceeds to examine other potential classes for improvement, focusing on cases where the student model underperforms the teacher model.

9.2. Improving Underperforming Classes

Improving the *tv monitor* class. Further reviewing the student performance view, Sam realizes for the *tv monitor* class, the student model lags behind the teacher models by ∼2%. While this is a good overall result for a student model, he can see that the main performance gap occurs in the *tv monitor* class's shaded blue area (Figure 8a), where the student mispredicts 4.08% of positive samples that are correctly identified by the teacher. Sam wants to address this nuanced misprediction issue, but to do so, he will need to increase the student's accuracy at true positive samples while avoiding adding false positives.

Sam realizes this can be accomplished by increasing the influence of concepts highly relevant to *tv monitors*. He clicks on the *tv monitor* text to load this student model into the student knowledge view and updates the embedding view. Using the embedding view, he identifies similar concepts such as `tv monitor`, `crt screen`, and `television`, which are relevant to the notion of tv monitors (this is shown in the concept embedding panel in Figure 5c), and then proceeds to examine the student knowledge view, sorting by weights. Though the `tv monitor` and `television` concepts already have high influences (ranked at #2 and #3, respectively), the performance charts suggest they can be further increased to achieve additional performance gains (see Figure 5d2). Sam uptunes these concepts and sees a significant improvement in performance. He scrolls down and notices that `crt screen` only has an influence of 0.165, even lower than `bookcase` and `shop`. Sam reasons this likely occurred because many training instances of tv monitors likely included a `bookcase` or `shop` setting as a part of the image, hence their concepts were quantified as more important to *tv monitor* predictions. However, from a human reasoning perspective, the presence of a `crt screen` is a much more important concept than these two items. As such, Sam uptunes the `crt screen` concept. Satisfied with this tuning, Sam submits this batch of tuning instructions and gains a 1.34% improvement (see Figure 8b).

Upon rechecking the performance view, Sam notes that while there was previously 4.08% difference in false negative rate (% more mispredicted positive samples in student than in teacher, as shown in Figure 8a), this difference has now dropped to zero. However, the false positive rate has also increased by 1.4%; previously there were only 0.56% more mispredicted negative samples in student than in teacher, and now the gap is 1.96%. In other words, based on this initial fine-tuning activity, some irrelevant concepts have gained increased influence which are causing some negative samples to be mispredicted as positive. To mitigate this confound, Sam decides to reduce the influence of irrelevant-yet-highly influential concepts. He reviews the list of concepts on the *tv monitor* class in the student performance view, and decreases the influence of several irrelevant concepts, including `bookcase`, `mezzanine`, `blinds`, `niche`, `sofa` and `decoration`, none of which are relevant to whether an image should classified as having a *tv monitor*. This gives an additional overall improvement of 0.55% in the *tv class* student model. He then goes through the list of concepts again and this time he sees two more irrelevant concept — `bookcase` — that can also be reduced to improve student performance. He does this and gains another 0.29% improvement.

Overall, Sam has improved the performance of the underperforming *tv monitor* student model by 2.18%, resulting in a predictive performance that is 0.77% ahead of the teacher model, and reducing several "gaps" where the student model shows issues (e.g., student mispredictions of positive samples that the teacher model correctly predicted) all in a no-code, interactive manner.

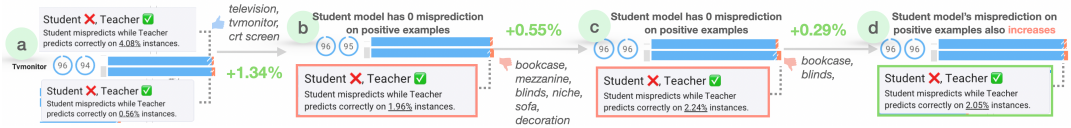

Figure 8: *In three iterations, Sam improves the tv monitor class from underperforming by 1.41% to outperforming by 0.77%, with an overall average precision increase of 2.18%.*

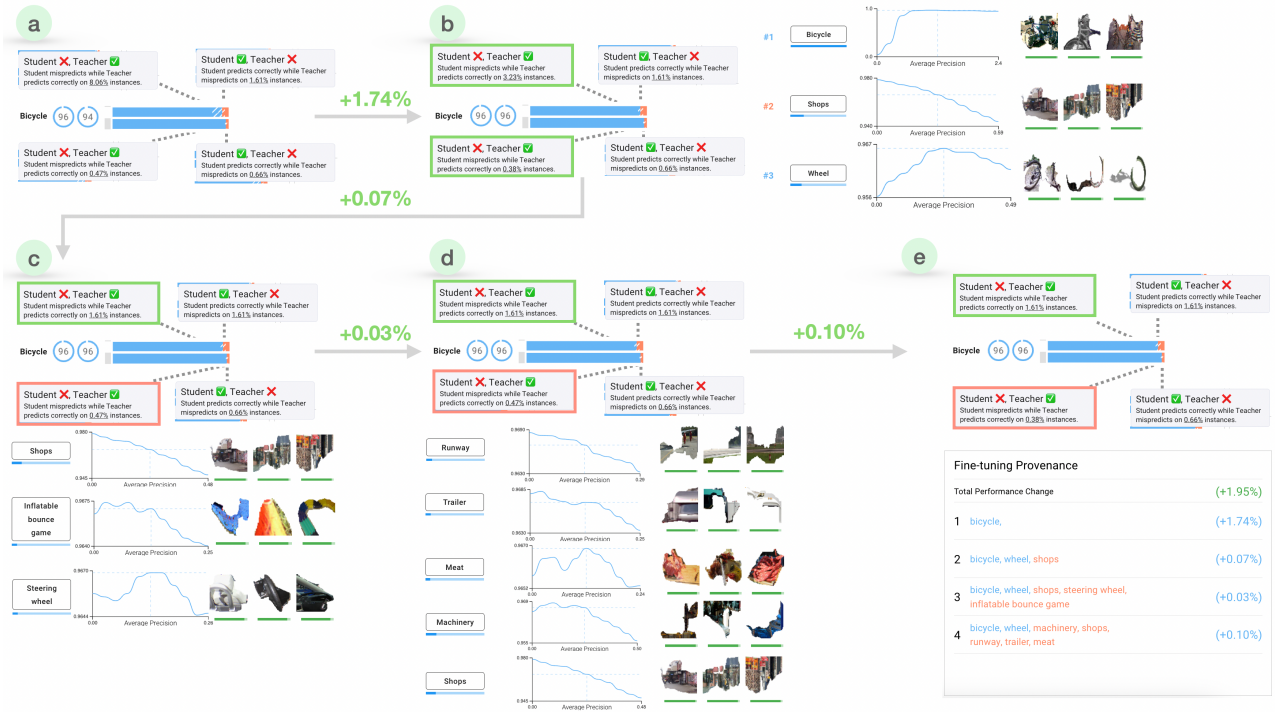

Figure 9: *In four iterations, Sam successfully transforms the bicycle class from an underperforming class to an overperforming one. Starting by identifying the primary performance gap as the student model's inability to accurately predict positive samples, Sam focuses on increasing the influence of concepts that are relevant to bicycle. He then examines the student knowledge view and reduces the influences of irrelevant concepts to decrease false negative rates. By taking a systematic and thorough approach, Sam achieves significant improvements in the student model's performance by 1.95%.*

Improving the *bicycle* class. Sam now addresses another significantly underperforming student model: *bicycle*, which has a 1.89% AP gap between the student and teacher models (Figure 9a) In particular, the student model mispredicts 8.06% more positive samples compared to the teacher model, highlighting the need to increase the influence of highly relevant concepts.

Sam examines concept relationships in the embedding view and identifies pertinent concepts such as `bicycle` and `wheel`. In the student knowledge view (Figure 9b), Sam sorts concepts by presence discrepancy among correct and incorrect positive samples. He identifies the `bicycle` concept as a good candidate for enhancement, so he uptunes it. However, Sam opts against uptuning the `wheel` concept, as the performance chart does not indicate a performance increase by enhancing its influence (Figure 9b). This decision feels logical to him, as "wheels" are related, but certainly not exclusive to, "bicycles." Sam checks the concept detail view for `wheel` and confirms its presence in additional classes, including *motorbike* and *car*. He decides to initially maintain the `wheel` concept's current state of influence, and only uptune the `bicycle` concept, which leads to a 1.83% performance increase.

Upon reexamining the performance view, Sam observes a significant drop in the false negative rate (4.83%) as the student model previously mispredicts 8.06% more positive samples than the teacher and it does only at 3.23% now (Figure 9b). In the negative samples, the student model is also mispredicting fewer samples than the teacher model. It previously mispredicts 0.47% more and it is now mispredicting only 0.38% more. However, it used to have 0.66% more correctly predicted positive samples than the teacher model; it now only has 0.47% (Figure 9c).

Examining the updated student knowledge view, Sam notices that the `shop` concept has more influence than the `wheel` concept, which intuitively does not make sense, as "wheels" are more

relevant to bicycles than "shops." Thus, he now decides it's time to tune up `wheel` and `bicycle` together, while downtuning `shop`. This brings him an increase of 0.07% (Figure 9c). However, he notices that the false positive rate increases by 0.09%. To address this issue, he continues to downtune irrelevant concepts with unreasonably high influence, while further uptuning `bicycle` and `wheel` to reinforce the effect. After downtuning `shop`, `steering wheel`, and `inflatable bounce game` (all of which he deems irrelevant to "bicycles"), he gains an improvement of 0.03%(Figure 9d). In the subsequent iteration, after downtuning `machinery`, `shops`, `runway`, `trailer`, and `meat`, he gains another 0.10% improvement (Figure 9e). While these individual fine-tunings only result in small overall performance increases for the student module (finally reaching an AP of 96.82% of AP, which outperforms the teacher model by 0.06%), this process has also aligned the student model's decision making process with his own human intuition — e.g., the *bicycle* class now emphasizes the concepts he deems most relevant, and minimizes irrelevant ones. Overall, Sam is pleased with his outcome and moves on to address other underperforming classes.

9.3. Mitigating Severe Underperformance

Mitigating underperformance of the *sofa* class. Sam now turns his attention to a severely underperforming class; *sofa*. Here, the student model lags the teacher model by 17.0%. To address this, Sam first examines the student performance view for the *sofa* class and observes a 9.62% performance difference in mispredictions of positive samples by the student model compared the teacher model (Figure 10a). Sam decides to increase the influence of relevant concepts, to see if this can help close this very high capacity gap.

After reviewing several concepts in the the student knowledge view, Sam uptunes the `sofa` and `armchair` concepts (as armchairs are very similar to sofas, and both seem highly relevant concepts to a *sofa* prediction itself). Uptuning only these two concepts results in a 2.10% improvement of the student model (Figure 10b).

In the student performance view, Sam sees a reduction of false negative rates by 5.77%, but also an unwanted increase in false positive rates by 0.74%. To address this, he decides to downtune irrelevant concepts and further uptune the `sofa` and `armchair` concepts. After again reviewing available concepts in the student knowledge view, he downtunes the irrelevant concepts `trouser`, `notebook`, `steering wheel`, `dog`, and `pillow`, resulting in a 1.42% increase (Figure 10c).

As Sam continues to iteratively tunes the student model, he gains incremental improvements each time. After 14 iterations (Figure 10e), the student model reaches 67.0% of AP, which is 7% higher than the initial 60% of AP (Figure 10d). The student performance view shows a significant reduction in the shaded blue area, which indicates a large decrease in false negatives and a closed capacity gap. The performance chart of the *sofa* class also suggests that if Sam continues to uptune the `sofa` concept, he can potentially come even closer to 70% of AP for the *sofa* class (Figure 10f).

Mitigating underperformance of the *dog* class. Sam next looks at the *dog* class, which is also severely underperforming. The initial performance of the student model is quite low, with only 90.72%

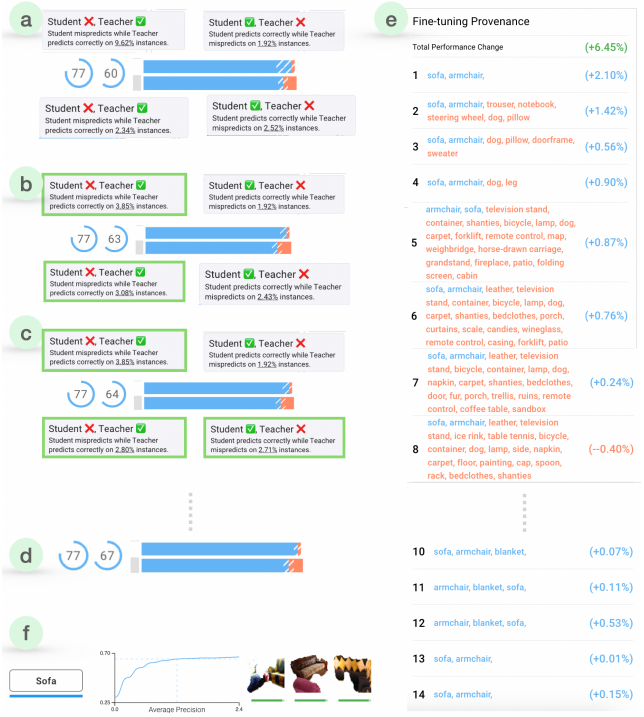

Figure 10: *An illustration for how Sam mitigates severe underperformance of the sofa class*

of AP compared to the teacher model's 99.94%. Upon examining the student performance view, Sam identifies the primary issue as a high false negative rate, with the student model mispredicting 9.09% more positive samples than the teacher model.

To address this issue, Sam decides to enhance the influence of the `dog` concept, which is the most relevant and influential concept for the *dog* class. In the first iteration, uptuning the `dog` concept alone brings an improvement of 2.02% and reduces the false negative rate by 1.17%. However, the student model still mispredicts 7.92% more positive samples than the teacher model, so Sam continues uptuning the `dog` concept. Besides uptuning the `dog` concept, he also uptunes `fur` and downtunes `grass`, terraces and apron concepts, which are concepts that can often co-occur with *dog*s (e.g., they are photographed outside, or on a terrace) but may lead to false positives.

In a second fine-tuning iteration, the results show an improvement of 0.48% AP and a reduction in false negative rate to 5.94%. Encouraged, Sam continues to uptune the `dog` concept in subsequent iterations while also downtuning other irrelevant-yet-influential concepts such as `grass` and `cushion`. After eight rounds, the student model reaches 94% AP, an improvement of 3.28% from the initial performance, and mispredicts only 3.74% more positive samples than the teacher model. With this performance gap significantly reduced, Sam decides to move on to other classes to further improve the student model.