Overlooked Factors in Concept-based Explanations: Dataset Choice, Concept Salience, and Human Capability Supplementary material

Anonymous CVPR submission

Paper ID 1118

In this supplementary document, we provide additional details on some sections of the main paper.

Section A We provide more information regarding our experimental setup over all experiments.

Section B We provide additional results from our experiments regarding probe dataset choice from Section 3 of the main paper.

Section C We provide additional results from our experiments regarding concept choice from Section 4 of the main paper.

Section D: We supplement Section 5 of the main paper and provide more information about our human studies.

Section E: We supplement Section 5 of the main paper and show snapshots of our full user interface.

A. Experimental details

Here we provide additional experimental details regarding all our setups, as well as the computational power we needed.

TCAV. Using the features extracted from the penultimate layer of the ResNet18-based [4] model trained on the Places365 dataset [13], we use scikit-learns's [10] `LogisticRegression` models to predict the ground truth attributes in each case. We use the `liblinear` solver, with an `l2` penalty, and pick the regularization weight as a hyperparameter, based on the performance (ROC AUC) on a validation set.

Baseline. Given the ground-truth labelled concepts for an image, this explanation attempts to predict the blackbox model's output on the image. We use scikit-learn's [10] `LogisticRegression` model with a `liblinear` solver, and an `l1` penalty, to prioritize learning simpler explanations. For the experiment reported in Section 3 of the main paper, we pick the regularization weight as a hyperparamater, choosing the weight with the best performance on a validation set. When generating explanations of different complexities for our human studies, we vary the regularization parameter, picking explanations that use a total of 4, 8, 16, 32, or 64 concepts.

Learning concepts. We computed features for all images from ADE20k [14, 15] using the penultimate layer of a ResNet18 [4] model trained on Imagenet [11]. We then learned a linear model for all concepts that had over 10 positive samples within the dataset, using the `LogisticRegression` model from scikit-learn [10]. Similar to other models, we use a `liblinear` solver, with an `l2` penalty, choosing the regularization weight based on performance (ROC AUC) on a validation set. As mentioned, we report the normalized AP [5] to be able to compare across concepts and target classes with varying base rates.

Run times. Computing each of the linear models used less than 2 min on a CPU. Computing features using a ResNet18 [4] model trained on either Places365 [13] or Imagenet [11] for the ADE20k [14, 15] and Pascal [2] datasets took less than 15 min using a NVIDIA GTX 2080 GPU.

B. Probe dataset choice: more details

In our first claim, we show that the choice of probe dataset can have a significant impact on the explanation output for concept-based explanations. We give more details from our experiments for this claim within this section.

B.1. Varying the probe dataset

Here we provide the full results from section 3.1 in the main text, where we compute concept-based explanations using 2 different methods (NetDissect [1] and TCAV [6]) when using either ADE20k or Pascal as probe datasets.

NetDissect. Table 1 contains the label generated for all neurons that are strongly activated when using either ADE20k [14, 15] or Pascal [2] as the probe dataset. A majority of neurons (69/123) correspond to very different concepts.

Neuron	ADE20k label	ADE20k score	Pascal label	Pascal score	Neuron	ADE20k label	ADE20k score	Pascal label	Pascal score
1	counter	0.059	bottle	0.049	3	sea	0.067	water	0.065
4	seat	0.064	tvmonitor	0.074	8	vineyard	0.048	plant	0.043
9	plant	0.082	pottedplant	0.194	22	bookcase	0.07	bus	0.048
30	house	0.094	building	0.043	37	boat	0.043	boat	0.213
43	bed	0.151	bed	0.075	47	pool table	0.135	airplane	0.079
60	plane	0.052	airplane	0.168	63	field	0.053	muzzle	0.042
69	person	0.047	hair	0.086	73	water	0.041	bird	0.080
79	plant	0.064	pottedplant	0.064	90	mountain	0.071	mountain	0.066
102	bathtub	0.040	cat	0.055	104	cradle	0.081	bus	0.112
105	sea	0.106	water	0.058	106	rock	0.048	rock	0.06
110	painting	0.119	painting	0.06	112	field	0.05	bus	0.051
113	table	0.116	table	0.066	115	plane	0.046	airplane	0.147
120	sidewalk	0.042	track	0.075	125	table	0.049	wineglass	0.047
126	stove	0.064	bottle	0.163	127	book	0.104	book	0.096
131	signboard	0.043	body	0.069	134	bathtub	0.088	boat	0.059
141	skyscraper	0.065	cage	0.068	155	mountain	0.091	train	0.058
158	book	0.042	book	0.052	165	sea	0.051	water	0.051
168	railroad train	0.055	train	0.193	172	car	0.055	bus	0.101
173	car	0.052	bus	0.099	181	plant	0.068	pottedplant	0.14
183	person	0.041	horse	0.187	184	cradle	0.046	cat	0.042
185	chair	0.077	horse	0.153	186	person	0.051	bird	0.094
191	swimming pool	0.044	pottedplant	0.072	198	pool table	0.064	ceiling	0.066
208	shelf	0.047	bus	0.062	211	computer	0.076	tvmonitor	0.089
217	toilet	0.049	hair	0.055	218	case	0.044	track	0.165
219	plane	0.065	airplane	0.189	220	road	0.066	road	0.066
222	grass	0.105	grass	0.046	223	house	0.069	airplane	0.055
231	grandstand	0.097	screen	0.047	234	bridge	0.05	train	0.042
239	pool table	0.069	horse	0.171	245	water	0.063	water	0.042
247	plane	0.079	airplane	0.177	248	bed	0.127	tvmonitor	0.063
251	sofa	0.073	pottedplant	0.053	257	tent	0.042	bus	0.279
260	flower	0.082	food	0.069	267	apparel	0.042	car	0.045
276	earth	0.041	rock	0.047	278	field	0.06	sheep	0.044
280	mountain	0.045	mountain	0.056	287	plant	0.078	pottedplant	0.07
289	pool table	0.049	food	0.059	290	mountain	0.085	mountain	0.097
293	shelf	0.074	bottle	0.105	298	path	0.047	motorbike	0.068
305	waterfall	0.057	mountain	0.047	309	washer	0.109	bus	0.065
318	computer	0.079	tvmonitor	0.251	322	ball	0.054	sheep	0.044
324	mountain	0.071	motorbike	0.048	325	person	0.04	head	0.059
327	waterfall	0.055	bird	0.087	337	water	0.072	boat	0.109
341	sea	0.153	boat	0.076	344	person	0.052	person	0.048
345	autobus	0.042	bus	0.142	347	palm	0.051	bicycle	0.083
348	mountain	0.058	mountain	0.125	354	cradle	0.042	chair	0.053
357	rock	0.058	sheep	0.061	360	pool table	0.048	bird	0.041
364	field	0.058	plant	0.041	372	work surface	0.045	cabinet	0.049
379	bridge	0.092	bus	0.046	383	bed	0.069	curtain	0.079
384	washer	0.043	bicycle	0.201	386	autobus	0.067	bus	0.200
387	hovel	0.04	train	0.085	389	chair	0.066	chair	0.051
398	windowpane	0.073	windowpane	0.07	400	plant	0.043	pottedplant	0.097
408	toilet	0.045	bottle	0.099	412	bed	0.079	airplane	0.086
413	pool table	0.09	motorbike	0.07	415	seat	0.044	tvmonitor	0.045
417	sand	0.06	sand	0.049	419	bed	0.061	tvmonitor	0.054
422	seat	0.089	tvmonitor	0.056	430	bed	0.078	bedclothes	0.042
434	case	0.047	cup	0.041	435	runway	0.072	airplane	0.189
438	plane	0.045	airplane	0.235	444	sofa	0.045	plant	0.09
445	car	0.201	car	0.093	446	pool table	0.193	tvmonitor	0.086
454	car	0.218	car	0.156	463	snow	0.059	snow	0.118
465	crosswalk	0.097	road	0.047	475	cradle	0.061	train	0.132
477	desk	0.104	tvmonitor	0.085	480	sofa	0.086	sofa	0.081
483	swivel chair	0.052	horse	0.041	484	water	0.15	water	0.102
485	sofa	0.056	airplane	0.045	500	sofa	0.156	sofa	0.11
502	washer	0.07	train	0.134	503	bookcase	0.109	book	0.075
509	computer	0.044	tvmonitor	0.074					

Table 1. We show labels for all neurons from the penultimate layer of a ResNet18 model that are marked as highly activated by both datasets by NetDissect [1]. We find that a 69/123 of neurons correspond to labels that are radically different (shown in red). The remainder correspond to either the same or very similar concepts.

As mentioned by Fong *et al*. [3] and Olah *et al*. [9], neurons in deep neural networks can be *poly-semantic*, i.e, some neurons can recognize multiple concepts. We check if the results from above are due to such neurons, and confirm that that is not the case: out of the 69 neurons, only 7 are highly activated (IOU¿0.04) by both concepts. Table 2 contains the IOU scores for both the ADE20k and Pascal label for each neuron outputting very different concepts.

TCAV. We report the cosine similarities between the concept activation vectors learned using ADE20k and Pascal as probe datasets for all 32 concepts that have a base rate of at least 1% in Table 3. On the whole, we see that the vectors are not very similar, despite the vectors predicting the concepts well.

B.2. Difference in probe dataset distribution

The first method we use to look at the difference in the 2 probe datasets we used was to consider the base rates of different concepts within the dataset. As noted in Section 3 of the main paper, there are some sizable differences. Figure 1 contains the base rates for all concepts highlighted in Table 2 of the main paper. Some concepts that have very different base rates are `wall` (highlighted for `bow-window` when using ADE20k, but not Pascal), `floor` (highlighted for `auto-showroom` when using ADE20k but not Pascal), `dog` (highlighted for `corn-field` when using Pascal, but not ADE20k) and `pole` (highlighted for `hardware-store` for Pascal, but not ADE20k).

However, more than just the base rate, the images themselves look very different across scenes. We visualize random images from different scenes in Figure 2, and find, for example, images labelled `bedroom` in Pascal tend to have either a person or animal sleeping on a bed, without much of the remaining bedroom being shown, whereas ADE20k features images of full bedrooms. Similarly, images labelled `tree-farm` contain people in Pascal, but do not in ADE20k.

Upper bounds.

Finally, we present a simple method to compare the similarity of the probe dataset with that of the training dataset by noting that the probe dataset establishes a strict *upper bound* on the fraction of the model that can be explained. This is intuitively true since the set of semantic labeled concepts is finite, but actually goes deeper than that. Consider the following experiment: we take the original black-box model, run it on a probe dataset to make predictions, and then train a new classifier to emulate those predictions. If this classifier is restricted to use only the labeled concepts then this is similar to a concept-based explanation. However, even if it's trained on the rich underlying visual features it would not perform perfectly due to the differences between the original training dataset and the probe dataset.

Concretely, consider a black-box ResNet18-based [4] model trained on the Places365 [13] dataset. We reset and re-train its final linear classification layer on the Pascal [2] probe dataset to emulate the original scene predictions; this achieves only 63.7% accuracy. Similarly, on the ADE20k [14, 15] as the probe dataset it achieves only slightly better 75.7% accuracy, suggesting that this dataset is somewhat more similar to Places365 than Pascal but still far from fully capturing the distribution. This is not to suggest that the only way to generate concept-based explanations is to collect concept labels for the original training set (which may lead to overfitting); rather, it's important to acknowledge this limitation and quantify the explanation method based on such upper bounds.

Similarly, we can ask how well the Concept Bottleneck model [8] can be explained using the CUB test dataset. However, in this case, since the training and test distributions are (hopefully!) similar, we would expect our upper bound to be reasonably high. We check this with our same set up, and find that this is indeed the case – resetting and retraining the final linear layer, using the model's predictions as our targets achieves an accuracy of 89.3%.

C. Concepts used: more details

Here, we provide additional results regarding learning CUB concepts from Section 4.2 of the main paper. The CUB dataset was used by Concept Bottleneck [8], an interpretable-by-design model. This method learned the concepts as an intermediate layer within the network, and then used these concepts to pretdict the target class. Figure 3 contains the histograms of the normalized AP scores for the 112 concepts from CUB [12] as well as the APs for the target bird classes learned by the model. Similar to learning classifiers for the Broden [1] concepts, we learn a linear model using features from an Imagenet [11] trained Resnet18 [4] model. On average, we see that the bird classes are much better learned than the concepts.

D. Human study details

In Section 5 of the main paper, we discuss the human studies we ran to understand how well humans are able to reason about concept-based explanations as the number of concepts used within the explanation increases. In this section, we provide additional details.

To recap, we compare four types of explanations: (1) concept-based explanations that use 8 concepts, (2) concept-based explanations that use 16 concepts, (3) concept-based explanations that use 32 concepts, and (4) example-based explanations that consist of 10 example images for which the model predicts a certain class. (4) is a baseline that doesn't use concepts.

For a fair comparison, all four types of explanations are evaluated on the same inputs. We generate five sets of input where each set consists of 5 images from one scene group (commercial buildings, shops, markets, cities, and towns) and 5 images from another scene group (home or hotel). Recall that these are images where the model output match the explanation output (i.e., the class with the highest ex-

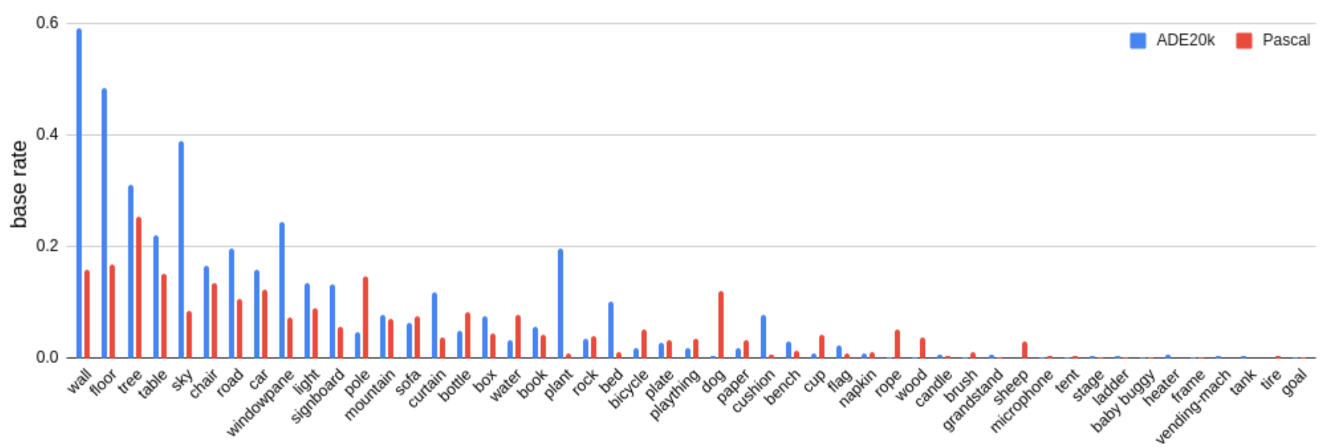


Figure 1. Different concepts have very different base rates across Pascal and ADE20k. The graph shows the base rates for the different concepts highlighted within Table 2 in the main paper.

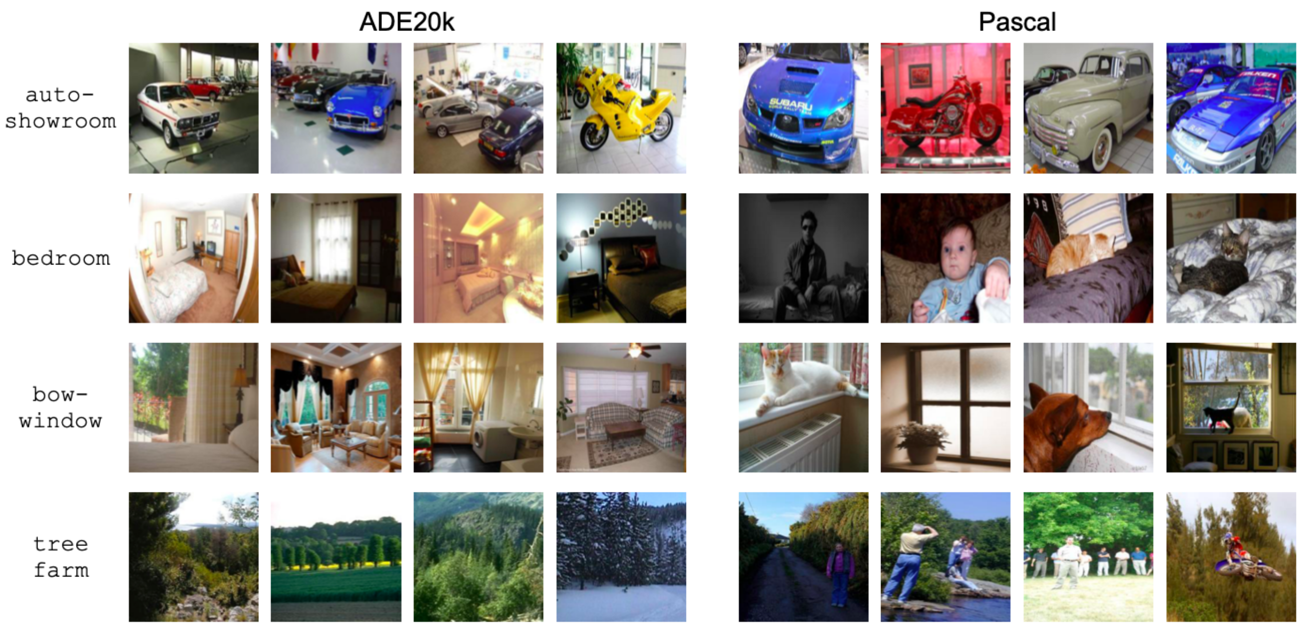


Figure 2. We view a few example images from ADE20k and Pascal for 4 scene classes that had very different explanations in Table 2 from the original paper. We see that these classes have very different distributions; for example, the images labelled as `bedroom` from the Pascal dataset tend to have an animal or person on a bed, whereas the ones from ADE20k do not.

planation score calculated based on ground-truth concept labels). Hence, if the participants correctly identify all concepts that appear in a given image, they are guaranteed to get the highest explanation score for the model output class.

To reduce the variance with respect to the input, we had 5 participants for each set of input and explanation type. For 32 concepts explanations, each participant saw 5 images from only one of the two scene groups because the study got too long and overwhelming with the full set of 10 images. For all other explanations, each participant saw the full set of 10 images. In total, we had 125 participants: 50 participants for the study with 32 concepts explanations and 25 participants for the other three studies. Each participant sees only one type of explanation as we conduct a between-group study.

More specifically, we recruited participants through Amazon Mechanical Turk who are US-based, have done over 1000 Human Intelligence Tasks, and have prior approval rate of at least 98%. The demographic distribution was: man 59%, woman 41%; no race/ethnicity reported 82%, White 17%, Black/African American 1%, Asian 1%. The self-reported machine learning experience was 2.5 ± 1.0, between "2: have heard about..." and "3: know the basics..." We did not collect any personally identifiable information. Participants were compensated based on the state-level minimum wage of \$12/hr. In total, ~\$800 was spent

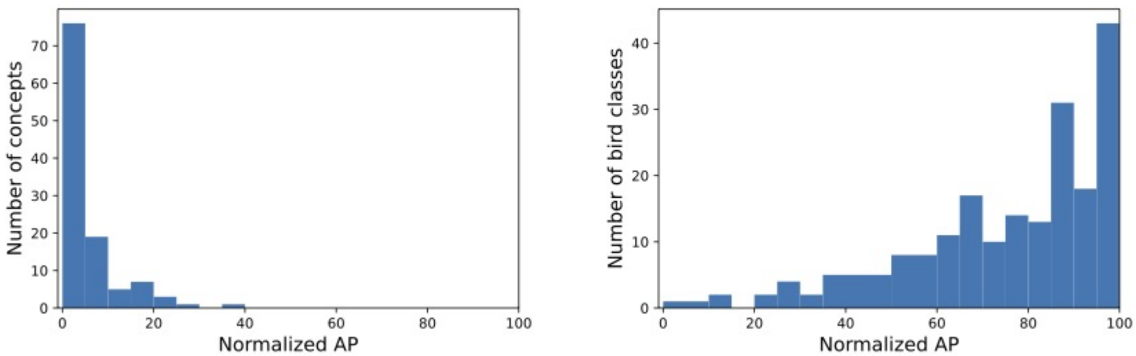


Figure 3. We compare the normalized APs when trying to learn CUB concepts (*left*) to the normalized APs of the CUB target classes for the Concept Bottleneck model(*right*). On average, the concepts are much harder to learn.

on running human studies.

E. User interface snapshots

In Section 5.1 of the main paper, we outlined our human study design.[1] Here we provide snapshots of our study UIs in the following order.

Study introduction.. For each participant, we introduce the study, present a consent form, and receive informed consent for participation in the study. The consent form was approved by our institution's Institutional Review Board and acknowledges that participation is voluntary, refusal to participate will involve no penalty or loss of benefits, etc. See Fig. 4.

Demographics and background.. Following HIVE [7], we request optional demographic data regarding gender identity, race and ethnicity, as well as the participant's experience with machine learning. We collect this information to help future researchers calibrate our results. See Fig. 5.

Method introduction.. We introduce concept-based explanations in simple terms. This page is not shown for the study with example-based explanations. See Fig. 6.

Task preview . We present a practice example to help participants get familiar with the task. This page is not shown for the study with example-based explanations. See Fig. 7.

Part 1: Recognize concepts and guess the model output. After the preview, participants move onto the main task where they are asked to recognize concepts in a given photo (for concept-based explanations) and predict the model output (for all explanations). We show the UI for each type of explanation we study:

- 8 concept explanations (Fig. 8)
- 16 concepts explanations (Fig. 9)
- 32 concepts explanations (Fig. 10)
- Example-based explanations (Fig. 11)

Part 2: Choose the ideal tradeoff between simplicity and correctness.. Concept-based explanations can have varying levels of complexity/simplicity and correctness. Hence, we investigate how participants reason with these two properties. To do so, we show examples of concept-based explanations that use different numbers of concepts, as well as bar plots with the correctness values for certain instantiations of concept-based explanations. We then ask participants to choose the explanation they prefer the most and provide a short written justification for their choice. See Fig. 12.

Feedback.. At the end of the study, participants can optionally provide feedback. See Fig. 13.

[1]We note that much of our study design and UI is based on the recent work by Kim et al. [**?**] who propose a human evaluation framework called HIVE for evaluating visual interpretability methods.

neuron	ADE20k label	Pascal label	Probe dataset: ADE20k		Probe dataset: Pascal	
			IOU ADE20k label	IOU Pascal label	IOU ADE20k label	IOU Pascal label
1	counter	bottle	0.059	0.006	0.006	0.049
4	seat	tvmonitor	0.064	0.0	0.0	0.074
22	bookcase	bus	0.07	0.0	0.0	0.048
47	pool table	airplane	0.135	0.0	0.002	0.079
63	field	muzzle	0.053	0.0	0.0	0.042
73	water	bird	*0.041*	*0.002*	*0.052*	*0.08*
102	bathtub	cat	0.04	0.0	0.0	0.055
104	cradle	bus	0.081	0.0	0.0	0.112
112	field	bus	0.05	0.0	0.0	0.051
120	sidewalk	track	0.042	0.001	0.023	0.075
125	table	wineglass	*0.049*	*0.0*	*0.043*	*0.047*
126	stove	bottle	0.064	0.029	0.005	0.163
131	signboard	body	*0.043*	*0.0*	*0.06*	*0.069*
134	bathtub	boat	0.088	0.001	0.005	0.059
141	skyscraper	cage	0.065	0.001	0.0	0.068
155	mountain	train	0.091	0.0	0.038	0.058
172	car	bus	0.055	0.0	0.015	0.101
173	car	bus	0.052	0.0	0.013	0.099
183	person	horse	0.041	0.016	0.003	0.187
184	cradle	cat	0.046	0.0	0.0	0.042
185	chair	horse	0.077	0.014	0.011	0.153
186	person	bird	0.051	0.001	0.017	0.094
191	swimming pool	pottedplant	0.044	0.0	0.0	0.072
198	pool table	ceiling	0.064	0.035	0.001	0.066
208	shelf	bus	0.047	0.0	0.0	0.062
217	toilet	hair	0.049	0.001	0.0	0.055
218	case	track	0.044	0.001	0.0	0.165
223	house	airplane	0.069	0.0	0.0	0.055
231	grandstand	screen	0.097	0.0	0.007	0.047
234	bridge	train	0.05	0.0	0.014	0.042
239	pool table	horse	0.069	0.011	0.0	0.171
248	bed	tvmonitor	0.127	0.0	0.027	0.063
251	sofa	pottedplant	0.073	0.0	0.033	0.053
257	tent	bus	0.042	0.0	0.005	0.279
260	flower	food	*0.082*	*0.033*	*0.064*	*0.069*
267	apparel	car	0.042	0.023	0.0	0.045
278	field	sheep	0.06	0.0	0.0	0.044
289	pool table	food	0.049	0.024	0.0	0.059
293	shelf	bottle	0.074	0.025	0.0	0.105
298	path	motorbike	0.047	0.0	0.0	0.068
305	waterfall	mountain	*0.057*	*0.049*	*0.0*	*0.047*
309	washer	bus	0.109	0.0	0.013	0.065
322	ball	sheep	0.054	0.0	0.005	0.044
324	mountain	motorbike	0.071	0.0	0.015	0.048
327	waterfall	bird	0.055	0.001	0.0	0.087
337	water	boat	*0.072*	*0.031*	*0.053*	*0.109*
341	sea	boat	0.153	0.014	0.0	0.076
347	palm	bicycle	0.051	0.001	0.0	0.083
354	cradle	chair	0.042	0.03	0.0	0.053
357	rock	sheep	0.058	0.0	0.006	0.061
360	pool table	bird	0.048	0.0	0.0	0.041
379	bridge	bus	0.092	0.0	0.03	0.046
383	bed	curtain	*0.069*	*0.064*	*0.01*	*0.079*
384	washer	bicycle	0.043	0.018	0.0	0.201
387	hovel	train	0.04	0.0	0.0	0.085
408	toilet	bottle	0.045	0.002	0.0	0.099
412	bed	airplane	0.079	0.0	0.008	0.086
413	pool table	motorbike	0.09	0.0	0.003	0.07
415	seat	tvmonitor	0.044	0.0	0.0	0.045
419	bed	tvmonitor	0.061	0.0	0.016	0.054
422	seat	tvmonitor	0.089	0.0	0.0	0.056
434	case	cup	0.047	0.001	0.0	0.041
444	sofa	plant	0.045	0.009	0.014	0.09
446	pool table	tvmonitor	0.193	0.0	0.006	0.086
475	cradle	train	0.061	0.0	0.0	0.132
477	desk	tvmonitor	0.104	0.0	0.0	0.085
483	swivel chair	horse	0.052	0.006	0.0	0.041
485	sofa	airplane	0.056	0.0	0.024	0.045
502	washer	train	0.07	0.0	0.006	0.134

Table 2. For all neurons from Tab. 1 that output radically different concepts when explanations are computed using ADE20k vs Pascal, we compute the IOU scores for the other concept as well. Other than the 7 attributes marked in red, the IOU scores are all below 0.04, suggesting that this is not because the neurons are polysemantic.

Concept	ADE20k AUC	Pascal AUC	Cos.sim.	Concept	ADE20k AUC	Pascal AUC	Cos.sim.
bag	79.4	75.4	0.006	book	90.4	84.6	0.138
bottle	88.5	85.6	0.035	box	83.0	80.1	0.086
building	97.4	90.0	0.161	cabinet	91.3	92.4	0.03
car	96.9	90.3	0.147	ceiling	96.6	93.0	0.267
chair	90.5	89.6	0.034	curtain	91.6	89.5	0.112
door	81.5	87.8	0.134	fence	86.1	84.7	0.09
floor	97.4	92.1	0.208	grass	95.1	91.7	0.04
light	92.4	85.0	0.043	mountain	94.2	90.8	0.02
painting	94.8	91.4	0.116	person	92.2	92.1	0.253
plate	90.6	94.8	-0.009	pole	89.0	79.3	0.059
pot	79.3	85.2	0.142	road	98.0	91.8	0.041
rock	92.6	82.8	-0.024	sidewalk	97.0	92.5	0.071
signboard	90.6	76.5	0.091	sky	98.9	79.8	0.104
sofa	95.9	91.2	-0.009	table	93.4	93.5	0.06
tree	96.8	89.2	0.172	wall	95.9	91.3	0.027
water	95.2	94.6	0.078	windowpane	91.5	90.1	0.078

Table 3. We report the cosine similarities between the concept activation vectors learned using ADE20k and Pascal datasets. In general, the vectors learned from different datasets do not correlate well.

References

[1] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*, 2017. 1, 2, 3

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*, 2010. 1, 3

[3] Ruth Fong and Andrea Vedaldi. Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *CVPR*, 2018. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 3

[5] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *ECCV*, 2012. 1

[6] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *ICML*, 2018. 1

[7] Sunnie S. Y. Kim, Nicole Meister, Vikram V. Ramaswamy, Ruth Fong, and Olga Russakovsky. HIVE: Evaluating the human interpretability of visual explanations. In *ECCV*, 2022. 5

[8] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *ICML*, 2020. 3

[9] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. 2

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *JMLR*, 2011. 1

[11] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. In *IJCV*, 2015. 1, 3

[12] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 3

[13] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017. 1, 3

[14] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20k dataset. In *CVPR*, 2017. 1, 3

[15] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20k dataset. *IJCV*, 2019. 1, 3

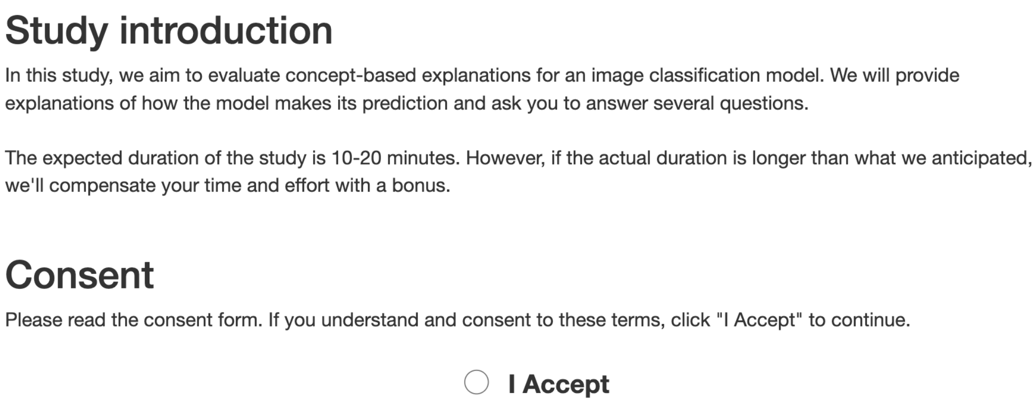


Figure 4. **UI - Study introduction**

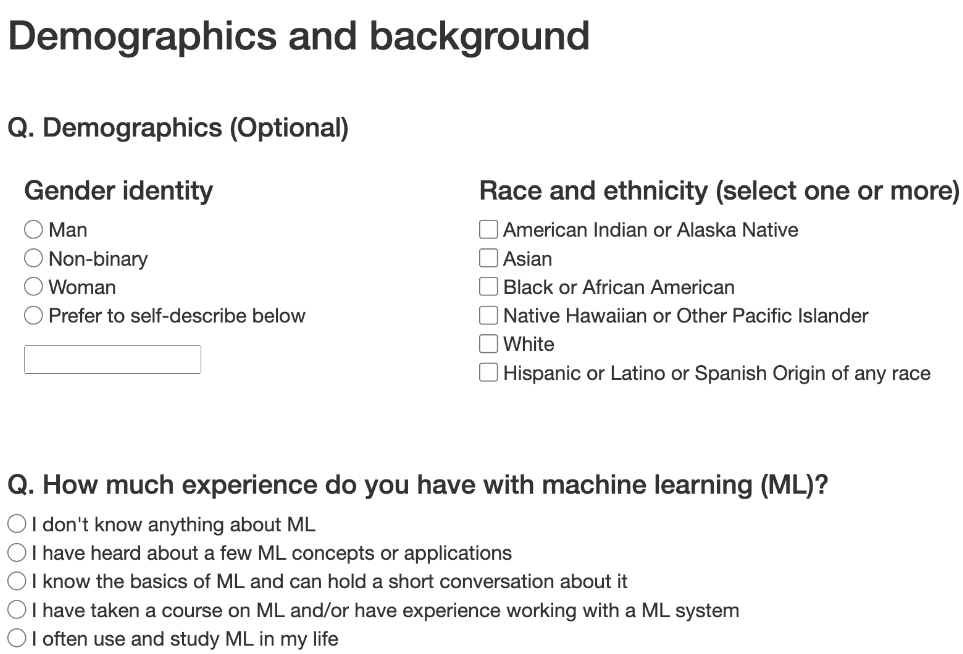


Figure 5. **UI - Demographics and background**

Concept-based explanations

We have a model that recognizes scenes in photos. The model predicts **golf course** for some photos, **park** for some other photos, **church** for some other photos, **supermarket** for some other photos, and so on.

One way of explaining the model's predictions is to use a set of concepts (e.g., things, stuff) and see which concepts are important to be present vs. absent in a photo for the model to predict a certain scene.

Example

For example, one explanation might say the model predicts the scene **golf course** based on the following concepts:

`golf course = + 2.61 grass - 1.94 building + 1.85 sky - 1.75 wall + 1.03 person - 0.87 road + 3.71`

According to this explanation, the model is more likely to predict the scene **golf course** when `grass, sky, person` are present and when `building, wall, road` are absent in a photo. The last constant (`+ 3.71`) balances the explanation scores between different classes.

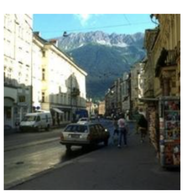
Using scene-level explanations to explain individual predictions

For a given photo, we can recognize which concepts are present and add up their coefficients to get an explanation score.

For example in the below photo, `grass, sky` are present while `building, wall, person, road` are absent. So the explanation score for **golf course** is 8.17 = 2.61 + 1.85 + 3.71.

Concepts	Explanation for `golf course`
☑ grass	= 8.17
☐ building	= + 2.61 x 1 (grass)
☑ sky	− 1.94 x 0 (building)
☐ wall	+ 1.85 x 1 (sky)
☐ person	− 1.75 x 0 (wall)
☐ road	+ 1.03 x 0 (person)
	− 0.87 x 0 (road)
	+ 3.71

For the below photo, `building, sky, person, road` are present while `grass, wall` are absent. So the explanation score for **golf course** is 3.78 = - 1.94 + 1.85 + 1.03 - 0.87 + 3.71. For comparison, we also show the explanation for **street**. The explanation score for **street** is 5.83 = 2.11 + 2.06 + 1.50 - 0.88 + 1.04. Since 5.83 is higher than 3.78, according to these explanations it is more likely for the model to predict **street** for this photo.



Concepts	Explanation for `golf course`	Explanation for `street`
☐ grass	= 3.78	= 5.83
☑ building	= + 2.61 x 0 (grass)	= + 2.11 x 1 (building)
☑ sky	− 1.94 x 1 (building)	+ 2.06 x 1 (person)
☐ wall	+ 1.85 x 1 (sky)	− 1.87 x 0 (grass)
☑ person	− 1.75 x 0 (wall)	+ 1.50 x 1 (road)
☑ road	+ 1.03 x 1 (person)	− 0.88 x 1 (sky)
	− 0.87 x 1 (road)	− 0.68 x 0 (wall)
	+ 3.71	+ 1.04

Figure 6. **UI - Method introduction**

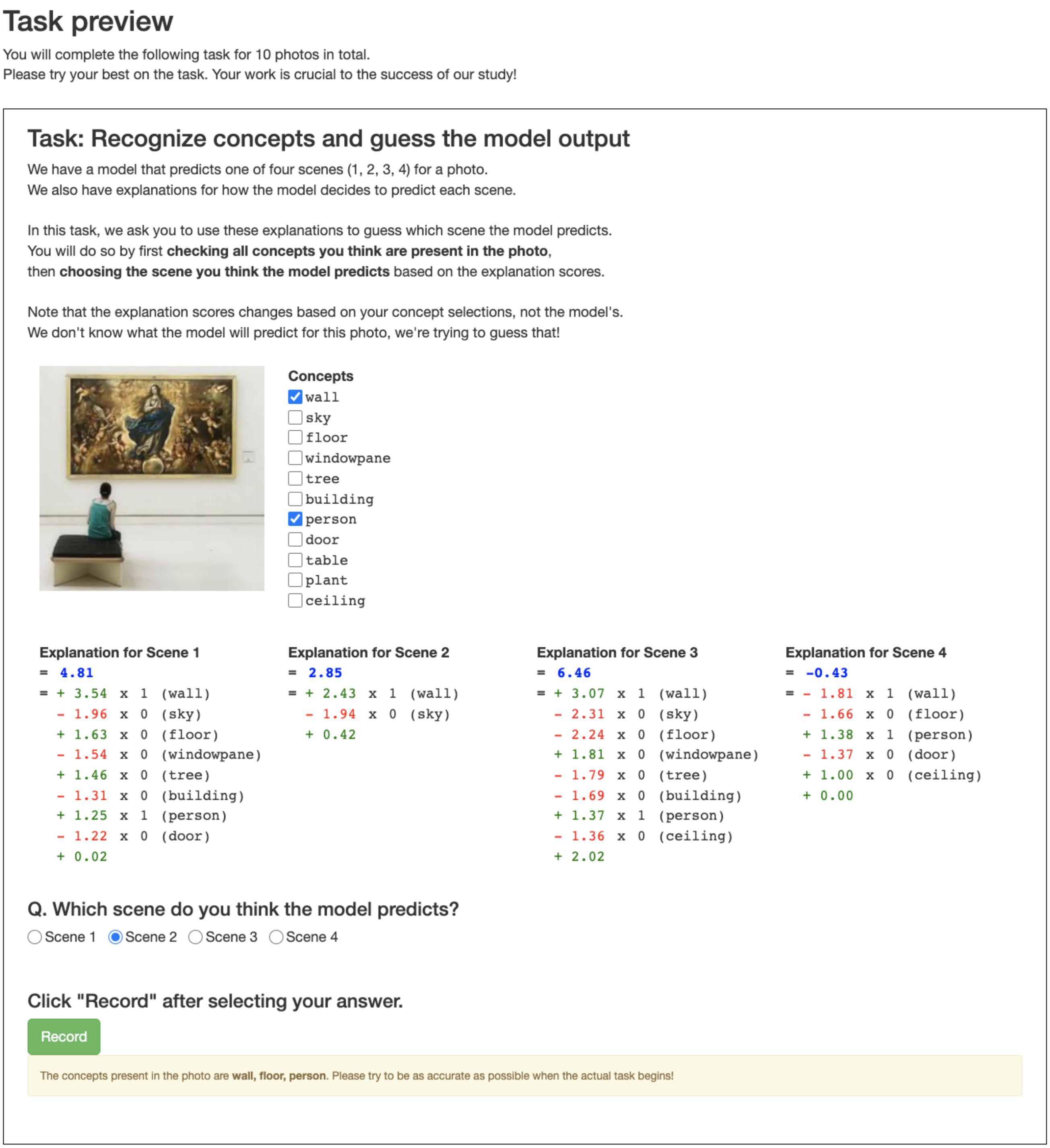


Figure 7. **UI - Task preview**

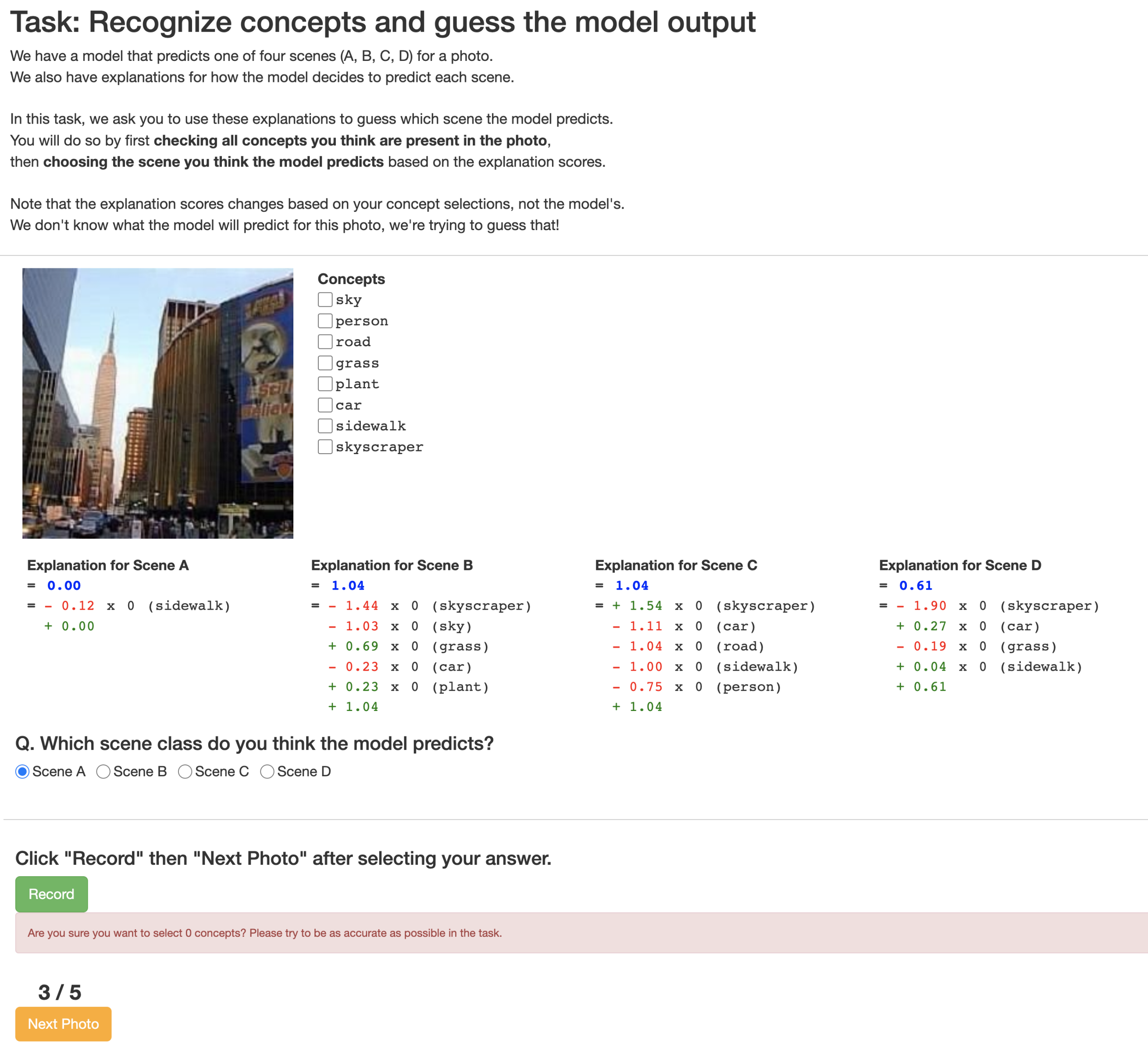


Figure 8. **UI - Part 1: Recognize concepts and guess the model output (8 concepts explanations)**

Task: Recognize concepts and guess the model output

We have a model that predicts one of four scenes (W, X, Y, Z) for a photo.
We also have explanations for how the model decides to predict each scene.

In this task, we ask you to use these explanations to guess which scene the model predicts.
You will do so by first **checking all concepts you think are present in the photo**,
then **choosing the scene you think the model predicts** based on the explanation scores.

Note that the explanation scores changes based on your concept selections, not the model's.
We don't know what the model will predict for this photo, we're trying to guess that!

Concepts

- [x] wall
- [x] floor
- [ ] windowpane
- [ ] table
- [ ] plant
- [ ] chair
- [ ] carpet
- [ ] lamp
- [x] bed
- [ ] sofa
- [x] cushion
- [ ] vase
- [ ] armchair
- [ ] sconce
- [ ] coffee table
- [ ] fireplace

Explanation for Scene W

```
= 1.88
= + 1.88 x 1 (bed)
  - 0.95 x 0 (chair)
  - 0.60 x 0 (sofa)
  - 0.28 x 0 (armchair)
  - 0.04 x 0 (table)
  - 0.03 x 0 (sconce)
  + 0.00
```

Explanation for Scene X

```
= -2.60
= - 3.20 x 1 (bed)
  + 1.47 x 0 (chair)
  - 1.38 x 0 (sofa)
  - 0.80 x 1 (cushion)
  - 0.39 x 0 (coffee table)
  - 0.14 x 0 (armchair)
  - 0.14 x 0 (lamp)
  + 1.40
```

Explanation for Scene Y

```
= 1.27
= + 1.36 x 1 (bed)
  - 1.02 x 0 (windowpane)
  - 0.92 x 1 (wall)
  - 0.31 x 0 (plant)
  - 0.24 x 0 (carpet)
  + 0.19 x 0 (sconce)
  - 0.18 x 1 (floor)
  - 0.15 x 1 (cushion)
  - 0.11 x 0 (vase)
  + 1.16
```

Explanation for Scene Z

```
= -0.54
= + 2.00 x 0 (sofa)
  - 1.73 x 1 (bed)
  - 0.88 x 0 (table)
  + 0.68 x 0 (coffee table)
  - 0.52 x 0 (chair)
  - 0.38 x 1 (wall)
  + 0.30 x 0 (armchair)
  + 0.20 x 0 (fireplace)
  + 0.17 x 1 (cushion)
  + 1.40
```

Q. Which scene class do you think the model predicts?

(•) Scene W ( ) Scene X ( ) Scene Y ( ) Scene Z

Click "Record" then "Next Photo" after selecting the rows and answering the question.

Record

1 / 5

Next Photo

Figure 9. **UI - Part 1: Recognize concepts and guess the model output (16 concepts explanations)**

CVPR #1118

Task: Recognize concepts and guess the model output

We have a model that predicts one of four scenes (A, B, C, D) for a photo.
We also have explanations for how the model decides to predict each scene.

In this task, we ask you to use these explanations to guess which scene the model predicts.
You will do so by first **checking all concepts you think are present in the photo**,
then **choosing the scene you think the model predicts** based on the explanation scores.

Note that the explanation scores changes based on your concept selections, not the model's.
We don't know what the model will predict for this photo, we're trying to guess that!

Concepts

- [ ] wall
- [x] sky
- [ ] floor
- [ ] tree
- [ ] person
- [ ] road
- [ ] grass
- [ ] plant
- [ ] car
- [ ] sidewalk
- [ ] mountain
- [ ] streetlight
- [ ] box
- [ ] earth
- [ ] rock
- [ ] pot

(continued)

- [ ] flowerpot
- [ ] stairs
- [ ] bag
- [ ] ashcan
- [ ] spotlight
- [ ] stairway
- [ ] van
- [ ] truck
- [ ] awning
- [ ] traffic light
- [ ] flag
- [ ] bucket
- [ ] pedestal
- [ ] trade name
- [ ] palm
- [x] skyscraper

Explanation for Scene A

```
= 1.62
= + 1.15 x 0 (flag)
  + 1.12 x 1 (skyscraper)
  - 0.99 x 0 (awning)
  + 0.86 x 0 (earth)
  + 0.83 x 0 (floor)
  + 0.80 x 0 (car)
  - 0.76 x 0 (pot)
  + 0.63 x 0 (trade name)
  + 0.57 x 0 (traffic light)
  + 0.55 x 0 (wall)
  + 0.55 x 0 (streetlight)
  - 0.53 x 0 (sidewalk)
  + 0.51 x 0 (stairs)
  + 0.50 x 1 (sky)
  + 0.43 x 0 (truck)
  + 0.40 x 0 (pedestal)
  - 0.39 x 0 (ashcan)
  + 0.37 x 0 (grass)
  + 0.32 x 0 (road)
  + 0.32 x 0 (flowerpot)
  + 0.32 x 0 (tree)
  + 0.26 x 0 (bag)
  - 0.26 x 0 (van)
  + 0.26 x 0 (palm)
  + 0.24 x 0 (bucket)
  - 0.19 x 0 (person)
  - 0.04 x 0 (spotlight)
  + 0.00
```

Explanation for Scene B

```
= -2.07
= - 3.74 x 1 (skyscraper)
  + 2.13 x 0 (stairway)
  + 1.73 x 0 (grass)
  - 1.37 x 1 (sky)
  + 1.26 x 0 (palm)
  - 0.90 x 0 (truck)
  + 0.89 x 0 (rock)
  + 0.89 x 0 (plant)
  - 0.84 x 0 (box)
  - 0.79 x 0 (car)
  - 0.48 x 0 (flowerpot)
  + 0.44 x 0 (flag)
  + 0.40 x 0 (traffic light)
  - 0.34 x 0 (streetlight)
  + 0.30 x 0 (road)
  - 0.28 x 0 (van)
  - 0.26 x 0 (mountain)
  + 0.20 x 0 (sidewalk)
  - 0.19 x 0 (spotlight)
  - 0.16 x 0 (awning)
  - 0.15 x 0 (bag)
  + 0.13 x 0 (ashcan)
  - 0.07 x 0 (stairs)
  - 0.01 x 0 (trade name)
  + 3.04
```

Explanation for Scene C

```
= 5.71
= - 2.69 x 0 (person)
  + 2.11 x 1 (skyscraper)
  - 1.71 x 0 (car)
  - 1.42 x 0 (road)
  - 1.41 x 0 (sidewalk)
  + 0.56 x 1 (sky)
  - 0.48 x 0 (wall)
  - 0.31 x 0 (tree)
  - 0.30 x 0 (streetlight)
  - 0.26 x 0 (flag)
  + 3.04
```

Explanation for Scene D

```
= -1.70
= - 2.73 x 1 (skyscraper)
  - 1.88 x 0 (grass)
  - 1.07 x 0 (flag)
  + 1.01 x 0 (road)
  - 0.92 x 0 (stairway)
  - 0.78 x 0 (traffic light)
  + 0.69 x 0 (sidewalk)
  + 0.68 x 0 (car)
  + 0.66 x 0 (awning)
  - 0.60 x 0 (plant)
  + 0.48 x 0 (person)
  + 0.41 x 0 (van)
  + 0.40 x 1 (sky)
  - 0.38 x 0 (palm)
  - 0.30 x 0 (wall)
  - 0.23 x 0 (earth)
  + 0.19 x 0 (spotlight)
  - 0.11 x 0 (trade name)
  + 0.07 x 0 (mountain)
  + 0.63
```

Q. Which scene class do you think the model predicts?

( ) Scene A ( ) Scene B (•) Scene C ( ) Scene D

Click "Record" then "Next Photo" after selecting your answer.

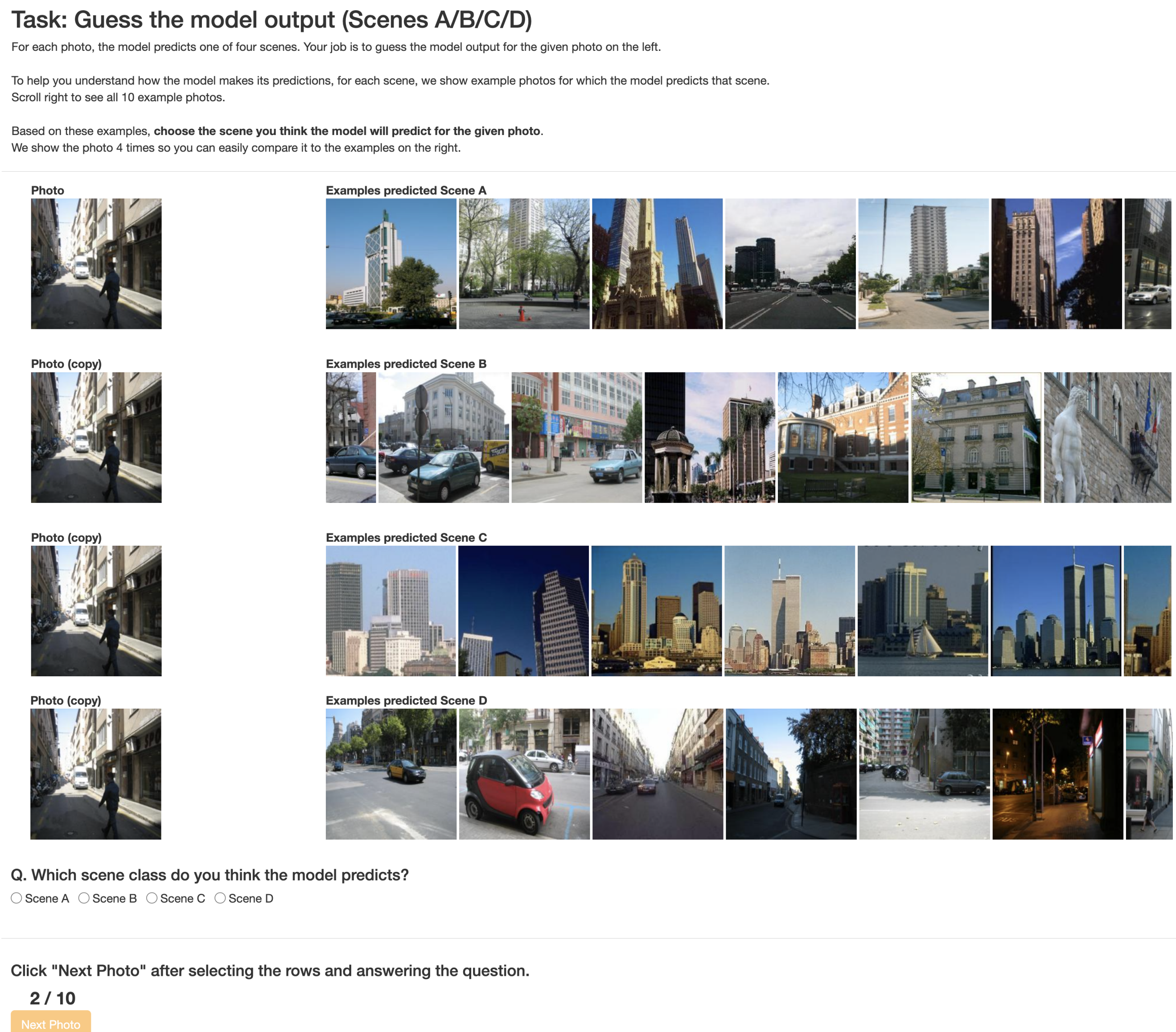


Figure 11. **UI - Part 1: Guess the model output (example-based explanations)**

Simplicity-Correctness Tradeoff

So far we have only shown photos where the scene with the highest explanation score matches the scene the model predicts.

However, this is not always the case, and you can choose the level of **simplicity** and **correctness** of concept-based explanation.

Simplicity refers to the number of concepts used in a given set of explanations.
Correctness refers to the percentage of times the explanations correctly explain the model prediction.

For reference, we show how explanations for scene **embassy** look when they use different number of concepts.

Up to 4 concepts

```
embassy =
- 0.39 sky
+ 0.33
```

Up to 8 concepts

```
embassy =
- 1.16 sky
- 1.15 grass
- 0.24 car
+ 0.24 skyscraper
+ 1.29
```

Up to 16 concepts

```
embassy =
- 2.88 sky
+ 1.49 grass
+ 1.48 plant
- 1.30 car
+ 0.68 streetlight
- 0.60 stairway
- 0.55 truck
+ 0.53 awning
- 0.22 traffic light
+ 0.19 flag
- 0.08 palm
+ 0.07 skyscraper
+ 2.36
```

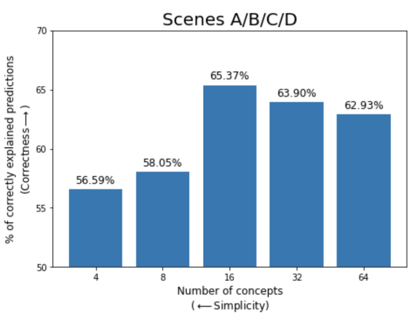
Up to 32 concepts

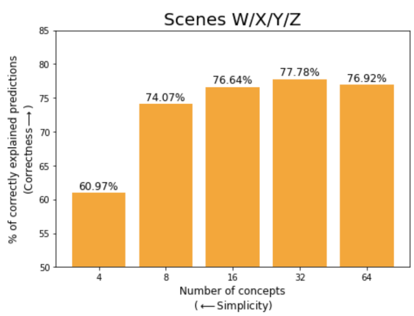
```
embassy =
- 3.46 sky
+ 1.99 road
+ 1.65 grass
- 1.37 plant
+ 1.09 car
+ 0.81 sidewalk
- 0.78 mountain
- 0.72 fence
+ 0.52 streetlight
- 0.46 box
+ 0.36 rock
- 0.34 flowerpot
+ 0.28 ashcan
- 0.25 spotlight
- 0.18 stairway
- 0.16 van
+ 0.14 truck
- 0.09 awning
- 0.08 traffic light
+ 0.04 flag
- 0.03 palm
+ 0.02 skyscraper
+ 3.12
```

Up to 64 concepts

```
embassy =
- 12.92 wall
+ 10.43 sky
+ 8.83 floor
+ 6.41 windowpane
+ 5.92 tree
- 5.87 building
- 5.69 person
- 5.60 door
- 5.59 table
+ 5.51 road
- 4.29 grass
- 4.09 plant
- 3.44 chair
+ 3.02 car
+ 2.87 painting
- 2.79 carpet
- 2.29 sidewalk
+ 2.26 signboard
- 1.92 mirror
- 1.80 lamp
+ 1.66 curtain
+ 1.56 pole
- 1.54 mountain
+ 1.54 fence
- 1.51 streetlight
+ 1.41 box
+ 1.37 earth
- 1.32 water
+ 1.31 railing
- 1.31 flower
+ 1.24 rock
+ 1.16 pot
+ 1.13 flowerpot
- 0.99 stairs
+ 0.92 clock
+ 0.83 bag
- 0.82 pillar
- 0.80 bicycle
+ 0.77 ashcan
+ 0.76 bench
+ 0.60 spotlight
- 0.56 basket
+ 0.55 path
- 0.54 stairway
+ 0.54 van
- 0.46 truck
- 0.41 awning
- 0.33 traffic light
- 0.33 bannister
+ 0.33 poster
- 0.31 flag
- 0.25 drinking glass
- 0.24 bucket
- 0.23 pedestal
- 0.20 trade name
+ 0.10 palm
+ 0.06 air conditioner
- 0.03 skyscraper
+ 7.70
```

Below is a plot that visualizes the tradeoff between simplicity and correctness. Overall, we see that explanations that use more concepts better explain model predictions.





Q. Which would you prefer?

◯ Explanations that use up to 4 concepts
◯ Explanations that use up to 8 concepts
◯ Explanations that use up to 16 concepts
◯ Explanations that use up to 32 concepts
◯ Explanations that use up to 64 concepts

Briefly describe the reason for your choice.

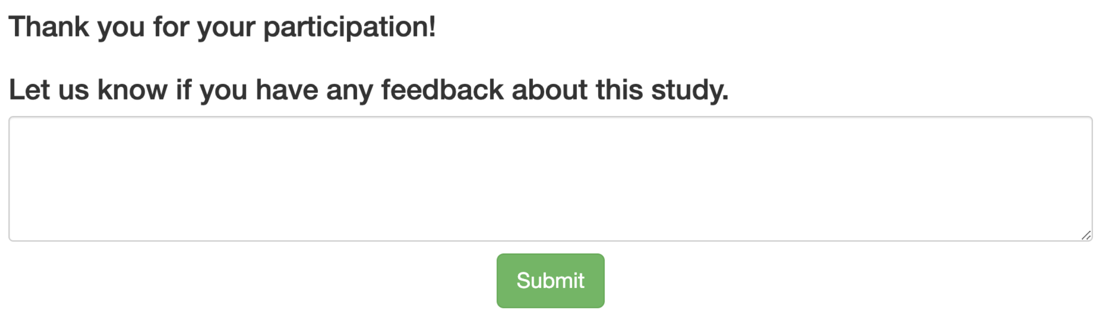


Figure 13. **UI - Feedback**