

A. Concepts

We provide the full list of concepts, along with the text phrases provided by the users. Each concept name was automatically added to the list of positive text phrases.

1. `gourmet tuna`
 - (a) Positive text phrases: tuna sushi, seared tuna, tuna sashimi
 - (b) Negative text phrases: canned tuna, tuna sandwich, tuna fish, tuna fishing
2. `emergency service`
 - (a) Positive text phrases: firefighting, paramedic, ambulance, disaster worker, search and rescue
 - (b) Negative text phrases: construction, crossing guard, military
3. `healthy dish`
 - (a) Positive text phrases: salad, fish dish, vegetables, healthy food
 - (b) Negative text phrases: fast food, fried food, sugary food, fatty food
4. `in-ear headphones`
 - (a) Positive text phrases: in-ear headphones, airpods, earbuds
 - (b) Negative text phrases: earrings, bone headphones, over-ear headphones
5. `hair coloring`
 - (a) Positive text phrases: hair coloring service, hair coloring before and after
 - (b) Negative text phrases: hair coloring product
6. `arts and crafts`
 - (a) Positive text phrases: kids crafts, scrapbooking, hand made decorations
 - (b) Negative text phrases: museum art, professional painting, sculptures
7. `home fragrance`
 - (a) Positive text phrases: home fragrance flickr, scented candles, air freshener, air freshener flickr, room fragrance, room fragrance flickr, scent sachet, potpourri, potpourri flickr
 - (b) Negative text phrases: birthday candles, birthday candles flickr, religious candles, religious candles flickr, car freshener, car freshener flickr, perfume, perfume flickr
8. `single sneaker on white background`
 - (a) Positive text phrases: one sneaker on white background

- (b) Negative text phrases: two sneakers on white background, leather shoe
9. `dance`
 - (a) Positive text phrases: ballet, tango, ballroom dancing, classical dancing, professional dance
 - (b) Negative text phrases: sports, fitness, zumba, ice skating
10. `hand pointing`
 - (a) Positive text phrases: hand pointing, meeting with pointing hand, cartoon hand pointing, pointing at screen
 - (b) Negative text phrases: thumbs up, finger gesture, hands, sign language
11. `astronaut`
 - (a) Positive text phrases: female astronaut, spacecraft crew, space traveler
 - (b) Negative text phrases: spacecraft, space warrior, scuba diver
12. `stop sign`
 - (a) Positive text phrases: stop sign in traffic, stop sign held by a construction worker, stop sign on a bus, stop sign on the road, outdoor stop sign, stop sign in the wild
 - (b) Negative text phrases: indoor stop sign, slow sign, traffic light sign, stop sign on a poster, stop sign on the wall, cartoon stop sign, stop sign only
13. `pie chart`
 - (a) Positive text phrases: pie-chart
 - (b) Negative text phrases: pie, bar chart, plot
14. `block tower`
 - (a) Positive text phrases: toy tower
 - (b) Negative text phrases: tower block, building

B. Evaluation strategy

Because we are eliciting the concept from users, only they can correctly label every image. Therefore, when generating an evaluation set, the annotations must come from the user. However, since our users are real people with real time restrictions, this means that we cannot ask them to exhaustively rate a large evaluation set. We target less than 1000 images for each concept's evaluation set.

B.1. Proposed evaluation strategies

We considered the following strategies for evaluation:

Labeling the entire unlabeled set. The most accurate evaluation metric is to label the entire unlabeled set. However, this is infeasible, as the user would have to label hundreds of millions of images.

Random sampling from unlabeled set. To reduce the number of images to label, we could randomly sample until we hit a desired amount. However, since most of the concepts are rare ($< 0.1\%$ of the total amount of data), this means our evaluation set would have very few positives.

Holdout of training data. As the user labels new ground truth, hold out a fraction of it for evaluation. The benefit is that the user does not have to label any extra data. The main detriment is that the evaluation set comes from the exact same distribution as the training set, leading to overestimates of performance, as there are no new visual modes in the evaluation set.

Random sampling at fixed prediction frequencies. Choose a set of operating points. For each operating point randomly sample K images with score higher than that operating point. The operating points can be selected as the model prediction frequency—for example, we can calculate precision of the highest confidence 100, 1000, and 10000 predictions. The metric that will be directly comparable across models is precision vs prediction frequency. To minimize rating cost we can use the deterministic hash approach. The main problem is that the choice of operating points varies depending on the particular class. Classes that are rare or harder to correctly predict may need stricter operating points than common and easy classes. Furthermore, with this approach we cannot compute a PR curve, just some metrics at specific operating points.

Stratified sampling without weights [our chosen approach]. Collect new evaluation images by (1) calculating model scores, (2) bucketing the images by model score (e.g., $[0, 0.1)$, $[0.1, 0.2)$, ..., $[0.8, 0.9)$, $[0.9, 1]$), (3) rating k examples per bucket. To minimize any bias towards any particular model, we can repeat this process to retrieve an evaluation set per model and merge to get the final evaluation set. Additionally, we can use a deterministic hash instead of random sampling to encourage high overlap across the images chosen to save on the total rating budget. The major upside is that, using a small number of images rated, we can get a relatively balanced dataset of positives and negatives, while also mining for hard examples to stress test the models. The main limitations of this method are:

1. Stratified sampling requires good bucket boundaries to work well, which is not guaranteed.

2. The metric will be biased since samples selected from buckets with a smaller number of candidates (such as the $[0.9, 1]$ bucket) will have more influence than samples from buckets with lots of candidates (e.g. the $[0, 0.1)$ bucket).
3. Merging image sets from multiple models may bias towards the models make common predictions. However, we hope that pseudorandom hashing selects the same images and prevents this from occurring.

Stratified sampling with weights. This involves the same process as stratified sampling without weights, but whenever computing a metric, you weigh the sample by the distribution of scores it came from. This unbiases sampling from each strata, but for very large buckets (e.g., the $[0, 0.1)$ bucket), the weight would be extremely large. This means that predicting incorrectly on any of these images overpowers all correct predictions on other buckets.

Based on the pros and cons of all these approaches, we chose *stratified sampling without weights* for our experiments, which we believe is most representative for our problem setting.

B.2. Evaluation set statistics

In Table 2, we show that our stratified sampling method chooses a tractable number of images to rate, while keeping the positive and negative count relatively balanced.

Concept Name	# Images	Pos. Rate
arts and crafts	707	0.66
astronaut	637	0.36
block tower	669	0.36
dance	730	0.47
emergency service	675	0.50
gourmet tuna	576	0.27
hair-coloring	645	0.67
hand-pointing	832	0.34
healthy dish	633	0.36
home-fragrance	716	0.39
in-ear-headphones	687	0.42
pie-chart	594	0.42
single sneaker on white background	556	0.49
stop sign	704	0.44

Table 2: Statistics showing the number of images and the positive rate in each concept’s evaluation set.

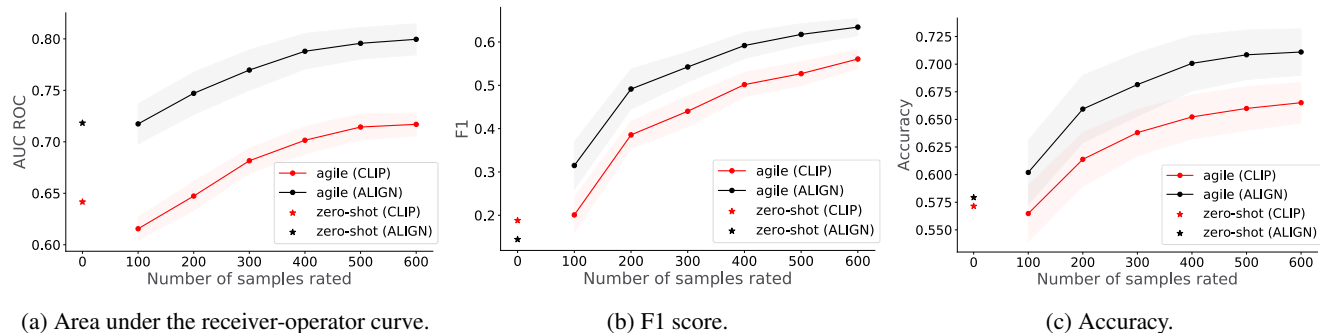


Figure 9: Model performance per amount of samples rated by the user. Mean and standard error over all concepts, for multiple metrics.

C. Additional active learning results

C.1. Additional metrics

We include here additional active learning results, measuring the amount of rating by user versus model performance. Figure 9 shows the results in terms of AUC ROC, F1 score, and accuracy. Note that, unlike AUC PR and AUC ROC, for computing the F1 score and accuracy one must choose a threshold on the model prediction score that determines whether a sample is on the positive or negative side of the decision boundary. For our trained MLP models, we used the common 0.5 threshold. For the zero-shot models, the threshold 0.5 is not a good choice, because the cosine similarities for both positive and negative are often smaller than this. In fact, [52] did an analysis of the right choice of threshold based on a human inspection on LAION-5B, and they recommend using the threshold 0.28 when using CLIP embeddings; we also use this threshold. We similarly chose 0.2 as a threshold when using ALIGN based on our own inspection.

Based on the results in Figure 9, we noticed the same consistent observations with all metrics: (1) the performance increases with every active learning round; (2) the performance increase is faster in the beginning, and starting to plateau in the later AL rounds; (3) the models that use ALIGN embeddings are consistently better than those using CLIP.

C.2. Margin versus Margin & Positive Mining

We show in detail the results per concept for the two active learning strategies considered in our paper: *margin sampling* and the *margin sampling & positive mining* of [39]. The results are shown in Figure 10. We observe that for the majority of the concepts the two methods are very close. Some exceptions include the concepts `healthy dish` and `hand pointing` for which *margin sampling* performs better, while for `block tower` *margin sampling & positive mining* works better. Overall it is not clear that one

method is significantly better than the other.

D. Concept difficulty

To be unbiased with respect to whom the rater is—whether it is the user or crowd raters—we decided to measure concept difficulty as the performance of a zero-shot model. We show the performance of the zero-shot model using CLIP embeddings for each concept, measured in terms of AUC PR on the test set, in Table 3.

Concept	Score
<code>gourmet tuna</code>	0.37
<code>healthy dish</code>	0.46
<code>hand-pointing</code>	0.47
<code>astronaut</code>	0.48
<code>block tower</code>	0.49
<code>home-fragrance</code>	0.50
<code>stop sign</code>	0.51
<code>emergency service</code>	0.53
<code>in-ear-headphones</code>	0.55
<code>single sneaker on white background</code>	0.56
<code>dance</code>	0.61
<code>pie-chart</code>	0.66
<code>hair-coloring</code>	0.73
<code>arts and crafts</code>	0.74

Table 3: Difficulty score per concept, estimated as AUC PR of the zero-shot model using CLIP embeddings.

With these scores, we can group the top 7 easiest and top 7 hardest concepts:

- **top 7 easiest concepts:** `emergency service`, `in-ear-headphones`, `single sneaker on white background`, `dance`, `pie-chart`, `hair-coloring`, `arts and crafts`
- **top 7 hardest concepts:** `gourmet tuna`, `healthy dish`, `hand-pointing`, `astronaut`, `block tower`, `home-fragrance`, `stop sign`

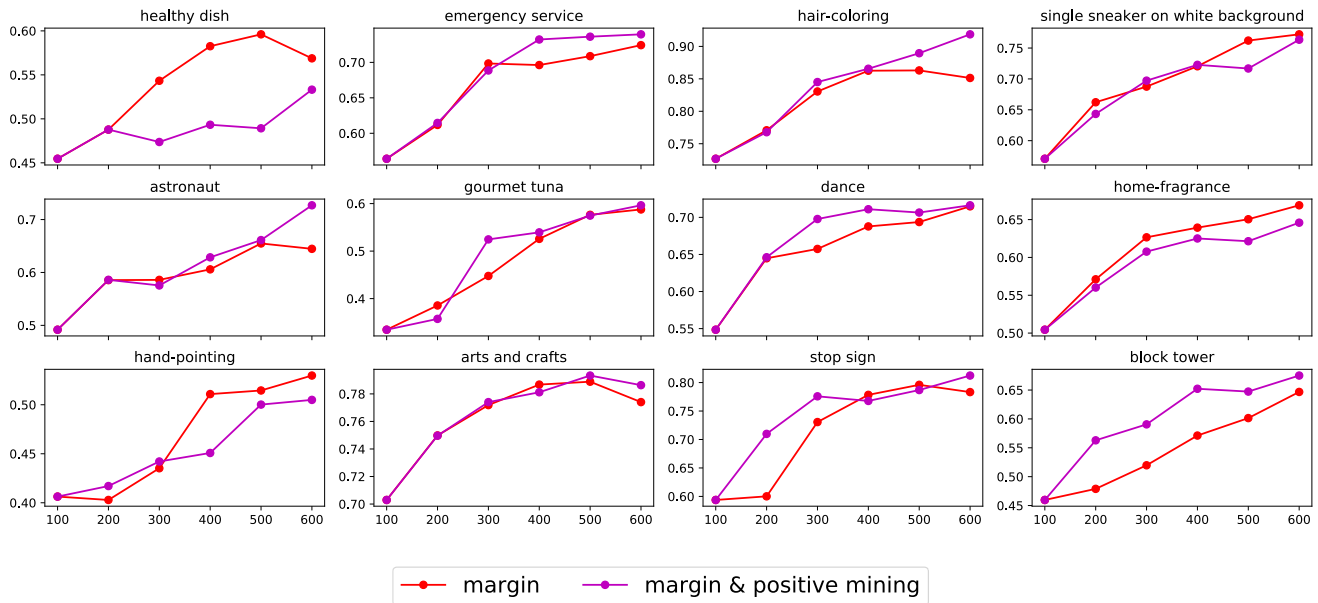



Figure 10: Results per concept for margin vs margin & positive mining of [39]. The each figure shows the AUC PR (on y-axis) for each active learning round (on x-axis) for the two methods.

After reading the description and examples, please label if the displayed image belongs to the class. If you are not sure, mark "Don't know".

Consider the image below.

Is this an image of "Astronaut"?


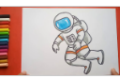




Description: *Any picture that shows a astronaut, even if it's a drawing, clip art, etc. The astronaut should show clearly that they are associated with being an astronaut – usually indicated by a space suit or NASA jumpsuit.*



[j] Yes
 [k] No
 [l] No Image

Submit
[Enter]

POSITIVE EXAMPLES (YES)

 ✓ This is clearly an astronaut	 ✓ Astronaut drawings are ok	 ✓ Astronauts don't need to be wearing their space suit. Often, they will be wearing NASA (or other space program associated) jump suits.	 ✓ Buzz Lightyear is a fictional astronaut	 ✓ Astronaut in a small region is okay	 ✓ It can tell from the scene even without wearing a spacesuit.
---	--	---	--	--	---

NEGATIVE EXAMPLES (NO)







 ✗ Even though Mark Kelly is an astronaut, he isn't wearing anything that depicts him to be one.	 ✗ This is a deep dive suit, not an astronaut.	 ✗ This is a child wearing an astronaut costume.	 ✗ This is a space soldier with armor.	 ✗ This is not a person	 ✗ It cannot tell if the persons are astronauts from the scene.
--	--	--	--	---	---

Figure 11: An example template we use for crowd labeling, for the astronaut concept.

E. Crowd task design

Crowd workers are onboarded to the binary image classification task then given batches of images to label, where each batch contains images from the same concept type to minimize cross-concept mislabeling. In Figure 11 we show

the task we present to crowd workers for image classification. The template contains the image to classify, as well as a description of the image concept and a set of positive and negative examples created by the user who created the concept. Each image is sent to three crowd workers and the label is decided by majority vote.

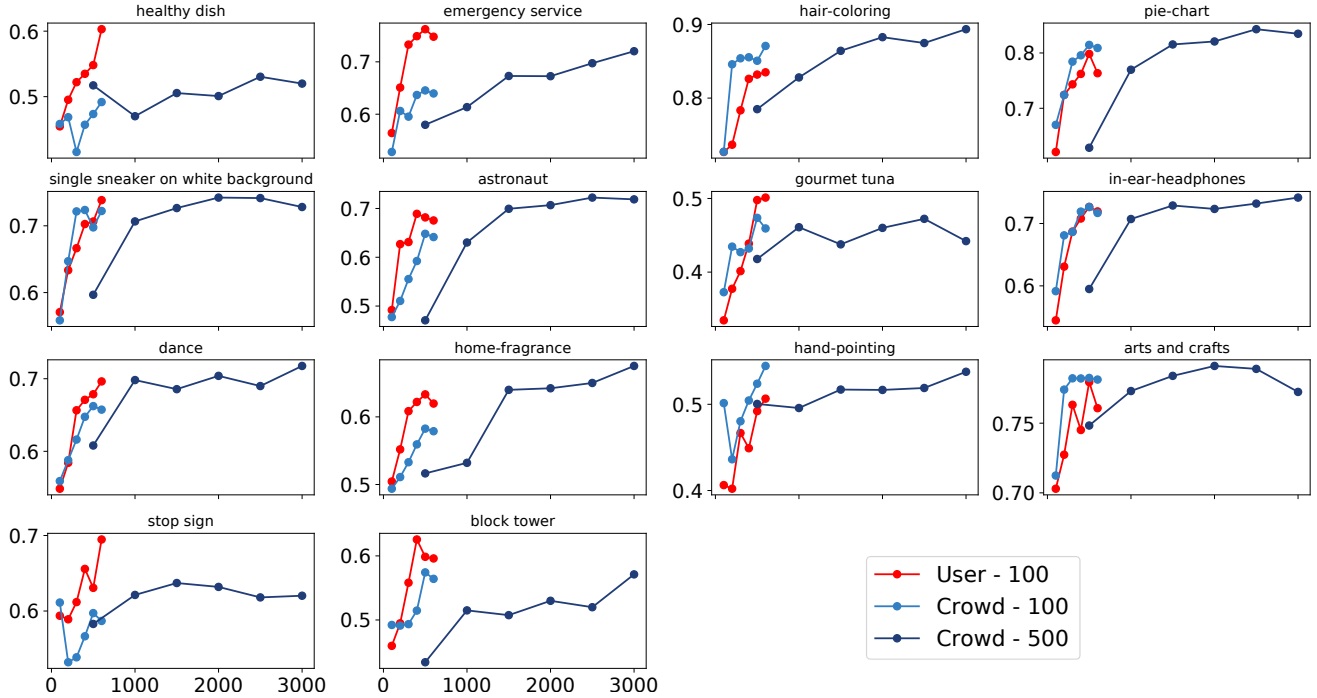


Figure 12: Results per concept comparing user model performance versus crowd. We show the AUC PR (y-axis) per number of samples rated (x-axis) for each of the three active learning experimental settings: user (batch size = 100), crowd (batch size = 100), and crowd (batch size = 500).

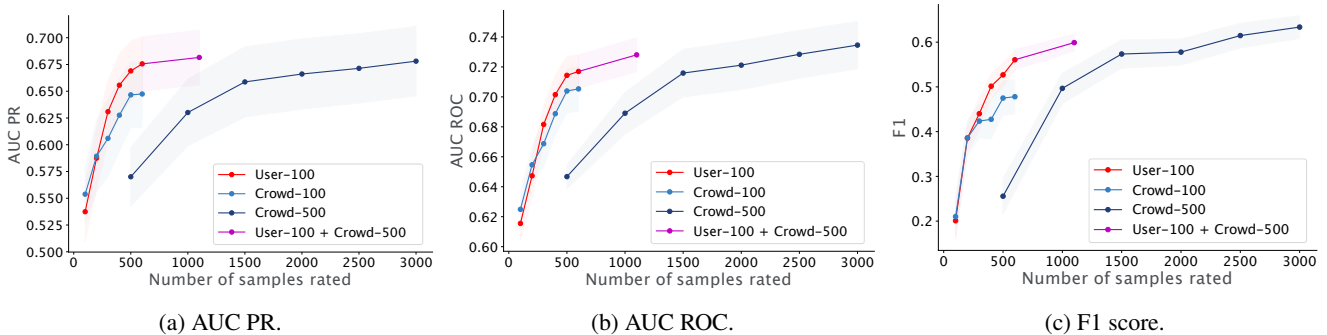


Figure 13: Model performance per amount of samples rated by the user and/or crowd raters. We also display an additional experimental setting `User-100 + Crowd-500`, where 5 rounds of user AL with batch size 100 are continued with another round of AL with crowd raters, with batch size 500. Mean and standard error over all concepts, for multiple metrics.

F. User-in-the-loop vs crowd raters

We include additional results comparing active learning with the user in the loop with active learning using crowd raters. Figure 12 shows detailed results, per concept, for the three experimental settings `User-100`, `Crowd-100` and `Crowd-500` described in Section 4.3.2. We can notice how for difficult concepts (according to the difficulty scores in Appendix D) such as `healthy dish`, the performance of the user models far exceeds that of the crowd raters, with

far less samples. On the other hand, for easy concepts such as `hair coloring` the models trained with more data from crowd raters end up superseding the best user model.

G. Augmenting user labeling with crowd-sourced ratings

One natural question to ask is what happens if we combine the benefits from doing active learning (AL) with users with those of AL with crowd raters. We considered such a

setting. For each concept, we took the model trained after 5 rounds of AL with the user (setting `User-100` in Section 4.3.2) and we used it for another round of active learning with a larger batch size (500), this time rated by crowd workers. The results are shown in Figure 13, where we named this setting `User-100 + Crowd-500`.

With additional data from the crowd raters, the model shows further improvements.

H. ImageNet21k experiment details

We use this subset of concepts in our ImageNet21k experiments:

50 easy concepts:

- | | |
|--|---|
| 1. tree frog (<i>n00442981</i>) | 25. desk (<i>n04073948</i>) |
| 2. harvestman (<i>n00453935</i>) | 26. desktop computer (<i>n04236702</i>) |
| 3. coucal (<i>n02911485</i>) | 27. gondola (<i>n04288272</i>) |
| 4. king penguin (<i>n02955540</i>) | 28. letter opener (<i>n04422875</i>) |
| 5. Irish wolfhound (<i>n02957755</i>) | 29. microwave (<i>n04571958</i>) |
| 6. komondor (<i>n02973017</i>) | 30. nail (<i>n04586581</i>) |
| 7. German shepherd (<i>n02975212</i>) | 31. patio (<i>n04970916</i>) |
| 8. bull mastiff (<i>n02982599</i>) | 32. pickup (<i>n07681926</i>) |
| 9. Newfoundland (<i>n02992032</i>) | 33. plane (<i>n07732747</i>) |
| 10. white wolf (<i>n03017168</i>) | 34. pot (<i>n07805254</i>) |
| 11. ladybug (<i>n03181293</i>) | 35. purse (<i>n07815588</i>) |
| 12. rhinoceros beetle (<i>n03340009</i>) | 36. racket (<i>n07819480</i>) |
| 13. leafhopper (<i>n03365991</i>) | 37. snowplow (<i>n07820497</i>) |
| 14. baboon (<i>n03413828</i>) | 38. sombrero (<i>n07820814</i>) |
| 15. marmoset (<i>n03439814</i>) | 39. stopwatch (<i>n07850083</i>) |
| 16. Madagascar cat (<i>n03454211</i>) | 40. strainer (<i>n07860988</i>) |
| 17. analog clock (<i>n03484083</i>) | 41. theater curtain (<i>n07867883</i>) |
| 18. apiary (<i>n03525454</i>) | 42. ice cream (<i>n07869391</i>) |
| 19. bathtub (<i>n03585875</i>) | 43. pretzel (<i>n07907161</i>) |
| 20. bookcase (<i>n03592245</i>) | 44. cauliflower (<i>n07918028</i>) |
| 21. CD player (<i>n03727837</i>) | 45. acorn squash (<i>n07933891</i>) |
| 22. chain mail (<i>n03779000</i>) | 46. lemon (<i>n08663860</i>) |
| 23. chest (<i>n03996145</i>) | 47. pizza (<i>n09213565</i>) |
| 24. cornet (<i>n04041544</i>) | 48. burrito (<i>n09305031</i>) |
| | 49. hen-of-the-woods (<i>n13908580</i>) |
| | 50. ear (<i>n14899328</i>) |

50 hard concepts:

- | | |
|--|---|
| 1. dive (<i>n00442981</i>) | 26. sleeve (<i>n04236702</i>) |
| 2. fishing (<i>n00453935</i>) | 27. spring (<i>n04288272</i>) |
| 3. buffer (<i>n02911485</i>) | 28. thermostat (<i>n04422875</i>) |
| 4. caparison (<i>n02955540</i>) | 29. weld (<i>n04571958</i>) |
| 5. capsule (<i>n02957755</i>) | 30. winder (<i>n04586581</i>) |
| 6. cartridge holder (<i>n02973017</i>) | 31. pink (<i>n04970916</i>) |
| 7. case (<i>n02975212</i>) | 32. cracker (<i>n07681926</i>) |
| 8. catch (<i>n02982599</i>) | 33. cress (<i>n07732747</i>) |
| 9. cellblock (<i>n02992032</i>) | 34. mash (<i>n07805254</i>) |
| 10. chime (<i>n03017168</i>) | 35. pepper (<i>n07815588</i>) |
| 11. detector (<i>n03181293</i>) | 36. mustard (<i>n07819480</i>) |
| 12. filter (<i>n03340009</i>) | 37. sage (<i>n07820497</i>) |
| 13. floor (<i>n03365991</i>) | 38. savory (<i>n07820814</i>) |
| 14. game (<i>n03413828</i>) | 39. curd (<i>n07850083</i>) |
| 15. glider (<i>n03439814</i>) | 40. dough (<i>n07860988</i>) |
| 16. grapnel (<i>n03454211</i>) | 41. fondue (<i>n07867883</i>) |
| 17. handcart (<i>n03484083</i>) | 42. hash (<i>n07869391</i>) |
| 18. holder (<i>n03525454</i>) | 43. Irish (<i>n07907161</i>) |
| 19. ironing (<i>n03585875</i>) | 44. sour (<i>n07918028</i>) |
| 20. jail (<i>n03592245</i>) | 45. herb tea (<i>n07933891</i>) |
| 21. mat (<i>n03727837</i>) | 46. top (<i>n08663860</i>) |
| 22. module (<i>n03779000</i>) | 47. bank (<i>n09213565</i>) |
| 23. power saw (<i>n03996145</i>) | 48. hollow (<i>n09305031</i>) |
| 24. radio (<i>n04041544</i>) | 49. roulette (<i>n13908580</i>) |
| 25. religious residence (<i>n04073948</i>) | 50. culture medium (<i>n14899328</i>) |