Please do not post this paper publicly online, as this is ongoing research. Thank you!

Task Discovery

CS230 Final Project: Milestone

Topic: Transfer Learning

Ajay Sohmshetty (collaboration with Amir Zamir)

Abstract

This project centers around the theme of task discovery. Specifically, we execute a principled study towards discovering members of the task space. To accomplish this, we first define members of the visual task space (i.e. different visual tasks, or better termed visual abstractions). We then explore a series of learnability experiments to better understand what makes a task real versus random. Finally, we investigate a series of reduction mechanisms to distill this space and make this problem more tractable, and execute a task discovery search. We find encouraging results. This research work is mentored by Amir Zamir.

Background and Related Work

Recently, transfer learning has garnered much attention. Taskonomy [1] helped to establish the inherent relationships between tasks, but these tasks have been hand chosen by humans as deemed to be relevant and important. What if there are other tasks, that we simply do not have datasets on, that could be useful to transfer learn on? This project aims to generate these tasks by defining a task and investigate a series of reduction mechanisms. The ultimate goal would be to apply these as well as other reduction mechanisms to filter down to a set of generated tasks that can be learned from other models, which can then be used for transfer learning purposes. Related work includes:

1. Taskonomy http://taskonomy.stanford.edu/taskonomy_CVPR2018.pdf [1]
2. Perceptual losses for real-time style transfer and super-resolution https://arxiv.org/abs/1603.08155 [5]
3. Random Features for Large-Scale Kernel Machines https://people.eecs.berkeley.edu/~brecht/papers/07.rah.rec.nips.pdf [6]
4. Projection Pursuit https://www.jstor.org/stable/2241175?seq=1#page_scan_tab_contents [7]
5. Multi-loss Regularized Deep Neural Network https://ieeexplore.ieee.org/document/7258343/ [8]
6. Differentiable parameterizations: https://distill.pub/2018/differentiable-parameterizations/ [9]

Task Formulation

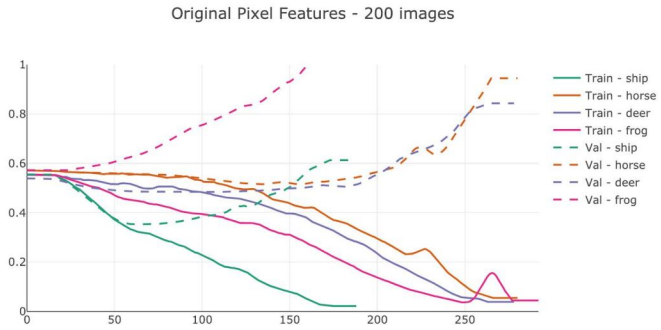
For these experiments, we use CIFAR10 and CIFAR100 datasets. Instead of the model attempting to predict the class of the image, we formulate the task into a binary classification problem. We pick a class A and generate *n-1* other random classes. For each class chosen, we pick *m* randomly chosen CIFAR100 images. The label will be a 1 if the image is of class A, and 0 otherwise. The dataset size is $N=n*m$. Note: this simplification is intentional, as it drastically simplifies the space of tasks. To further simplify the space of tasks, throughout this project we predominantly work with relatively small datasets, typically ranging from about 100-200 datapoints. We employ the standard 80-20 train-dev split.

Architecture Used

Layer	Details
input	32x32x3 (cifar100 images)
conv1	5x5 filters, 6 of them
pool	2x2 filter, stride 2
conv2	5x5 filters, 16 of them
fc1	120 hidden units
fc2	84 hidden units
fc3	2 output units

The above table shows the architecture we used. We intentionally chose to use very shallow network. Our dataset size is quite small, so using a more sophisticated architecture (as in the architecture presented in the ResNet [10] paper) would overfit immediately. The goal here is to identify learnable tasks - not to solve them.

We pick a 200 image task (for classes “ship”, “horse”, “deer”, and “frog”) and run our baseline architecture. The figure below shows the results. Note how almost all tasks generalize, though the frog task seems to be a particularly difficult one.

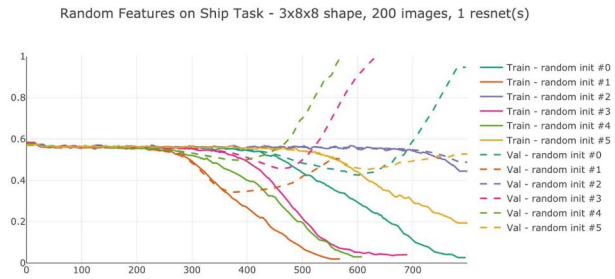


Baseline Classifiers

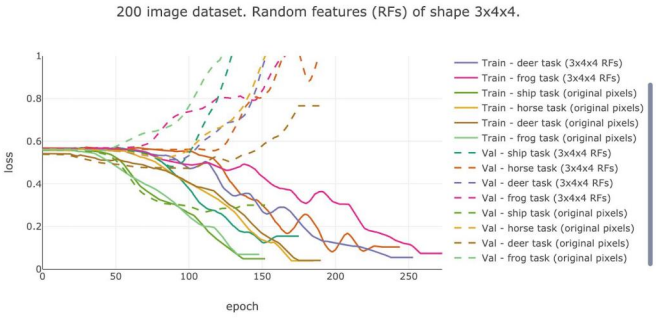
For our baselines, we use a statistically informed as well as a blind classifier.

Random Features Exploration

CIFAR100 images are already quite small at 32x32x3 dimensionality. Here, we investigated whether there a potentially more concise representation of these images. Can we use a feature extractor and train on derived features to identify learnability instead? The immediate thought was to simply use a pretrained ResNet (or similar arch) to generate these embeddings. However, these pretrained models are biased: they extract features relevant to the task that they were pretrained for. This is ultimately problematic because our end goal would be to generate “tasks” in large scale. What follows has been inspired by recent work has showcased the power of random features generated from large randomly initialized ResNets. I used the CIFAR100 ResNet architecture proposed in the original Deep Residual Networks paper as inspiration for the model to use to extract features, and randomly initialized 6 networks. I adjusted the final few layers to make the output size 3x8x8. The graph below shows the results of this experiment.



Note that this on the “ship task” which has clear signs of generalization as in the above baseline. We can see that there are random initializations that generalize. However, many of the initializations do not seem to generalize. These random initializations, while sometimes useful, are clearly volatile and require aggregation / ensembling in order to be useful. Furthermore, the original pixel task seems to performing better than all initializations. So there is no clear input dimensionality reduction benefit, but an additional computational cost (with generating these random features) and a loss in generalization. Here’s another set of experiments I ran, this time for 3x4x4 random features. I overlaid the original pixel task curves. Clearly, anything below 3x8x8 doesn’t work; none of the tasks’ random features show signs of generalization.

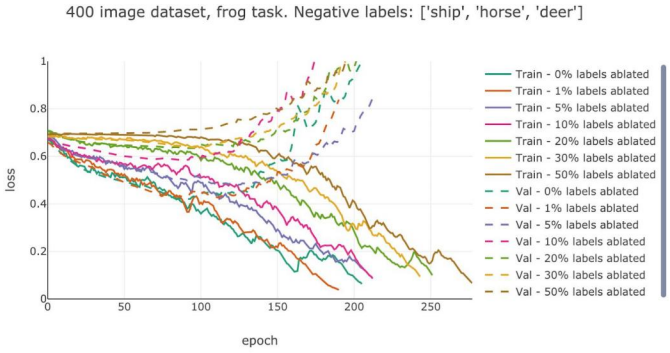


Learnability Studies

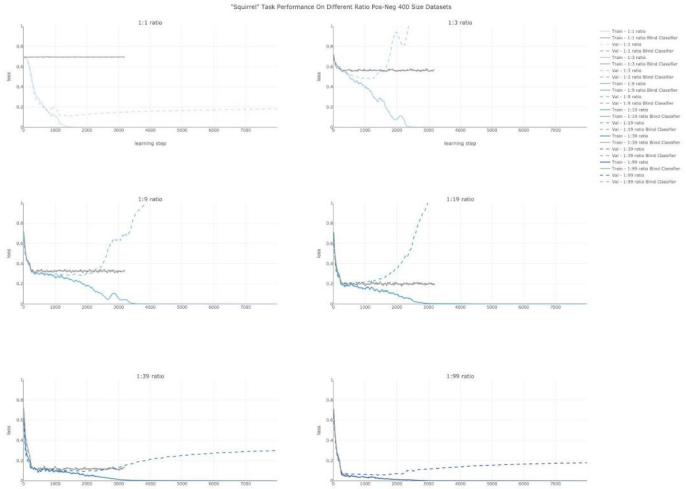
In order to better understand the concept of task learnability, we performed a series of additional studies.

Ablation study. This tests the degree to which we can ablate labels of a particular task and still be able to discover it. By identifying this tolerance threshold, we can use it to cluster similar tasks, and thus leverage this mechanism to reduce the full space of tasks to try. From the below graph, we

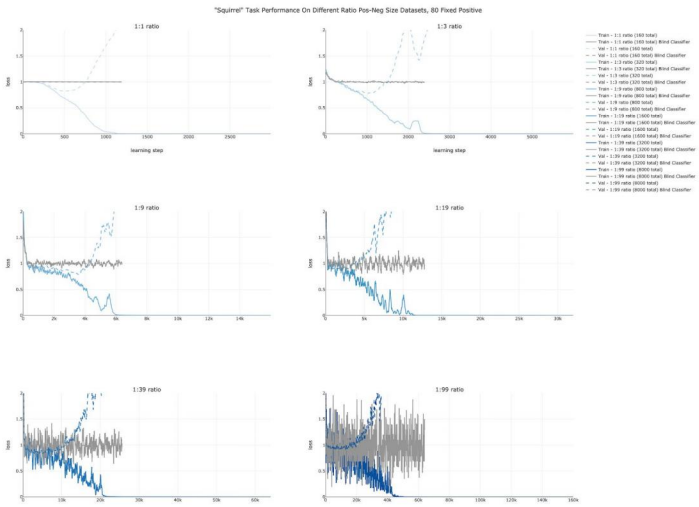
pick a 20% ablation tolerance, since we still safely see generalization in this regime.



Dataset size and class distribution study. We define class distribution as the ratio between the positive and negative classes. In our experiments, found that the dataset size is highly dependant on the class distribution and the classes chosen. Given a 400 size dataset, as in the graph below, generalization is lost beyond the 1:19 class ratio data regime. Hence, we will stick with at most a 1:19 ratio moving forward.



Fixed number of positive datapoints study. This investigates whether it is indeed the class ratio, or rather the number of positive datapoints that the model is exposed that is in fact salient. We conclude here that number of positive datapoints is the more important factor, but too high of a class ratio results in the model overfitting too quickly.



Real vs. Random Task Evaluation

Given these learnability experiments, we are now armed with a way to distinguish between a real and a random task. In order to calibrate losses, we first run our baseline models (see **Baseline** section), then take the average validation loss; let this be *b*. Given the validation loss per learning step for a given training run, we smooth the validation losses using a rolling window of size 5, taking the median for each window. The minimum value of this list gives us the generalization level of the run, let this be *g*. Therefore, we define the task-ness score to be *(b-g)/b*. Furthermore, a particular dataset is a task if $(b-g)/b > T_{task}$, where T_{task} is the task threshold. A task discovery system attempts to find a label set (i.e. an assignment of labels to a given set of images), such that it maximizes the task-ness score.

Clustering As A Reduction Mechanism

The task space is combinatorially massive. Consider a dataset of size 200. The number of label assignments for a binary classification is 2^{200}! To evaluate every single label set would be simply intractable. Therefore, we use the ablation tolerance found (20%) from previous experiment to cluster label sets. Upon initial inspection, the meanshift clustering seemed like an attractive clustering algorithm to use, since the bandwidth parameter has a direct geometric interpretation (can tie back to the ablation tolerance of 20%), and also does not require a cluster *k* parameter as in k-means. However, due to the symmetry and uniformly distributed nature of the label set space,

this approach did not work; the algorithm either found 1 single cluster, or *N* number of clusters (where *N* is the size of the dataset). Hence, we moved onto k-means, and attempted to empirically determine the *k* value required to achieve a 20% average ablation for each cluster. Unfortunately, we did not see any significant reduction in this space; likely because asymptotically this is a linear reduction, whereas the label space grows exponentially. We are still investigating this and evaluating the mathematics behind this phenomenon. The below table shows our results on this.

N, ratio	*K*	Label set size	Avg Cluster Ablation
400, 1:9	6000	20000	50%
400, 1:9	6000	30000	57%
400, 1:9	7000	30000	54%
400, 1:9	9000	30000	50%
400, 1:9	10000	40000	47%
400, 1:9	9000	40000	55%
400, 1:9	10000	40000	53%
400, 1:9	12000	40000	50%
100, 1:9	1000	20000	51%
100, 1:9	1000	30000	54%
100, 1:9	1000	40000	57%
100, 1:9	1000	50000	58.5%
100, 1:9	1000	60000	60%
100, 1:9	1000	80000	62.6%
100, 1:9	1000	100000	63.9%
100, 1:9	1000	150000	65.4%
20, 1:4	150	1000	20.1%
20, 1:4	150	2000	22.5%
20, 1:4	150	3000	23.3%
20, 1:4	150	5000	24.0%
20, 1:4	150	6000	24.3%
20, 1:4	150	7000	24.7%
20, 1:4	150	8000	25.0%
20, 1:4	150	9000	24.9%
20, 1:4	150	10000	25.1%
20, 1:4	150	13000	25.4%
20, 1:4	150	15000	25.2%
20, 1:4	150	18000	25.4%
20, 1:4	150	25000	25.8%
20, 1:4	150	30000	25.3%
20, 1:4	150	40000	25.4%
20, 1:4	150	50000	25.3%
20, 1:4	150	100000	25.7%

Note that we simply do not see average cluster ablation percentage stabilize under any regime for dataset size of 100 and 400 (ie increasing label set size also increases average cluster ablation). It only stabilizes under the 20 dataset size regime, where the total number of label sets possible is exactly 4845. As such it becomes invariant to label set size only after it passes that mark. Meanwhile, in the non-20 dataset size regime, the relationship starts off as roughly linear, then seems to decay sublinearly. We defer further analysis to a later exploration, and instead move on without performing clustering as a reduction mechanism.

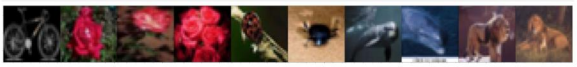
Task Discovery

As an initial experiment, we first generated 100 random tasks (500 dataset size, 1:9 ratio), and then examined the random task with the highest taskness score. The result was noise, and the labels were not anything intelligible. So, we employed gradient free optimization techniques to better search this space, using the NeverGrad

package by Facebook [3]. This experiment proved to be more fruitful, and showed encouraging results. We formulated the problem as follows. We select a set of *N* images (we use 50), of class ratio 1:*R* (we use 4). Hence there are *N/(R+1)* total true positive CIFAR classes in the image set. NeverGrad optimizes over a hyperparameter vector of size *N*, where each dimension takes a value between 0 and 1. During evaluation, we take the highest *N/(R+1)* values in the hyperparameter vector, and set the corresponding labels in our label set to 1. We then split our dataset and train the model, defining the evaluation loss of this run as 1-mean taskness score. We set a budget of 4000 and use the TwoPointsDE optimization strategy, an evolution strategy black-box optimization algorithm [2]. Here are some sample results from repeated runs of this setup:

Run 1: Taskness score 1-0.21/0.45 = 0.53

Run 2: Taskness score 1-0.29/0.45 = 0.35:



Run 3: Taskness score 1-0.33/0.45 = 0.26:

Run 4: Taskness score 1-0.17/0.45 = 0.62:

Here, these tasks are clearly related! For example in run 1, we see clear common patterns in the images (ex. triangles present, blue colors). Similar types of patterns exist in the other 4 runs as well. Compared with the purely random search, this seems to be performing much better, and we are discovering what seem like real tasks!

Conclusion

We define a visual perception task, and using CIFAR100 images, perform a series of learnability experiments. From these explorations, we find an automated method of computing taskness score, and run gradient-free blackbox optimization on this space to discover tasks. We see encouraging results, as discovered task look meaningful and return real task-like taskness scores. The next logical step would be to perform a large scale task discovery search. This involves enforcing orthogonality among found tasks. I.e. we want to find sufficiently different tasks in each subsequent search. We hope to discover the original CIFAR100 tasks in this dataset, alongside other real, interesting tasks. If successful, we can leverage these tasks for transfer learning purposes. We've seen from Taskonomy [1] that establishing and leveraging relationships between tasks can build better, more generalizable models. An automated method of discovering tasks such as the method presented in this paper may unlock the true potential of transfer learning. We hope to submit to NeurIPS this upcoming May.

Contributions

I would like to thank Amir Zamir for his insight and direction, as well as Shervine Amidi for his mentorship and availability.

Github Link

https://github.com/ajay1495/task_discovery

References

1. http://taskonomy.stanford.edu/taskonomy_CVPR2018.pdf
2. http://www.cs.cmu.edu/~aarti/SMLRG/miguel_slides.pdf
3. https://code.fb.com/ai-research/nevergrad/
4. https://arxiv.org/pdf/1703.03864.pdf
5. https://arxiv.org/abs/1603.08155
6. https://people.eecs.berkeley.edu/~brecht/papers/07.rah.rec.nips.pdf
7. https://www.jstor.org/stable/2241175?seq=1#page_scan_tab_contents
8. https://ieeexplore.ieee.org/document/7258343/
9. https://distill.pub/2018/differentiable-parameterizations/
10. https://arxiv.org/abs/1512.03385