SUPPLEMENTARY MATERIALS FOR AN INFORMATION-THEORETIC APPROACH TO TRANSFERABILITY IN TASK TRANSFER LEARNING

Yajie Bao[1⋆] *Yang Li*[1⋆] *Shao-Lun Huang*[1] *Lin Zhang*[1] *Lizhong Zheng*[2] *Amir Zamir*[3,4] *Leonidas Guibas*[3]

[1] Tsinghua-Berkeley Shenzhen Institute [2] Massachusetts Institute of Technology
[3] Stanford University [4] University of California, Berkeley

1. DERIVATION OF EQUATION (5)

First, we need to introduce some additional notations. Let X, x, $\mathcal{X}$ and P_X represent a random variable, a value, the alphabet and the probability distribution respectively. $\sqrt{\mathrm{P}_X}$ denotes the vector with entries $\sqrt{P_X(x)}$ and $[\sqrt{\mathrm{P}_X}] \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{X}|}$ denotes the diagonal matrix of $\sqrt{\mathrm{P}_X}$. For joint distribution P_{YX}, $\mathrm{P}_{YX} \in \mathbb{R}^{|\mathcal{Y}|\times|\mathcal{X}|}$ represents the probability matrix. Given k feature functions $f_i : \mathcal{X} \to \mathbb{R}, i = 1, ..., k$, let $f(x) = [f_1(x), ..., f_k(x)] \in \mathbb{R}^k$ be the feature vector of x, and $F = [f(x_1)^{\mathrm{T}}, ..., f(x_{|\mathcal{X}|})^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{|\mathcal{X}|\times k}$ be the feature matrix over all elements in $\mathcal{X}$.

The left hand side of Equation (5) can be expressed as

$$\begin{aligned}||\tilde{B}\Phi(\Phi^{\mathrm{T}}\Phi)^{-\frac{1}{2}}||_F^2 =& \mathrm{tr}\left((\Phi^{\mathrm{T}}\Phi)^{-\frac{1}{2}}\Phi^{\mathrm{T}}\tilde{B}^{\mathrm{T}}\tilde{B}\Phi(\Phi^{\mathrm{T}}\Phi)^{-\frac{1}{2}}\right)\\ =& \mathrm{tr}\left((\Phi^{\mathrm{T}}\Phi)^{-1}\Phi^{\mathrm{T}}\tilde{B}^{\mathrm{T}}\tilde{B}\Phi\right) \qquad \text{(S1)}\end{aligned}$$

Since any feature function can be centered by subtracting the mean, without the loss of generality, we assume $\mathbb{E}[f(X)] = 0$. Using the one-to-one correspondence between Φ and F, i.e. $\Phi = \left[\sqrt{\mathrm{P}_X}\right] F \in \mathbb{R}^{|\mathcal{X}|\times k}$, we have

$$\begin{aligned}\Phi^{\mathrm{T}}\Phi &= \left(\left[\sqrt{\mathrm{P}_X}\right] F\right)^{\mathrm{T}}\left(\left[\sqrt{\mathrm{P}_X}\right] F\right)\\ &= \mathbb{E}[f(X)^{\mathrm{T}} f(X)]\\ &= \mathrm{cov}(f(X)) \qquad \text{(S2)}\end{aligned}$$

The DTM matrix $\tilde{B}$ introduced in Definition 1 can be written in matrix notation: $\tilde{B} = \left[\sqrt{\mathrm{P}_Y}\right]^{-1} \mathrm{P}_{YX} \left[\sqrt{\mathrm{P}_X}\right]^{-1} - \sqrt{\mathrm{P}_Y}\sqrt{\mathrm{P}_X}^{\mathrm{T}}$. Then we have,

$$\begin{aligned}\tilde{B}\Phi &= \left(\left[\sqrt{\mathrm{P}_Y}\right]^{-1} \mathrm{P}_{YX} \left[\sqrt{\mathrm{P}_X}\right]^{-1} - \sqrt{\mathrm{P}_Y}\sqrt{\mathrm{P}_X}^{\mathrm{T}}\right)\cdot\\ &\quad \left[\sqrt{\mathrm{P}_X}\right] F\\ &= \left[\sqrt{\mathrm{P}_Y}\right]\left([\mathrm{P}_Y]^{-1}\mathrm{P}_{YX}F - \mathbf{1}\cdot\mathbb{E}[f(X)]^{\mathrm{T}}\right),\end{aligned}$$

where $\mathbf{1}$ is a column vector with all entries 1 and length $|\mathcal{Y}|$. It follows that,

$$\begin{aligned}\Phi^{\mathrm{T}}\tilde{B}^{\mathrm{T}}\tilde{B}\Phi =& \left([\mathrm{P}_Y]^{-1}\mathrm{P}_{YX}F - \mathbf{1}\cdot\mathbb{E}[f(X)]^{\mathrm{T}}\right)^{\mathrm{T}}\cdot\\ & [\mathrm{P}_Y]\left([\mathrm{P}_Y]^{-1}\mathrm{P}_{YX}F - \mathbf{1}\cdot\mathbb{E}[f(X)]^{\mathrm{T}}\right)\\ =& \mathbb{E}_{P_Y}\left[(\mathbb{E}[f(X)|Y] - \mathbf{1}\cdot\mathbb{E}[f(X)]^{\mathrm{T}})^{\mathrm{T}}\cdot\right.\\ & \left.(\mathbb{E}[f(X)|Y] - \mathbf{1}\cdot\mathbb{E}[f(X)]^{\mathrm{T}})\right]\\ =& \mathrm{cov}\left(\mathbb{E}[f(X)|Y]\right) \qquad \text{(S3)}\end{aligned}$$

By substituting (S2) and (S3) into (S1), we have

$$||\tilde{B}\Phi(\Phi^{\mathrm{T}}\Phi)^{-\frac{1}{2}}||_F^2 = \mathrm{tr}\left(\mathrm{cov}(f(X))^{-1}\mathrm{cov}(\mathbb{E}[f(X)|Y])\right)$$

2. OPERATIONAL MEANING OF H-SCORE

In this section, we will show that H-score characterizes the asymptotic probability of error in the hypothesis testing context. We will start with some background on error exponents from statistics, then explain how to estimate it using information geometry. Finally, we will show how computing H-score is in fact estimating the error exponent of a feature function on the sample data.

2.1. Error Exponent and Hypothesis Testing

Consider the binary hypothesis testing problem over m i.i.d. sampled observations $\{x^{(i)}\}_{i=1}^m \triangleq x^m$ with the following hypotheses: $H_0 : x^m \sim P_1$ or $H_1 : x^m \sim P_2$.

Let P_{x^m} be the empirical distribution of the samples. The optimal test, i.e., the log likelihood ratio test $\log(T) = \log\frac{P_1(x^m)}{P_2(x^m)}$ can be stated in terms of information-theoretic quantities as follows:

$$m[D(P_{x^m}||P_2) - D(P_{x^m}||P_1)] \underset{H_1}{\overset{H_0}{\gtrless}} \log T$$

where D is the Kullback-Leibler (KL) divergence operator.

Further, using Sannov's theorem, we have the asymptotic probability of type I error:

$$\alpha = P_1(A^c) \approx 2^{-mD(P_1^*||P_1)}$$

⋆ Joint-first authors

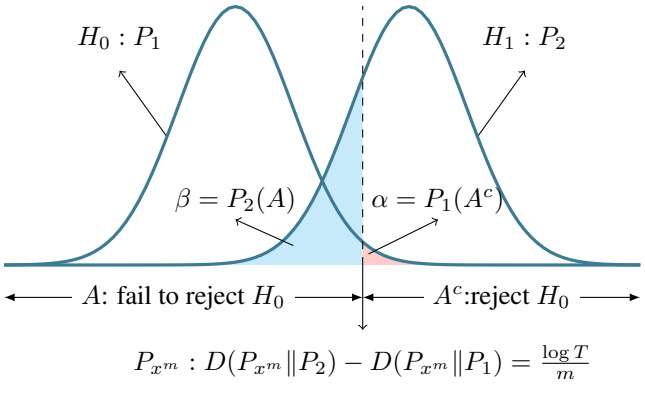

Fig. S1: The binary hypothesis testing problem. The blue curves shows the probility density functions for P_1 and P_2. The rejection region A^c and the acceptance region A are highlighted in red and blue, respectively. The vertical line indicates the decision threshold.

where $P_1^* = \text{argmin}_{P\in A^c} D(P||P_1)$ and $A^c(T) = \{x^m : D(P_{x^m}||P_2) - D(P_{x^m}||P_1) < \frac{1}{m}\log T\}$ represents the rejection region. Similarly, the asymptotic probability of type II error is

$$\beta = P_2(A) \approx 2^{-mD(P_2^*||P_2)},$$

where $P_2^* = \text{argmin}_{P\in A} D(P||P_2)$ and $A = \{x^m : D(P_{x^m}||P_2) - D(P_{x^m}||P_1) > \frac{1}{m}\log T\}$ represents the acceptance region (See Figure S1). Using the Bayesian approach for hypothesis testing, the overall error probability of the log likelihood ratio test is defined as:

$$P_e^{(m)} = \alpha P_1 + \beta P_2$$

and the *best achievable exponent in the Bayesian probability of error* (a.k.a. *error exponent*) is defined as:

$$E = \lim_{m\to\infty} \min_{A\subseteq\mathcal{X}^m} -\frac{1}{m}\log P_e^{(m)}$$

Error exponent E expresses the best rate at which the error probability decays as sample size increases for a particular hypothesis testing problem. See [1] for more background information on error exponents and its related theorems.

2.2. Estimating Error Exponents

Suppose P_1 and P_2 are sampled from the ϵ-neighborhood $\mathcal{N}_\epsilon(P_0) \triangleq \{P|\sum_{x\in\mathcal{X}} \frac{(P(x)-P_0(x))^2}{P_0(x)} \le \epsilon^2\}$ centered at a reference distributon P_0, and let $\phi_1, \phi_2 \in \mathbb{R}^{|\mathcal{X}|}$ be vectors defined as:

$$\phi_i(x) \triangleq \frac{P_i(x) - P_0(x)}{\epsilon\sqrt{P_0(x)}}$$

for $i = 1, 2$. The following lemma express the optimal error exponent E using ϕ_1 and ϕ_2:

Lemma 1. *Under the local assumption defined earlier, the best achievable error exponent of the binary hypothesis testing problem with probabilities P_1 and P_2 is:*

$$E = \frac{\epsilon^2}{8}\|\phi_1 - \phi_2\|^2 + o(\epsilon^2)$$

where ϵ is a constant [2].

While the above lemma characterizes the asymptotic error probability distinguishing P_1 and P_2 based on the optimal decision function, most decision functions we learn from data are not optimal, as P_1 and P_2 are unknown. Given sample data x^m and an arbitrary feature function $f : \mathcal{X} \to \mathbb{R}$, which could be learned from a pre-trained model, the error exponent of the decision function based on f is reduced in a way defined by the following Lemma:

Lemma 2. *Given a zero-mean, unit variance feature function $f : \mathcal{X} \to \mathbb{R}$ and i.i.d. sampled data x^m, the error probability of a mismatched decision function of the form $l = \frac{1}{m}\sum_{i=1}^m (f(x^{(i)}))$ has an exponent*

$$E_f = \frac{\epsilon^2}{8}\langle\xi, \phi_1 - \phi_2\rangle^2 + o(\epsilon^2)$$

where $\xi \in \mathbb{R}^{|\mathcal{X}|}$ is a vector with entries $\xi(x) = \sqrt{P_0(x)}f(x)$ [2].

This lemma characterizes the error probability of using a normalized feature of the input data to solve a learning task by a linear projection of this feature between the input and output domains. Note that normalizing features to zero-mean and unit variance results in an equivalent decision function with a different threshold value, thus we can apply Lemma 2 to any features without the loss of generality. Further, it's obvious that the reduced exponent E_f is maximized when $\xi = \phi_1 - \phi_2$, and the optimal value is exactly the optimal error exponent E in Lemma 1. To estimate the reduced error exponent for multi-dimensional features, we present the k-dimensional generalization of Lemma 2 below:

Lemma 3. *Given k normalized feature functions $f(x) = [f_1(x), \ldots, f_k(x)]$, such that $\mathbb{E}[f_i(X)] = 0$ for all i, and $\text{cov}(f(X)) = I$, we define a k-d statistics of the form $l^k = (l_1, ..., l_k)$ where $l_i = \frac{1}{m}\sum_{l=1}^m f_i(x^{(l)})$. Let $\xi_1, \ldots, \xi_k$ be k vectors with entries $\xi_i(x) = \sqrt{P_X(x)}f_i(x)$, $0 \le i \le k$. Then the error exponent of l^k is*

$$E_f^k = \sum_{i=1}^k E_{f_i} = \sum_{i=l}^k \frac{\epsilon^2}{8}\langle\xi_i, \phi_1 - \phi_2\rangle^2 + o(\epsilon^2) \qquad \text{(S4)}$$

The proof of this Lemma can be found in [3].

2.3. H-score and Error Exponents

Now we return to the binary classification problem. Using Lemma 3, we will show the linear relationship between H-score and error exponents.

Theorem 1. *Given $P_{X|Y=0}, P_{X|Y=1} \in \mathcal{N}_\epsilon^{\mathcal{X}}(P_{0,X})$ and features f such that $\mathbb{E}[f(X)] = 0$ and $\mathbb{E}[f(X)f(X)^{\mathrm{T}}] = I$, there exists some constant c independent of f such that $E_f^k = c\mathcal{H}(f)$.*

Proof. By Lemma 3, the L.H.S. of the equation can be written as $E_f^k = c_0 \sum_{i=l}^{k} \langle \xi_i, \phi_1 - \phi_2 \rangle^2$ for some constant c_0. It follows that

$$
\begin{aligned}
& c_0 \sum_{i=l}^{k} \langle \xi_i, \phi_1 - \phi_2 \rangle^2 \\
=& c_0 \left(\left(\mathrm{P}_{X|Y=0} - \mathrm{P}_{X|Y=1} \right)^{\mathrm{T}} F \right) \left(\left(\mathrm{P}_{X|Y=0} - \mathrm{P}_{X|Y=1} \right)^{\mathrm{T}} F \right)^{\mathrm{T}} \\
=& c_0 \left(\mathbb{E}[f(X)|Y=0] - \mathbb{E}[f(X)|Y=1] \right)^{\mathrm{T}} \cdot \\
& \left(\mathbb{E}[f(X)|Y=0] - \mathbb{E}[f(X)|Y=1] \right) \\
=& c_0 \frac{P_Y(0) + P_Y(1)}{P_Y(0) P_Y(1)} \left(\frac{P_Y(0) P_Y(1) + P_Y(1)^2}{P_Y(0)} \right) \cdot \\
& \mathbb{E}\left[f(X)|Y=1 \right]^{\mathrm{T}} \mathbb{E}\left[f(X)|Y=1 \right] \\
=& c \, \mathrm{tr}\left(\mathrm{cov}(\mathbb{E}[f(X)|Y]) \right) \\
=& c \, \mathcal{H}(f)
\end{aligned}
$$

The last equation uses the fact $\mathrm{cov}(f(X)) = I$. □

3. EXPERIMENT DETAILS

3.1. Experiment 4.1

Experiment Setup. The training data for the target task in this experiment consists of $20,000$ images randomly sampled from the Cifar-100 dataset [4]. It is further split 9:1 into a training set and a testing set.

We first extracted features of the Cifar-100 training images from five different layers (4a - 4f) of the ResNet-50 model pretrained on ImageNet-1000 [5]. Then we computed the H-score and the empirical transfer performance of each feature function for the Cifar-100 task. To compute the empirical performance, we trained the transfer network using stochastic gradient descent with batch size $20,000$ for 100 epochs.

Result Discussion. As shown in Fig. 2.a of the main paper, transfer performance is better when an upper layer of the source networks is transferred. This could be due to the inherent similarity between the target task and the source task, such that the optimal representation learned for one task can still be suitable for the other. For the experiment of selecting the best target task (Fig. 2.b), we used the same network as the former experiment to compute the empirical transfer performance with batch size 64 for 50 epochs.

In addition, we validated H-score under different target sample sizes between 5-50K. Fig. S2 shows that target sample size does not affect the relationship between H-score and log-loss, which further demonstrates that the H-score computation is sample efficient.

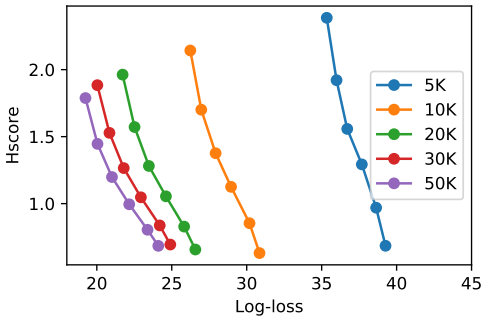

Fig. S2: H-score and transferability vs. the empirical transfer performance measured by log-loss for different target sample size (5K-50K).

3.2. Experiment 4.2

Data and Tasks. The Taskonomy dataset [6] contains 4,000,000 images of indoor scenes collected from 600 buildings. Every image has annotations for 26 computer vision tasks. For the transferability experiment, we randomly sampled $20,000$ images as the target task training data, and selected eight supervised tasks, shown in Table S1.

Tasks	Description	Output	Quantize-level
Edge2D	2D edges detection	images	16
Edge3D	3D occlusion edges detection	images	16
Keypoint2D	2D keypoint detection	images	16
Keypoint3D	3D keypoint detection	images	16
Reshading	Image reshading	images	16
Depth	Depth estimation	images	16
Object Class.	Object classification	labels	none
Scene Class.	Scene classification	labels	none

Table S1: Task descriptions

Feature Extraction and Data Preprocessing. For each task, [6] trained a fully supervised network with an encoder-decoder structure. When testing the transfer performance from source task $\mathcal{T}_S$ to target task $\mathcal{T}_T$, the encoder output of $\mathcal{T}_S$ is used for training the decoder of $\mathcal{T}_T$. For a fair comparison, we used the same trained encoders to extract source features. The output dimension of all encoders are $16 \times 16 \times 8$ and we flattened the output into a vector of length 2048. To reduce the computational complexity, we also resized the ground truth images into 64×64.

Label Quantization. Fig. S3 illustrates the resizing and quantization process of a pixel-to-pixel task label. During the quantization process, we are primarily concerned with two factors: computational complexity and information loss. Too much information loss will lead to bad approximation

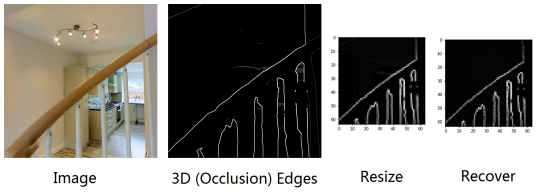

Fig. S3: Quantization. Recover is done with the centroid of corresponding cluster of each pixel.

of the original problem. On the other hand, having little information loss requires larger label space (cluster size) and higher computation cost. To test the sensitivity of the cluster size, we use cluster centroids to recover the ground truth image pixel-by-pixel. The recovery results for 3D occlusion edge detection is shown in Figure S4. When the cluster size is $N = 5$ (right), most detected edges in the ground truth image (left) are lost. We found that $N = 16$ strikes a good balance between recoverability and computation cost.

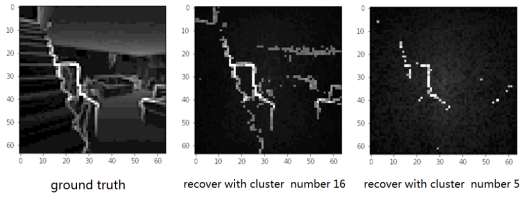

Fig. S4: Effect of quantization cluster size for 3D occlusion Edge detection.

Comparison of H-scores and Affinities. Table S2 presents the numerical values of the transferability and affinity scores between every pair of tasks, with columns representing source tasks and rows representing target tasks. This table is in direct correspondence with the ranking matrices in Fig. 3 of the main paper. For each target task, the upper row shows our results while the lower one shows the results in [6]. Score values are included in parenteses.

Here we present some detailed results on the comparison between H-score and the affinity score in [6] for pairwise transfer. The results of transferring from all tasks to the two classification tasks (Object Class. and Scene Class.) are shown in Figure S5; The results of transferring to Depth is shown in S6. We can see in general, although affinity and transferability have totally different value ranges, they tend to agree on the order of the top few ranked tasks.

Computing Efficiency. We ran the experiment on a workstation with 3.40 GHz $\times 8$ CPU and 16 GB memory. Each pairwise H-score computation finished in less than one hour including preprocessing.

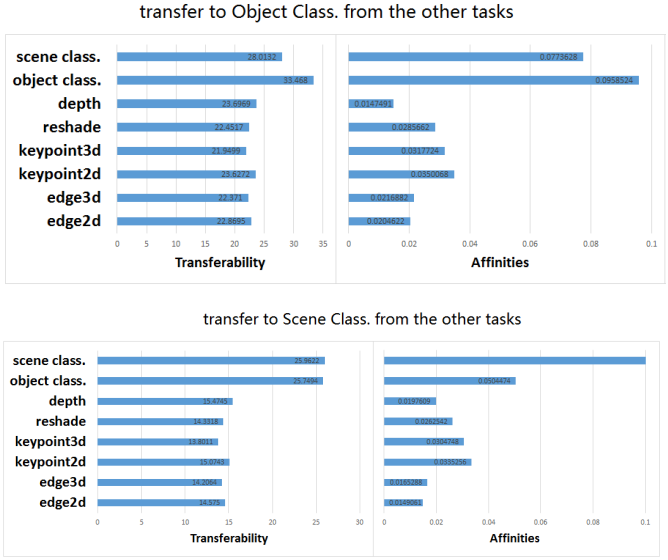

Fig. S5: Source task transferability ranking for classification tasks. For each target task, the left figure shows H-score results, and the right figure shows task affinity results.

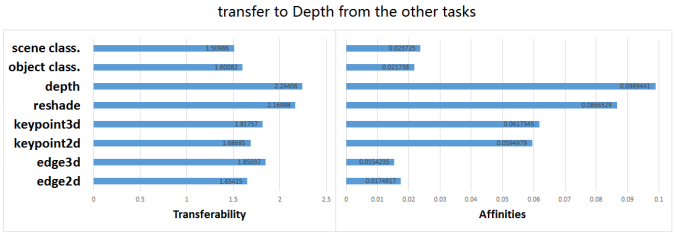

Fig. S6: Comparison between source task rankings for Depth. with H-score results on the left and affinity scores [6] on the right. The Top 3 transferable source tasks in both methods are the same: Depth, Image Reshading and 3D Occlusion Edges.

Table S2: Transferability ranking comparison, between H-score's estimation and task affinity

Tasks	2D Edges	2D Keypoints	3D Edges	3D Keypoints	Reshading	Depth	Object Class.	Scene Class.
2D Edges	1 (1.8216)	2 (1.7334)	5 (1.5704)	6 (1.5696)	4 (1.6146)	3 (1.6201)	7 (1.5097)	8 (1.4402)
	1 (0.0389)	2 (0.0117)	4 (5.8920e-5)	3 (8.8011e-5)	7 (2.9001e-5)	8 (2.2110e-5)	5 (4.9141e-5)	6 (4.8720e-5)
2D Keypoints	2 (1.6698)	1 (1.7859)	7 (1.5248)	5 (1.5287)	4 (1.5481)	3 (1.5632)	6 (1.5253)	8 (1.4725)
	2 (0.0002)	1 (0.0542)	7 (7.7797e-5)	5 (8.1029e-5)	6 (7.8464e-5)	8 (7.2724e-5)	3 (0.0002)	4 (0.0001)
3D Edges	5 (1.4828)	4 (1.4910)	3 (1.5167)	7 (1.4701)	2 (1.5405)	1 (1.6739)	8 (1.4644)	6 (1.4730)
	6 (0.0117)	7 (0.0108)	1 (0.1179)	2 (0.0734)	4 (0.0622)	3 (0.0636)	8 (0.0094)	5 (0.0151)
3D Keypoints	6 (1.5375)	5 (1.5466)	4 (1.5910)	3 (1.6456)	1 (1.7198)	2 (1.7122)	7 (1.4709)	8 (1.4121)
	5 (0.0141)	6 (0.0136)	2 (0.0531)	1(0.1275)	3 (0.0400)	4 (0.0247)	7 (0.0132)	8 (0.0121)
Reshading	5 (1.5504)	6 (1.5426)	3 (1.8174)	4 (1.7990)	1 (2.2339)	2 (2.1200)	7 (1.4774)	8 (1.3804)
	6 (0.0147)	8 (0.0143)	2 (0.0781)	4(0.0545)	1 (0.1121)	3 (0.0765)	7 (0.0144)	5 (0.0174)
Depth	6 (1.6542)	5 (1.6870)	3 (1.8504)	4 (1.8176)	2 (2.1700)	1 (2.2441)	7 (1.6008)	8 (1.5099)
	7 (0.0175)	8 (0.0154)	3 (0.0595)	4 (0.0617)	2 (0.0867)	1 (0.0989)	6 (0.0217)	5 (0.0237)
Object Class.	5 (22.866)	4 (23.627)	7 (22.371)	8 (21.950)	6 (22.452)	3 (23.697)	1 (33.468)	2 (28.013)
	7 (0.0205)	6 (0.0217)	3 (0.0350)	4 (0.0318)	5 (0.0286)	8 (0.0147)	1 (0.0959)	2 (0.0774)
Scene Class.	5 (14.575)	4 (15.074)	7 (14.206)	8 (13.801)	6 (14.332)	3 (15.474)	2 (25.750)	1 (25.962)
	8 (0.0149)	7 (0.0165)	3 (0.0335)	4 (0.0305)	5 (0.0263)	6 (0.0198)	2 (0.0504)	1 (0.1474)

3.3. Experiment 4.3

Data and Tasks. The NUS-WIDE dataset [7] contains 161,789 web images for training and 107,859 images for evaluation. Its tag set consists of 81 concepts, among which we selected two subsets, shown in Table S3. One subset contains 36 common concepts including scenes, animals and objects; The other subset contains only animal concepts. The NUS-WIDE dataset provides six types of low-level features. In this experiment, we used the 500 dimensional bag-of-words feature based on the SIFT descriptors.

Common concepts	beach, birds, boats, bridge, buildings, cars, cat, clouds, dancing, dog, fish, flowers, garden, grass, house, lake, leaf, moon, mountain, ocean, person, plants, rainbow, rocks, running, sand, sky, sports, street, sun, swimmers, town, tree, vehicle, water, window
Animal concepts	animal, bear, cat, cow, dog, elk, fox, horses, tiger, zebra

Table S3: Subsets of NUS-WIDE tag concepts used in Experiment 4.3

Feature Extraction and Data Preprocessing. We consider the prediction of each concept as an unbalanced binary classification task and train a 4-layer fully-connected neural network (Fig. S7) with batch-size 2048 for 50 epochs. For an arbitrary target task, the layer 3 activation output of the source models are used to calculate the H-scores.

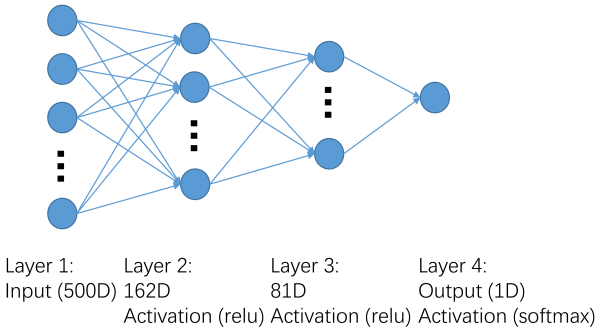

Fig. S7: Network structure for NUS-WIDE experiment.

Task Transfer Curriculum. Given n tasks $\mathcal{T}_1, \ldots, \mathcal{T}_n$, first we compute the pairwise transferability matrix $M \in \mathbb{R}^{n \times n}$ using H-score, where $M(i,j) = \frac{\mathcal{H}_{\mathcal{T}_j}(f_{\mathcal{T}_i})}{\mathcal{H}_{\mathcal{T}_j}(f_{\mathcal{T}_j})}$ for all $1 \leq i, j \leq n$. We assume that the task-specific features are close to optimal, such that $\mathcal{H}_{\mathcal{T}_j}(f_{\mathcal{T}_j}) \approx \mathcal{H}_{\mathcal{T}_j}(f_{\mathcal{T}_j opt})$. Then by Definition 3, $M(i,j) \approx \mathfrak{T}(\mathcal{T}_i, \mathcal{T}_j)$. Using the transferability matrix, we define an undirected graph G over the tasks, where the edge weight between node i and node j is defined by

$$W(i,j) = \begin{cases} 1 - \max\{M(i,j), M(j,i)\} & \text{if } \max\{M(i,j), M(j,i)\} \geq \alpha \\ 0 & \text{otherwise} \end{cases}$$

Parameter α defines the threshold to filter out less related task pairs, as the transferred representation in these cases contribute very little to the training of the target task. If G is connected, the minimum spanning tree outputs a set of task pairs that

maximizes the total transferability with $n-1$ pairwise transfers. If G is not connected, we have a minimum spanning forest that represents several task groups that can be learned independently. Finally, we recover the transfer directions on the minimum spanning tree from the transferability matrix.

For the subset with 36 common concepts in the NUS-WIDE experiment, we found that when $\alpha = 0$, i.e. no edges are filtered, 33 of the 35 edges in the resulting tree indicate transfers to concept 'sky' from other concepts. This phenomenon is reasonable in that sky and other concepts coexist in images with a high probability. To demonstrate the most significant task relationships, we set edge threshold α to be the 2.3 percentile of all weights, resulting in the minimum spanning tree in Fig. 7 of the main paper. On the other hand, edge filtering is not needed for the Taskonomy tasks and the animal concepts in the NUS-WIDE experiment, since most tasks are transferable to a similar extent. Therefore we chose $\alpha = 0$ in these cases.

4. RELATED WORKS

Transfer learning. Transfer learning can be devided into two categories: *domain adaptation*, where knowledge transfer is achieved by making representations learned from one input domain work on a different input domain, e.g. adapt models for RGB images to infrared images [8]; and *task transfer learning*, where knowledge is transferred between different tasks on the same input domain [9]. Our paper focus on the latter prolem.

Empirical studies on transferability. [10] compared the transfer accuracy of features from different layers in a neural network between image classification tasks. A similar study was performed for NLP tasks by [11]. [6] determined the optimal transfer hierarchy over a collection of perceptual indoor scene understanidng tasks, while transferability was measured by a non-parameteric score called "task affinity" derived from neural network transfer losses coupled with an ordinal normalization scheme.

Task relatedness. One approach to define task relatedness is based on task generation. Generalization bounds have been derived for multi-task learning [12], learning-to-learn [13] and life-long learning [14]. Although these studies show theoretical results on transferability, it is hard to infer from data whether the assumptions are satisfied. Another approach is estimating task relatedness from data, either explicitly [15, 16] or implicitly as a regularization term on the network weights [17, 18]. Most works in this category are limited to shallow ones in terms of the model parameters.

Representation learning and evaluation. Selecting optimal features for a given task is traditionally performed via feature subset selection or feature weight learning. Subset selection chooses features with maximal relevance and minimal redundancy according to information theoretic or statistical criteria [19, 20]. The feature weight approach learns the task while regularizing feature weights with sparsity constraints, which is common in multi-task learning [21, 22]. In a different perspective, [23] consider the universal feature selection problem, which finds the most informative features from data when the exact inference problem is unknown. When the target task is given, the universal feature is equivalent to the minimum error probability feature used in this work.

5. REFERENCES

[1] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley-Interscience, New York, NY, USA, 1991.

[2] Anuran Makur, Fabián Kozynski, Shao-Lun Huang, and Lizhong Zheng, "An efficient algorithm for information decomposition and extraction," in *Communication, Control, and Computing (Allerton), 2015 53rd Annual Allerton Conference on*. IEEE, 2015, pp. 972–979.

[3] Shao-Lun Huang, Anuran Makur, Gregory W. Wornell, and Lizhong Zheng, "On universal features for high-dimensional learning and inference," `http://allegro.mit.edu/~gww/unifeatures`, 2019.

[4] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," 2009.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[6] Amir Zamir, Alexander Sax, William Shen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese, "Taskonomy: Disentangling task transfer learning," *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[7] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yan-Tao Zheng, "Nus-wide: A real-world web image database from national university of singapore," in *Proc. of ACM Conf. on Image and Video Retrieval (CIVR'09)*, Santorini, Greece., July 8-10, 2009.

[8] Mei Wang and Weihong Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, 2018.

[9] Lisa Torrey and Jude Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pp. 242–264. IGI Global, 2010.

[10] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, "How transferable are features in deep neural networks?," in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[11] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes, "Supervised learning of universal sentence representations from natural language inference data," *arXiv preprint arXiv:1705.02364*, 2017.

[12] Jonathan Baxter, "A model of inductive bias learning," *Journal of artificial intelligence research*, vol. 12, pp. 149–198, 2000.

[13] Andreas Maurer, "Transfer bounds for linear feature learning," *Machine learning*, vol. 75, no. 3, pp. 327–350, 2009.

[14] Anastasia Pentina and Christoph H. Lampert, "A pac-bayesian bound for lifelong learning," in *International Conference on International Conference on Machine Learning*, 2014, pp. II–991.

[15] Edwin V Bonilla, Kian M Chai, and Christopher Williams, "Multi-task gaussian process prediction," in *Advances in neural information processing systems*, 2008, pp. 153–160.

[16] Yu Zhang, "Heterogeneous-neighborhood-based multi-task local learning algorithms," in *Advances in neural information processing systems*, 2013, pp. 1896–1904.

[17] Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram, "Multi-task learning for classification with dirichlet process priors," *Journal of Machine Learning Research*, vol. 8, no. Jan, pp. 35–63, 2007.

[18] Laurent Jacob, Jean-philippe Vert, and Francis R Bach, "Clustered multi-task learning: A convex formulation," in *Advances in neural information processing systems*, 2009, pp. 745–752.

[19] Hanchuan Peng, Fuhui Long, and Chris Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[20] Mark Andrew Hall, "Correlation-based feature selection for machine learning," 1999.

[21] Xuejun Liao and Lawrence Carin, "Radial basis function network for multi-task learning," in *Advances in Neural Information Processing Systems*, 2006, pp. 792–802.

[22] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil, "Multi-task feature learning," in *Advances in neural information processing systems*, 2007, pp. 41–48.

[23] Shao-Lun Huang, Anuran Makur, Lizhong Zheng, and Gregory W Wornell, "An information-theoretic approach to universal feature selection in high-dimensional inference," in *Information Theory (ISIT), 2017 IEEE International Symposium on*. IEEE, 2017, pp. 1336–1340.