PERFORMANCE GUARANTEES FOR TRANSFERRING REPRESENTATIONS

Daniel McNamara[*]
The Australian National University and Data61
ACT 0200 Australia
daniel.mcnamara@anu.edu.au

Maria-Florina Balcan
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213 USA
ninamf@cs.cmu.edu

ABSTRACT

A popular machine learning strategy is the transfer of a representation (i.e. a feature extraction function) learned on a source task to a target task. Examples include the re-use of neural network weights or word embeddings. Our work proposes novel and general sufficient conditions for the success of this approach. If the representation learned from the source task is fixed, we identify conditions on how the tasks relate to obtain an upper bound on target task risk via a VC dimension-based argument. We then consider using the representation from the source task to construct a prior, which is fine-tuned using target task data. We give a PAC-Bayes target task risk bound in this setting under suitable conditions. We show examples of our bounds using feedforward neural networks. Our results motivate a practical approach to weight sharing, which we validate with experiments.

1 INTRODUCTION

Empirical studies have shown the success of transferring representations between tasks (Donahue et al., 2014; Hoffman et al., 2014; Girshick et al., 2014; Socher et al., 2013; Bansal et al., 2014). Word embeddings learned on a source task have been shown (Qu et al., 2015) to perform better than unigram features on target tasks such as part of speech tagging, and comparably or better than embeddings fine-tuned on the target task. Yosinski et al. (2014) learned neural network weights using half of the ImageNet classes, and then learned the other classes with a neural network initialized with these weights, finding a benefit compared to random initialization only with target task fine-tuning. The transfer of representations, both with and without fine-tuning, is widely and successfully used.

Often a representation is learned by a different organization that may have greater access to data, computational and human resources. Examples are the Google word2vec package (Mikolov et al., 2013) and downloadable pre-trained neural networks.[1] Under this 'representation-as-a-service' model, a user may expect to access the representation itself, as well as information about its performance on the source task data on which it was trained. We aim to convert this into a guarantee of the usefulness of the representation on the user's target task. Our analysis also covers the case where the source task is constructed from unlabeled data, as in neural network unsupervised pre-training.

We consider two approaches to transferring a representation learned from a source task to a target task, as shown in Figure 1. We may either treat the representation as fixed, or we may narrow the class of representations considered on the target task, which we call *fine-tuning*. The fixed option may be attractive when very little labeled target task data is available and hence overfitting is a strong concern, while the advantage of fine-tuning is relatively greater hypothesis class expressiveness.

Let X, Y and Z be sets known as the input, output and feature spaces respectively. Let F be a class of *representations*, where $f : X \rightarrow Z$ for $f \in F$. Let G be a class of *specialized classifiers*, where $g : Z \rightarrow Y$ for $g \in G$. Let the hypothesis class $H := \{h : \exists f \in F, g \in G$ such that $h = g \circ f\}$. Let $h_S, h_T : X \rightarrow Y$ be the labeling functions and P_S, P_T be the input distributions for source task S and target task T respectively. Let the risk of a hypothesis h on S and T be $R_S(h) := \mathbb{E}_{x \sim P_S}[h_S(x) \neq h(x)]$ and $R_T(h) := \mathbb{E}_{x \sim P_T}[h_T(x) \neq h(x)]$ respectively. Let $\hat{R}_S(h)$ and $\hat{R}_T(h)$ be the corresponding empirical (i.e. training set) risks. We have m_S labelled points for S and m_T labelled points for T. Let d_H be the VC dimension of H.

[*] Daniel McNamara was a visitor at Carnegie Mellon University during the period of this research.

[1] For examples see http://code.google.com/archive/p/word2vec, http://caffe.berkeleyvision.org/model_zoo and http://vlfeat.org/matconvnet/pretrained.

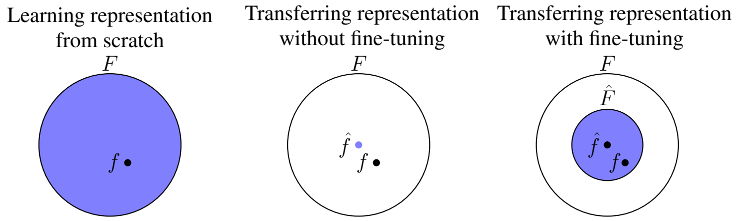
Learning representation from scratch
F
f
Transferring representation without fine-tuning
F
$\hat{f}$
f
Transferring representation with fine-tuning
F
$\hat{F}$
$\hat{f}$
f

Figure 1: A comparison of approaches to learning a representation on a target task, where the search space in each case is the shaded area. Learning from scratch, we search a representation class F for a good representation $f \in F$. Without fine-tuning, we fix a representation $\hat{f}$ learned from the source task. With fine-tuning, we narrow the search to $\hat{F} \subseteq F$ near $\hat{f}$, which still contains f.

2 Representation fixed by source task

Suppose labeled source data is abundant, labeled target data is scarce, and we believe the tasks share a representation. A natural approach to leveraging the source data is to learn $\hat{g}_S \circ \hat{f} \in H$ on S, from which we assume we may extract $\hat{f} \in F$, then conduct empirical risk minimization over $G \circ \hat{f} := \{g \circ \hat{f} : g \in G\}$ on T yielding $\hat{g}_T \circ \hat{f}$. Theorem 1 upper-bounds the risk on T for this method via a VC dimension-based argument involving four terms: a function ω measuring a transferrability property obtained analytically from the problem setting (while the property does not hold in general, see example below where it does hold), the empirical risk $\hat{R}_S(\hat{g}_S \circ \hat{f})$, the generalization error of a hypothesis in H learned from m_S samples, and the generalization error of a hypothesis in G learned from m_T samples. If $\omega(R) = O(R)$, $\hat{R}_S(\hat{g}_S \circ \hat{f})$ is a small constant, $m_S \gg m_T$ and $d_H \gg d_G$, the bound is tighter compared to learning T from scratch since we avoid the generalization error of a hypothesis in H learned from m_T samples. We may use the result to select S given several options.

Theorem 1. *Let $\omega : \mathbb{R} \to \mathbb{R}$ be some non-decreasing function. Suppose P_S, P_T, h_S, h_T, $\hat{f}$, G have the property that $\forall \hat{g}_S \in G$, $\min_{g \in G} R_T(g \circ \hat{f}) \leq \omega(R_S(\hat{g}_S \circ \hat{f}))$. Let $\hat{g}_T := \arg\min_{g \in G} \hat{R}_T(g \circ \hat{f})$. Then with probability at least $1 - \delta$ over pairs of training sets for tasks S and T, $R_T(\hat{g}_T \circ \hat{f}) \leq \omega(\hat{R}_S(\hat{g}_S \circ \hat{f}) + 2\sqrt{\frac{2d_H \log(2em_S/d_H) + 2\log(8/\delta)}{m_S}}) + 4\sqrt{\frac{2d_G \log(2em_T/d_G) + 2\log(8/\delta)}{m_T}}$.*

In Theorem 2 we give an example of the property required by Theorem 1. We consider a neural network with a single hidden layer. We transfer the lower-level weights (corresponding to $\hat{f}$) learned on S, so that only the upper-level weights (corresponding to G) are learned on T. We assume that some lower-level weights perform well on both tasks, and that a point x for which $\hat{f}(x)$ contributes to the risk on T cannot be 'hidden' from the risk of using $\hat{f}$ on S either through low $P_S(x)$ or low magnitude upper-level weights. Hence $R_S(\hat{g}_S \circ \hat{f})$ reliably indicates the usefulness of $\hat{f}$ on T.

The proof of Theorem 2 relies on a novel argument which exploits the following assumptions. Let $X = \mathbb{R}^n$ and $Z = \mathbb{R}^k$. Let F be the function class s.t. $f(x) = [a(w_1 \cdot x), \ldots, a(w_k \cdot x)]$, where $w_i \in \mathbb{R}^n$ for $1 \leq i \leq k$ and $a : \mathbb{R} \to \mathbb{R}$ is an odd function. Let G be the function class s.t. $g(z) = sign(v \cdot z)$, where $v \in \{-1, 1\}^k$. Suppose $\exists f \in F, g_S, g_T \in G$ s.t. $\max[R_S(g_S \circ f), R_T(g_T \circ f)] \leq \epsilon$. Let $\hat{f}(x) := [a(\hat{w}_1 \cdot x), \ldots, a(\hat{w}_k \cdot x)]$. Pick constants α_i and β_i s.t. $||w_i|| = ||\alpha_i \hat{w}_i - \beta_i w_i||$ and $w_i \cdot (\alpha_i \hat{w}_i - \beta_i w_i) = 0$. Let M be a $2k \times n$ matrix with rows $w_1, \alpha_1 \hat{w}_1 - \beta_1 w_1, \ldots, w_k, \alpha_k \hat{w}_k - \beta_k w_k$. Suppose M is full rank. Suppose $\forall x, x'$ s.t. $||Mx|| = ||Mx'||$, $P_T(x) \leq cP_S(x')$ for some $c \geq 1$.

Theorem 2. *Let $\omega(R) := cR + \epsilon(1 + c)$. Then $\forall \hat{g}_S \in G$, $\min_{g \in G} R_T(g \circ \hat{f}) \leq \omega(R_S(\hat{g}_S \circ \hat{f}))$.*

3 Representation fine-tuned using target task

Consider learning $\hat{g}_S \circ \hat{f}$ on S, and then using $\hat{f}$ and $R_S(\hat{g}_S \circ \hat{f})$ to find $\hat{F} \subseteq F$, as in Figure 1. Let $\tilde{h}_{g \circ f}$ be a distribution over H associated with $g \circ f$ (e.g. $g \circ f$ is the mode of $\tilde{h}_{g \circ f}$). We propose learning T with the hypothesis class $\tilde{H}_{G \circ \hat{F}} := \{\tilde{h}_{g \circ f} : f \in \hat{F}, g \in G\}$ and the prior $\tilde{h}_{\hat{g}_S \circ \hat{f}}$. Let $R_T(\tilde{h}) := \mathbb{E}_{x \sim P_T, h \sim \tilde{h}}[h_T(x) \neq h(x)]$ and let $\hat{R}_T(\tilde{h})$ be computed on the training set distribution of T. In Theorem 3 we show that if $\hat{F}$ is 'small enough' that all $\tilde{h} \in \tilde{H}_{G \circ \hat{F}}$ have a small KL divergence from $\tilde{h}_{\hat{g}_S \circ \hat{f}}$, we may apply a PAC-Bayes bound to the generalization error of hypotheses in $\tilde{H}_{G \circ \hat{F}}$. $\hat{F}$ is useful if it is also 'large enough' in the sense that $\exists \tilde{h}_{g_T \circ f} \in \tilde{H}_{G \circ \hat{F}}$ such that $R_T(\tilde{h}_{g_T \circ f}) \leq \epsilon$.

Table 1: Evaluation of transferring representations. Entries are the test set accuracy of the technique (row) for the task (column) averaged over 10 trials, with the best result for each task shown in bold.

TECHNIQUE	MNIST, $\gamma =$			NEWSGROUPS, $\gamma =$		
	0.6	0.8	1	0.6	0.8	1
Learn T from scratch	88.4	87.9	87.9	**62.6**	63.2	66.1
Transfer $\hat{f}$ from S, fine-tune f and train g on T	**91.9**	**93.9**	95.4	62.3	**72.3**	83.3
Transfer $\hat{f}$ from S and fix, train g on T	87.5	92.3	97.3	52.2	69.6	83.3
Transfer $\hat{g}_S \circ \hat{f}$ from S and fix	67.4	85.6	**98.1**	55.5	70.7	**83.6**

Theorem 3. *Let $\omega : \mathbb{R} \to \mathbb{R}$ be non-decreasing. Suppose given $\hat{f} \in F$ and $R_S(\hat{g}_S \circ \hat{f})$ estimated from S, it is possible to construct $\hat{F}$ with the property $\forall \tilde{h} \in \tilde{H}_{G \circ \hat{F}}$, $KL(\tilde{h} || \tilde{h}_{\hat{g}_S \circ \hat{f}}) \leq \omega(R_S(\hat{g}_S \circ \hat{f}))$. Then with probability at least $1 - \delta$ over pairs of training sets for tasks S and T, $\forall \tilde{h} \in \tilde{H}_{G \circ \hat{F}}$, $R_T(\tilde{h}) \leq \hat{R}_T(\tilde{h}) + \sqrt{\frac{\omega(\hat{R}_S(\hat{g}_S \circ \hat{f}) + 2\sqrt{\frac{2d_H \log(2em_S/d_H) + 2\log(8/\delta)}{m_S}}) + \log 2m_T/\delta}{2(m_T - 1)}}$.*

We transfer and fine-tune weights in a feedforward neural network with one hidden layer to instantiate the property required by Theorem 3. We learn a deterministic hypothesis of this type on S and obtain estimated lower-level weight vectors $\hat{w}_i$. For T we consider hypotheses formed by adding independent sources of noise to the weights of a deterministic network, using lower-level weights near $\hat{w}_i$ (corresponding to $\hat{F}$). We assume some lower-level weights w_i perform well on both S and T. We make $\hat{F}$ 'small enough' by only including lower-level weights with small angles to $\hat{w}_i$, and 'large enough' by upper-bounding the angle between each pair w_i and $\hat{w}_i$ in terms of the risk using $\hat{w}_i$ on S. Using similar assumptions to those for Theorem 2, we derive a comparable result.

4 MODIFIED REGULARIZATION PENALTY

Relaxing the hard constraint on $\hat{F}$ motivates a loss function with modified regularization (1). Let y_i and $\hat{y}_i$ be the label and prediction respectively for the ith training point. In a fully-connected feedforward network with l layers of weights, let $W^{(j)}$ be the jth weight matrix, $\hat{W}^{(j)}$ be its estimate from S (excluding weights for bias units in both cases), and $||\cdot||_2$ be the entry-wise 2 norm. Since we expect the tasks to share a low-level representation (e.g. edge detectors for vision, word embeddings for text) but be distinct at higher levels (e.g. image components for vision, topics for text), we set $\lambda_1(\cdot)$ to be a decreasing function, while $\lambda_2(\cdot)$ controls standard L2 regularization. The technique is novel to our knowledge, although other approaches to transferring regularization between tasks exist (Evgeniou & Pontil, 2004; Raina et al., 2006; Argyriou et al., 2008; Ghifary et al., 2014).

$$\sum_{i=1}^{m} [-y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i)] + \sum_{j=1}^{l} [\frac{\lambda_1(j)}{2} ||W^{(j)} - \hat{W}^{(j)}||_2^2 + \frac{\lambda_2(j)}{2} ||W^{(j)}||_2^2] \quad (1)$$

We experiment on basic image and text classification datasets.[2] We randomly partition label classes into sets S_+ and S_-, where $|S_+| = |S_-|$. We construct T_+ by randomly picking from S_+ up to $\gamma := \frac{|S_+ \cap T_+|}{|S_+|}$, then randomly picking from S_- such that $|T_+| = |T_-|$. We let S be the task of distinguishing between S_+ and S_- and T be that of distinguishing T_+ and T_-. We set $\lambda_1(1) = \lambda_2(2) = \lambda := 1$, $\lambda_1(2) = \lambda_2(1) = 0$, $m_T = 500$ and use the sigmoid activation. For MNIST we use raw pixel intensities, a 784 × 50 × 1 network and $m_S = 50000$. For NEWSGROUPS we use TF-IDF weighted counts of most frequent words, a 2000 × 50 × 1 network and $m_S = 15000$.

For MNIST, fine-tuning with (1) outperforms learning T from scratch with a $\frac{\lambda}{2}||W^{(j)}||_2^2$ penalty for all j (see Table 1). It appears that learning a digit requires a dense weight vector, so that $\hat{W}^{(1)}$ tends to encode single digits. On NEWSGROUPS it appears we may learn a newsgroup with a sparse weight vector and so $\hat{W}^{(1)}$ tends to encode disjunctions of newsgroups, somewhat reducing transferrability.

[2] The MNIST and 20 Newsgroups datasets are available at `http://yann.lecun.com/exdb/mnist` and `http://qwone.com/~jason/20Newsgroups` respectively.

Acknowledgments

Daniel McNamara's research visit to Carnegie Mellon University was supported by a Fulbright Postgraduate Scholarship.

References

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. Tailoring continuous word representations for dependency parsing. In *Association for Computational Linguistics*, pp. 809–815, 2014.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. DeCAF: a deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning*, pp. 647–655, 2014.

Theodoros Evgeniou and Massimiliano Pontil. Regularized multitask learning. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 109–117, 2004.

Muhammad Ghifary, W Bastiaan Kleijn, and Mengjie Zhang. Domain adaptive neural networks for object recognition. In *Pacific Rim International Conference on Artificial Intelligence*, pp. 898–904. Springer, 2014.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. LSDA: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pp. 3536–3544, 2014.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pp. 3111–3119, 2013.

Lizhen Qu, Gabriela Ferraro, Liyuan Zhou, Weiwei Hou, Nathan Schneider, and Timothy Baldwin. Big data small data, in domain out-of domain, known word unknown word: the impact of word representation on sequence labelling tasks. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pp. 89–93, 2015.

Rajat Raina, Andrew Y. Ng, and Daphne Koller. Constructing informative priors using transfer learning. In *International Conference on Machine Learning*, pp. 713–720, 2006.

Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems*, pp. 935–943, 2013.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pp. 3320–3328, 2014.