Model Performance Scaling with Multiple Data Sources

Tatsunori Hashimoto [1]

Abstract

Real-world machine learning systems are often trained using a mix of data sources with varying cost and quality. Understanding how the size and composition of a training dataset affect model performance is critical for advancing our understanding of generalization, as well as designing more effective data collection policies. We show that there is a simple scaling law that predicts the loss incurred by a model even under varying dataset composition. Our work expands recent observations of scaling laws for log-linear generalization error in the i.i.d setting and uses this to cast model performance prediction as a learning problem. Using the theory of optimal experimental design, we derive a simple rational function approximation to generalization error that can be fitted using a few model training runs. Our approach can achieve highly accurate ($r^2 \approx .9$) predictions of model performance under substantial extrapolation in two different standard supervised learning tasks and is accurate ($r^2 \approx .83$) on more challenging machine translation and question answering tasks where many baselines achieve worse-than-random performance.

1. Introduction

The success of large scale machine learning systems depends critically on the quantity and quality of data used during training, and we cannot expect these systems to succeed if there is not enough training data or if that data does not cover all the phenomena contained in the test distribution (Ben-David et al., 2010). Knowing this, the designer of a machine learning system might create multiple sources of data, with each one targeting a different feature or domain that the model ought to do well on (Crammer et al., 2007; Wang et al., 2019a). This data-driven design strategy provides powerful tools to improve and evaluate model behavior, but also poses an additional challenge: what is the right way to combine these various data sources? What is the optimal data collection policy for a given budget?

Our goal is to answer these questions by quantifying the relationship between data sources and model performance. How well will our model do if we were to train it on n samples using a data mixture $(q_1 \dots q_k)$ (where q_i is the fraction of the dataset coming from data source i). A precise model for predicting model performance will allow us to both identify the optimal data collection policy and quantify cost-performance tradeoffs.

The starting point of our work is the recent observation across speech, vision and text (Hestness et al., 2017; Kaplan et al., 2020; Rosenfeld et al., 2020) that the empirical performance of a model is remarkably predictable, and follows the log-linear formula

$$\log(\text{error}) \approx -\alpha \log(n) + C. \tag{1}$$

In this work, we expand this observation to the multi-data-source setting and conjecture that the slope of the log-linear relationship (α) does not vary with data composition and that the data composition only affects the intercept (C). We prove this holds in a range of parametric and nonparametric models.

The simple dependence of log-error on data size allows us to reduce the problem of estimating model error into a learning problem. Our approach is straightforward. First, we hypothesize that model error follows $V(n, q) := \exp(-\alpha \log(\text{n}) + \log(C(q)))$ for a simple parametric functional form $C(q)$. Next, we fit this functional form to observed pairs of (n, q, error) that we obtain by subsampling the dataset and re-training a model. We show that there is a natural and simple choice of $C(q)$ as a rational function that we derive from optimal experimental design for linear regression, M-estimation, and nonparametric smoothing. The simple and parametric dependence of $V(n, q)$ on n allows us to use our resulting estimates to predict model performance under substantial extrapolation in data size.

As a concrete example of how this may be useful, consider the Amazon sentiment prediction task (Mansour et al., 2009), where we have the ability to collect review data from multiple product categories. A practitioner may wish to un-

[1]Work done while author was at Microsoft Semantic Machines. Correspondence to: Tatsunori Hashimoto <v-hashimotot@microsoft.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

derstand how the dataset size n and the mixture proportion over these categories q affect model performance, so that they can better understand tradeoffs in data collection. To do this, they can collect a small pilot dataset with data from all sources, and then subsample this pilot dataset in order to vary the data size n and mixture proportion q. Measuring the errors of models trained on these subsets now gives us the triples of (n, q, error), and they can fit a model $V(n, q)$ to predict error on these examples. The resulting $V(n, q)$ can be used as a way to improve the data collection policy q for substantially larger dataset sizes n than the pilot dataset.

Empirically, we show that the rational function approximation is a promising approach, and that the resulting predictions are accurate and hold under extrapolation. On the Amazon review prediction dataset (Mansour et al., 2009), we can learn to predict model performance nearly perfectly ($r^2 = 0.96$) from a small dataset of 1200 examples across 3 sources and extrapolate to predict the model error on datasets of up to 4000 examples. We show this high accuracy continues to hold on a real-world task oriented dialogue system ($r^2 = 0.89$), a multi-domain machine translation system ($r^2 = 0.83$), and boolean question answering with weak supervision ($r^2 = 0.85$). In each of the cases, our proposed approach matches or outperforms the best baseline, with most baselines performing worse-than-random in both the machine translation and question answering tasks.

Related work Quantifying the effect of data composition on model performance is closely related to the classical ideas of optimal experimental design, as well as more recent machine learning methods such as active learning and data valuation.

Our work will draw inspiration from the classical V-optimal experimental design (John & Draper, 1975) as a way to understand how model performance will change with the data collection policies. However, our approach differs substantially beyond this. Instead of making strong linearity assumptions and identifying closed form formulas for model performance, we treat identifying the impact of data sources on errors as itself a prediction problem, which allows us to quantify these effects for neural networks and non-separable objectives.

Scaling laws (Kaplan et al., 2020; Hestness et al., 2017) and empirical prediction of model performance (Kolachina et al., 2012) are closely related to our work and share our motivation of identifying relationships between data and model performance. Our work differs in studying the multi-data-source settings, which pose substantial additional challenges due to the non-i.i.d nature of the training and test distributions.

Active learning provides methods for incrementally selecting new points to rapidly reduce a loss (Hanneke, 2007). Implicitly, these methods often rely upon an estimate of how data collection affects downstream performance (Ghorbani & Zou, 2019). However, these approaches only consider the problem of optimal data collection and do not seek to predict model performance under *all* data collection strategies (including suboptimal ones), which is critical when making cost-performance tradeoffs across data sources. The model performance predictions produced in our work complements existing work on active learning by providing accurate forecasts of model performance under different data collection strategies.

Finally, data valuation methods such as the Shapley value attempt to assign estimate the impact of a data source on model performance (Ghorbani & Zou, 2019; Jia et al., 2019; Ghorbani et al., 2020; Yoon et al., 2019). These approaches are natural when pricing data sources as part of a market mechanism (Ohrimenko et al., 2019; Agarwal et al., 2019) due to the axiomatic properties of the Shapley value. Our approach differs in that we seek simply to estimate the performance of a model rather than to assign a single price to examples from a data source. This difference means that axioms such as *additivity* that are critical for the Shapley value are not relevant for our goal. We show that for the purpose of predicting errors, a rational function (rather than a linear cost) follows naturally from optimal experimental design. Our experiments also suggest that our rational function approximation provides better model performance predictions than a linear, additive model.

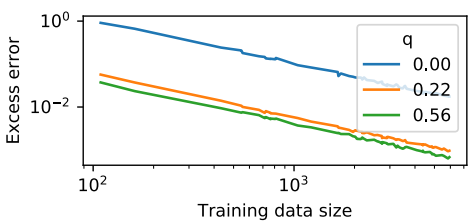
2. Problem setting

Our goal is to predict the performance of a model as a function of the number of training samples n as well as the dataset composition q, where q_k represents the fraction of the training data drawn from data source k. We will now define this goal more formally in terms of the training data distribution, model fitting, and test loss.

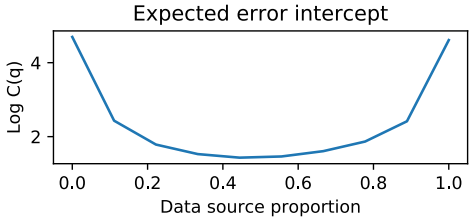
The training data consists of an n-sample training set $p_{n,q}$ that is created by sampling from the mixture $p := \sum_{k\in[K]} q_k p_k$ where p_k are data generating distributions for each of the K data sources and q_k are mixture weights with $q_k \geq 0$ and $\sum_{k\in[K]} q_k = 1$. Using this dataset, we learn a prediction model $\hat{\theta}$ that incurs loss $\ell(\hat{\theta}; x, y)$ for a training example (x, y). The fitted model is the empirical loss minimizer, which we define as

$$\hat{\theta}(p_{n,q}) := \arg\min_{\theta\in\Theta} \mathbb{E}_{p_{n,q}} \left[\ell(\theta; x, y)\right].$$

The performance of this classifier is evaluated on a test distribution with the same conditional label distribution (i.e. $p(y \mid x) = p_{\text{test}}(y \mid x)$). We are interested in model performance as a function of the data size and composition (and not a fixed empirical distribution $p_{n,q}$) and thus our



(a) Log excess loss (y axis) is linear with log-dataset size (x axis). Changing the data distribution by varying q (line color) changes the intercept but not the slope.



(b) Intercept ($C(q)$) of the loss-dataset log linear relationship. The loss is lowest when the dataset is a mix of both data sources ($q \approx 0.5$) and rapidly increases when exclusively using one data source.

Figure 1. The log-linear effects of data composition and size on the linear toy dataset.

goal is to predict the model's expected excess loss averaged over draws in both the training and test distributions,

$$L(n, q) := \mathbb{E}\left[\ell(\hat{\theta}(p_{n,q}); x, y)\right] - \inf_{\theta} \mathbb{E}\left[\ell(\theta; x, y)\right].$$

Estimating L requires that we hypothesize a relationship between (n, q) and the expected model loss. Following earlier observations by Hestness et al. (2017), we expect a log-linear relationship between $L(n, q)$ and $\log(n)$ for any fixed q, which implies a possible approximation as

$$\log(L(n, q)) \approx \log(V(n, q)) := \alpha(q)\log(n) + C(q). \quad (2)$$

We now examine this hypothesis in a simple toy example.

Linear toy data: We will start with the simplest nontrivial example of linear least-squares regression to study $L(n, q)$. In this example, there are two data sources over $x \in \mathbb{R}^2$. The first data source has substantial variability on the first coordinate x_0 but not x_1 and vice versa for the second data source. The overall generative process is

$$y \mid x \sim [0.5, 1]^\top x + \epsilon \qquad z \sim \text{Bern}(q) \qquad \epsilon \sim N(0, 1)$$
$$x \mid z = 0 \sim N\left(0, \begin{bmatrix} 1 & 0 \\ 0 & 0.001 \end{bmatrix}\right)$$
$$x \mid z = 1 \sim N\left(0, \begin{bmatrix} 0.001 & 0 \\ 0 & 1 \end{bmatrix}\right).$$

Let $L(n, q)$ be the excess squared loss of a linear least squares model trained with n samples from a mixture q and evaluated on a test distribution with $q = 0.5$. What will $L(n, q)$ look like? Figure 1a shows a clear linear relationship between log dataset size ($\log(n)$) and $\log(L(n, q))$. The intercept of the linear relationship seems to vary with the data mixture q, but the slope seems constant.

Examining Figure 1a more closely, we find that the extremes of using either data source exclusively (blue line) performs worse than a mix suggesting that $\log(L(n, q))$ is unlikely to be linear in q. Intuitively, we can think of each data distribution as having a different strength (i.e. more variance in either x_0 or x_1) and combining the two results in a better data distribution than either alone. We can see this more clearly when we estimate the intercept for each of these lines (Figure 1b). The estimated intercepts show a U-shaped curve that rapidly increases as $q \to 0$ or $q \to 1$ and is generally flat from 0.2 to 0.8.

3. Method and theory

We have observed that in the case of a simple linear regression, the log-error not only follows the relationship outlined in equation 2, but also that the slope α is constant as we vary the data composition (and we will further validate this claim on more complex tasks and models in subsequent sections). This observation shows we may be able to further simplify the log-linear approximation as

$$\log(L(n, q)) \approx \log(V(n, q)) := -\alpha \log(\text{n}) + \log(C(q)).$$

Now note that this functional form decouples the data size n and mixture proportions $C(q)$ into two terms. This is the key hypothesis of our work: $\log(V(n, q))$ has a very simple dependence on n, and the more complex term $C(q)$ has no dependence on n. Therefore we can cast this as a learning problem, where we learn α and a parametric function $C_\lambda(q)$ based on the model's error over a range of q and small n, and extrapolate this for large n using the log-linear dependence of $\log V$ on n.

Concretely, we are given a dataset that is comprised of k data sources, where each data source contributes $\{n_1 \dots n_k\}$ examples. To predict the performance of a model under varying data composition, we take the following steps.

First, we generate a subsampled dataset with $\hat{n}_k \sim \text{Unif}(0, n_k)$ samples from each source. This results in a training set with data size $\hat{n} = \sum_k \hat{n}_k$ and composition $\hat{q}_k = \frac{\hat{n}_k}{\hat{n}}$.

Next, we fit a model to this subsampled data and compute

its loss $R(\hat{n}, \hat{q}) = \mathbb{E}\left[\ell(\hat{\theta}(p_{\hat{n},\hat{q}}); x, y)\right]$. Given the triple $(\hat{n}, \hat{q}, R(\hat{n}, \hat{q}))$ we fit the hypothesized functional form,

$$\min_{\lambda,\alpha} \mathbb{E}_{\hat{q},\hat{n}}\left[(\log(R(\hat{n},\hat{q}) - \epsilon) - \alpha\log(\hat{n}) + \log(C_\lambda(\hat{q})))^2\right].$$

Here, ϵ approximates the optimal asymptotic error $\inf_\theta \mathbb{E}\left[\ell(\theta; x, y)\right]$ (which is the Bayes error rate whenever the model is well specified) and $L(n, q) \approx R(n, q) - \epsilon$. This approach of modeling excess loss with respect to asymptotic error is standard in existing work on scaling laws (Kaplan et al., 2020; Hestness et al., 2017; Rosenfeld et al., 2020).

Finally, given the fitted α and $C_\lambda(\hat{q})$, we can predict the performance of any model by extrapolating to $-\alpha\log(n) + \log(C_\lambda(q))$.

This approach of predicting model performance is reminiscent of response surface methods (Belkhir et al., 2017) but we have an additional challenge that we do not have a good estimate of $C_\lambda(\hat{q})$. We find empirically that generic function approximators such as multilayer neural networks do not perform well.

The experimental data does not specify the functional form of $C_\lambda(q)$ except that it should handle convex functions like those seen in Figure 1b. We will now study $V(n, q)$ theoretically and argue that a natural choice is the rational function

$$C_\lambda(q) := \sum_{i=1}^{M}\left(\sum_{k=1}^{K}\lambda_{ik}q_k\right)^{-1}.$$

In the subsequent sections, we will study three settings: ordinary linear regression, M-estimation, and nonparametric regression and show that our hypothesized log-linear approximation arises naturally in all three cases.

3.1. Linear regression

We begin by characterizing $L(n, q)$ in the linear regression case, where we can derive closed form expressions for the expected loss as a function of training data. Our setting is d-dimensional, n-sample linear regression, defined as $y = x^\top\beta + \epsilon$ with i.i.d. $\epsilon \sim N(0, 1)$. Our training data follows $x \sim p := \sum_{k\in[K]} q_k p_k$ where each data source has full-rank second moments $\Sigma_k := \mathbb{E}_{x\sim p_k}\left[xx^\top\right]$.

Define the ordinary least squares estimator $\hat{\beta} := (X^\top X)^{-1}X^\top Y$ in terms of the features $X \in \mathbb{R}^{n\times d}$ and $Y \in \mathbb{R}^n$. The excess test loss of this estimator over any $x^* \sim p^*$ and $y^* := x^{*\top}\beta + \epsilon$ is defined as

$$L(n, q) = \mathbb{E}[\|x^*(\beta - \hat{\beta})\|_2^2].$$

The theory of V-optimal experimental design(Pukelsheim, 2006) allows us to characterize this excess loss.

Proposition 3.1. *The excess expected loss for ordinary least squares trained on a mixture q with data size n and sub-gaussian x follows*

$$\log(L(n, q)) = -\log(n) + \log\Bigg(\underbrace{Tr\left(\Sigma^*\left(\sum_k q_k\Sigma_k\right)^{-1}\right)}_{C(q)}\Bigg) + O\left(\frac{\sqrt{\log(1/\delta)}}{\sqrt{n}}\right),$$

with probability at least $1 - \delta$ where $\Sigma^* := \mathbb{E}_{x\sim p^*}\left[xx^\top\right]$ and $\Sigma_k := \mathbb{E}_{x\sim p_k}\left[xx^\top\right]$.

We will defer all proofs to the supplement due to space constraints. Clearly $C(q)$ is not linear even in this simple case, and the terms for q_k appear within an inverse. Naively, we might hypothesize that it behaves much more closely to a linear rational function (i.e. $(\sum_i \lambda_i q_i)^{-1}$) and this intuition will turn out to be correct whenever Σ^* and Σ_k are approximately diagonalizable.

Corollary 3.1. *Let P be an orthogonal matrix which approximately simultaneously diagonalizes $P^{-1}\Sigma^*P = D^*$, $P^{-1}\Sigma_k P = D_k + R_k$ for diagonal some matrices D. Then for full-rank Σ^* and sufficiently small R_k,*

$$Tr\left(\Sigma^*\left(\sum_{k\in[K]} q_k\Sigma_k\right)^{-1}\right) = \sum_{i\in[d]}\frac{D^*_{ii}}{\sum_k q_k D_{k,ii}} + o\left(\|\sum_k q_k R_k\|_F\right).$$

The first order term exactly matches the hypothesized $C(q)$ as a rational function with d terms and validates this choice for linear regression. To interpret this corollary, the approximate diagonalizability condition states that the eigenvectors for Σ^* and Σ_k coincide, and that D^*_{ii} and $D_{k,ii}$ are these eigenvalues. The ratio $\frac{D^*_{ii}}{\sum_k q_k D_{k,ii}}$ measures the ratio of variance in the test distribution to that of the training distribution for the i-th eigenvector.

The key observation is that the variance (i.e. the information each data source contributes to a particular coordinate i) is linear, but the dependence of model error to training variance is *inverse* and that there are d different coordinates making the overall dependence of errors on data composition nonlinear. There are clear qualitative differences between a linear and rational function approximation to $C(q)$, with the rational function being strongly convex with diminishing returns in q.

3.2. General M estimators

We might rightfully ask whether this kind of approximation continues to hold for nonlinear models and losses like neural

networks. The same analysis as above can be extended to the asymptotic behavior of a substantially more general class of models known as M-estimators, which are empirical loss minimizers of a differentiable loss.

For the regression case, we relied on a closed-form characterization of β. For M-estimators we will use asymptotic normality under the sampling distribution,

Theorem 3.1 (van der Vaart (1998))**.** *Consider a twice differentiable loss ℓ whose gradients are bounded and Donsker. Let θ_n be an estimator which fulfills the approximate first-order optimality condition with minimizer θ_∞,*

$$\mathbb{E}_{p_n}[\nabla \ell(y, x; \theta_n)] = o(n^{-1/2}) \quad \textit{and} \quad \mathbb{E}_p[\nabla \ell(y, x; \theta_\infty)] = 0.$$

If $\theta_n \xrightarrow{p} \theta_\infty$ and both $I_{\theta_\infty}^{-1} := \mathbb{E}_p[H\ell(y, x; \theta_\infty)]^{-1}$ and $\Sigma_{\theta_\infty} := \mathbb{E}_p[\nabla \ell(y, x; \theta_\infty) \nabla \ell(y, x; \theta_\infty)^\top]$ exist,

$$\sqrt{n}(\theta_n - \theta_\infty) \to N\left(0, I_{\theta_\infty}^{-1} \Sigma_{\theta_\infty} I_{\theta_\infty}^{-1}\right).$$

As with earlier, we defer all proofs to the supplement.

Now that we have the asymptotic distribution of the M-estimator, we can quantify the (asymptotic) form of $C(q)$ with respect to a test distribution p^* simply by taking the Taylor expansion of the loss at θ_∞.

Corollary 3.2. *Under the conditions of Theorem 3.1, let $\ell(y, x; \theta) = -\log p_\theta(y \mid x)$ and there exists some $\theta^* = \theta_\infty$ such that $p_{\theta^*}(y \mid x) = p(y \mid x)$ then*

$$\log(L(n, q)) = -\log(n) + \log\left(Tr\left(\Sigma^* \left(\sum_k q_k \Sigma_k\right)^{-1}\right) + o(n^{-1})\right).$$

for $\Sigma_k := \mathbb{E}_{p_k}[H\ell(y, x; \theta^*)]$ and $\Sigma^* := \mathbb{E}_{p^*}[H\ell(y, x; \theta^*)]$

Note how this has the same functional form as before: $C(q)$ is the trace of a test distribution dependent matrix Σ^* and the inverse of data source matrices Σ_k. The difference now is that instead of covariances, we are looking at the Hessian of the parameters with respect to the unknown optimal model θ^*. Applying the simultaneous diagonalization argument from earlier once again results in a rational function that is captured by $C(q)$.

Our result here relies on two additional assumptions: the loss is a log loss, and the model is well-specified. The first assumption is weak, as many models today use log softmax type losses. The well-specified assumption is stronger but may be reasonable for nearly nonparametric functions such as neural networks. We relax this assumption in a corollary below.

Corollary 3.3. *Under the conditions of Theorem 3.1 and either $\mathbb{E}_{p^*}[\nabla \ell(y, x; \theta_\infty)] = 0$ or $\mathbb{E}[\theta_n] = \theta_\infty + o(n^{-1})$,*

$$\log(L(n, q)) := \log(\mathbb{E}[\ell(y, x; \theta_n)] - \mathbb{E}[\ell(y, x; \theta_\infty)]) = -\log(n) + \log\left(Tr\left(\mathbb{E}_{p^*}[H\ell(y, x; \theta_\infty)] I_{\theta_\infty}^{-1} \Sigma_{\theta_\infty} I_{\theta_\infty}^{-1}\right) + o(n^{-1})\right).$$

When the model is well-specified, $I_{\theta_\infty} = \Sigma_{\theta_\infty}$ and we recover our earlier result. Corollary 3.3 captures all of the parametric situations above, including well-specified linear regression but also includes common other models that have not been covered such as ridge regression.

3.3. Nonparametric models

Finally, we show that the same relationship holds for nonparametric models such as kernel smoothing or binning. Our goal will be to estimate some ground truth map $y = f(x) + \epsilon$ for ϵ i.i.d $N(0, 1)$ and f a differentiable L-Lipschitz function. The quality of an estimate will be measured by some twice-differentiable loss $\ell(y, x)$ with bounded first two derivatives.

Given n samples $(x_1, y_1) \dots (x_n, y_n) \in [0, 1]^d \times \mathbb{R}$ drawn i.i.d from some density $p = \sum_k q_k p_k$, one natural estimator for this problem is the nonparametric binning estimator $\hat{f}$ which we define in terms of axis-aligned hypercubes $B_\delta(x, S) := \{x' \in S : \lfloor x'/\delta \rfloor = \lfloor x/\delta \rfloor\}$. Let $X_n := \{x_1 \dots x_n\}$ then we can define our estimator,

$$\hat{f}_\delta(x) := \frac{1}{|B_\delta(x, X_n)|} \sum_{x_i \in B_\delta(x, X_n)} y_i.$$

Assuming we choose δ and n sufficiently large that each bin concentrates to its expected value, we have the following error estimate

Proposition 3.2. *Let $B_\delta(x, p_k) = \mathbb{E}_{x' \sim p_k}[|B_\delta(x, \{x'\})|]$ be the probability of drawing $x' \sim p_k$ in the same bin as x, and assume $B_\delta(x, p_k)$ is bounded away from zero. Then*

$$\log(L(n, q)) := \log(\mathbb{E}[\ell(\hat{f}_\delta(x), x) - \ell(f(x), x)]) = -\log(n) + \log\left(\mathbb{E}\left[\frac{\ell''(f(x), x)}{\sum_k q_k B_\delta(x, p_k)}\right] + O\left(\frac{\sqrt{\log(\gamma^{-1}) + d\log(\delta)}}{\sqrt{2n}}\right) + O(L\delta\sqrt{d} + L^2\delta^2 d)\right),$$

holds with probability at least $1 - \gamma$, where the expectation is taken with respect to draws of y.

Once again, we see a rational function in q, with no further approximation needed. Each bin is a term in the rational function approximation with weight $\ell''(x)$.

4. Experiments

We have seen that a rational function is a reasonable approximation to $C(q)$ across 3 different settings. We will now show that this is the case in practice, and additionally that $C(q)$ can be accurately estimated using a few models trained on small datasets. The resulting estimates of model performance are accurate for models with an order of magnitude more data.

Baselines and implementation Our evaluations focus on our ability to predict the loss incurred by a model $L(n, q)$. To do so, we will compare the rational function approximation procedure against several natural baselines for predicting the loss of a model. Each of the baselines correspond to a different assumption about the functional form of $\log(V(n, q))$ that we use to approximate $\log(L(n, q))$.

Datasize: Assume a functional form of $\log(V(n, q)) = \alpha \log(n) + c$ ignoring the data composition and dependence on q. We solve for α via least squares regression in closed form.

Linear: Assume a functional form of $\log(V(n, q)) = \alpha \log(n) + \beta^\top q + c$. This is the natural approach if we treat $\log(V(n, q))$ as linear in q and log-linear in n. As with the datasize baseline, we solve for the parameters using least squares regression.

Ablation and **Shapley**: further constrain the linear baseline by setting β to either the log-Shapley value obtained as the marginal contribution of a data source (for the Shapley baselines) or the log-ratio of losses obtained after removing a data source (ablation). We use this approach as we found it to dominate the usual assumption of treating $V(n, q)$ as being linear in the Shapley value.

MLP (small, medium, large): a multi-layer fully connected neural network with tanh nonlinearities that directly regresses $\log(V(n, q))$ as a function of $\log(n)$. The **small** model has 1 layer and hidden units equal to K={number of data sources +1}; **medium** has K layers and K hidden units; **large** has a depth of 5 and 50 hidden units each. We train these three models to show that generic nonlinear regression models do not necessarily succeed at extrapolation.

We will refer to our approach as **Rational**, and we fit this using the Adagrad (Duchi et al., 2010) optimizer with 20000 steps and learning rate set over the interval [0.005, 0.5] via goodness-of-fit on a held out set. We re-parametrize the weights λ by log-transforming them for numerical stability, and initialize it with a Xavier initialization. This prevents degeneracies near $\lambda = 0$ and we empirically found the optimization process to be stable over the cross-validation range we used. We fixed the number of factors in the rational approximation (M) to one greater than the number of data sources to reduce the number of hyperparameters to tune. We found $\epsilon = 0$ to work well on the regression and classification datasets, and we use this value throughout.

4.1. Focused evaluation: Amazon sentiment

We now consider the Amazon sentiment prediction regression dataset in Mansour et al. (2009) where the goal is to predict Amazon ratings for books (from 0 to 5 stars) using bag-of-words features from the reviews. The training data comes from 3 domains that differ from the test data: kitchen, DVD, and electronics reviews. The model is a standard ridge regularized regression model; we add the ridge regularization term in order to show that Proposition 3.1 continues to hold even when the assumptions are slightly violated. Our experimental setup for estimating model loss is the following: we uniformly randomly sample the dataset size for each source (resulting in between 0 and 1200 examples for each source), and train a model on this dataset. We measure the test error via average squared loss on the books domain.

We fit $V(n, q)$ with 4 terms for $C(q)$ by minimizing the squared loss with respect to log-error on models containing 0-1200 examples total with $\epsilon = 0$. We then use $V(n, q)$ to predict log-error on the models trained on 1200-3600 examples from each domain. The results of this extrapolation task are shown in Table 2. Our $V(n, q)$ estimate is highly accurate ($r^2 = 0.96$) and extrapolate from the low data to high data regime without issue. This correlation is substantially higher than either using data set size ($r^2 = -0.65$), a linear model ($r^2 = 0.76$) and even better *the training error* of the best additive model ($r^2 = 0.87$). The MLP models all underperform the rational function approximation and even substantially larger capacity models do not help, demonstrating that the performance of our approach is not merely due to a more flexible family of predictors. The MLP (medium) model achieves the best fit and substantially larger models that fit the training data well (MLP large) do not improve performance. Attempting to optimize the MLP model further for this task by varying the hidden unit sizes did not help, as both increasing and decreasing the number of hidden units per layer resulted in decreased performance. Finally, this experiment used a ridge regression model which deviates from the least-squares regression analyzed in the theory section. We find that increasing the ridge penalty to make this gap larger leads to even better results, with the r^2 for the rational function approximation increasing to 0.96 as the regularization strength is varied from 300 to 1000.

The data size predictor has a negative r^2 on the extrapolation setting which may seem surprising. However, this can happen whenever a predictor fails to perform better than predicting the mean of the test set. It is nontrivial to predict the mean of the test set in an extrapolation setting, and in this case, data size estimates are generally uninformative as

Metric	Datasize	Ablation	Shapley	Linear	Rational	MLP (small)	MLP (medium)	MLP (large)
Train	0.20	0.77	0.80	0.87	0.96	0.93	0.97	0.99
Extrapolation	-0.65	0.43	0.51	0.76	**0.96**	0.23	0.75	0.57

Table 1. Accuracy of $L(n, q)$ estimates on the Amazon review sentiment prediction task. Bold indicates the best performing model under extrapolation, identified by a bootstrapped paired difference test.

data from the kitchen domain is less useful for predicting book review scores. Next, we will examine the limits of predicting model performance by considering two additional settings: when the estimates $V(n, q)$ are fitted only for a small subset of q (extrapolation on q) and when the ratio of training and testing data sizes exceed a factor of 10.

Extrapolation over q In our previous experiment, we subsampled a subset of the Amazon sentiment reviews dataset to obtain model performance measurements for a wide range of qs. While this emulates how we estimate $V(n, q)$ from a pilot dataset, we may additionally be interested in an ablation experiment where we intentionally restrict the set of values q that are used to fit $V(n, q)$. This allows us to further validate our rational function approximation by testing for extrapolation in both n and q.

We use the Amazon sentiment prediction task from before, but restrict the set of qs by ensuring that all examples in the training set used to fit $V(n, q)$ have $q_i < 1/3$ for the q_i corresponding to the "kitchen" category. This means that the model must learn to estimate the value of examples from the kitchen domain when most of the dataset consists of dvd and electronics reviews. We find only minor degradation in the performance of the rational function model, with $r^2 = 0.92$. The best baseline (linear) shows slight improvements ($r^2 = 0.83$) due to the fact that the kitchen examples contribute substantially to the nonlinearity of $V(n, q)$, but does not match the performance of the rational function approximation. The MLP methods perform worse-than-random in this setting, suggesting that extrapolation over q is substantially more challenging using an arbitrary function approximator.

Large train-test gaps in n We may also be interested in substantially larger extrapolation settings, beyond the factor of 4 scaling considered earlier. To test this, we changed the train-test split for the earlier Amazon experiment to split by $0 - 360$ training examples and $360 - 3600$ test. This results in a close to a factor of 10 gap in data size – going beyond this made the training uninformative, as models trained with fewer than 100 examples per category have extremely high error and variance. This substantially degrades the performance of all models, but the overall conclusions remain similar: the rational function approximation has relatively high predictive power ($r^2 = 0.77$) with substantial gaps to the best baseline (linear, $r^2 = 0.65$). Once again, we find that the MLP based approaches perform worse than random and degrade substantially in these more challenging situations. For more complex experimental settings in the next section, we were unable to achieve satisfactory performance by any of the methods at $10\times$ extrapolation, and we view this as an interesting future work to build predictors for more extreme extrapolation in general settings.

4.2. Broad evaluation: semantic parsing, translation, and question answering

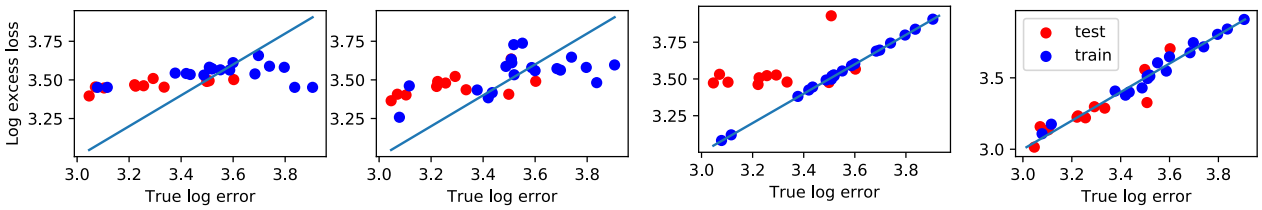
We now perform a shallow but broader evaluation of the 3 methods (linear, rational, MLP, and datasize) on 3 tasks that violate our assumptions about model performance prediction. We excluded the two ablation based methods as they are special cases of the linear model, and generally performed worse.

Task-oriented dialogue We perform this analysis on a real world task-oriented dialogue system that the SMCalFlow dataset and model (Andreas et al., 2020) is based on. The task differs from the Amazon setting in two ways: the model is a nonlinear neural model for which there is no closed form optimal experimental design and the task is semantic parsing which is a more complex structured prediction problem. There are 105727 total dialogues across 4 data sources consisting of a wizard-of-oz style crowdsourced dialogues, paraphrases of existing dialogues, on-policy dialogues between the system and crowdworkers, and hand-crafted dialogues by expert data scientists. We sample the number of dialogues for each source with a uniform distribution to determine q and then further subsample each data source by $[0.1, 0.3, 0.7, 1.0]$ to vary n. Test errors are measured by whether the execution of the model matches human references.

We fit $V(n, q)$ with 5 terms for $C(q)$ on 10 models containing less than 16,000 examples, and testing on 19 models containing between 16,000 and 100,000 examples. The results in Table 2 show our approach is accurate ($r^2 = 0.89$) and matches the best baseline ($r^2 = 0.90$) under a bootstrapped paired difference test. Both methods outperform other baselines including data size ($r^2 = 0.64$). We see more substantial gaps between the best MLP model (small) and the rational function approximation here, with the MLP model performing worse ($r^2 = 0.35$) due to overaggressive extrapolation. Analyzing these results together with the

	Task-oriented dialogue		Machine Translation		Multitask QA	
Method	Train	Extrapolation	Train	Extrapolation	Train	Extrapolation
Datasize	0.54	0.64 (0.41, 0.78)	0.07	-0.80 (-3.48, -0.09)	0.49	0.38 (-0.19, 0.58)
Linear	0.99	**0.90** (0.73, 0.95)	0.30	-0.69 (-3.02, -0.05)	0.87	-1.5 (-10.0, -0.29)
MLP	0.99	0.35 (-0.54, 0.61)	0.91	-0.95 (-3.49, -0.26)	0.99	-0.17 (-2.9, 0.20)
Rational	0.99	**0.89** (0.72, 0.94)	0.97	**0.83** (0.57, 0.92)	0.96	**0.85** (0.43, 0.92)

Table 2. Accuracy of error estimates on 3 real-world tasks that pose challenges for performance prediction due to their use of deep neural networks, non-separable losses such as BLEU, and weak supervision. Bolded method indicates best method with at a 5% significance level on bootstrapped paired differences. For MLP, we report the best of 3 models for brevity.



(a) Data size (b) Linear (c) MLP (overparam) (d) Rational function

Figure 2. Performance prediction on a multi-domain machine translation task with BLEU as the performance measure. There is little correlation between dataset size and loss (left panel) while the rational function approximation provides reasonable predictions (right).

amazon experiment suggests that the linearity assumption may be substantially violated in certain situations (amazon sentiment) even though it may perform well in some situations (dialogue).

Machine translation Thus far, we have evaluated on separable losses such as mean squared error, or model accuracy. We now show that our approach to predicting model performance continues to work for non-separable losses such as BLEU for machine translation. Our task is the standard multi-domain machine translation dataset from Koehn & Knowles (2017). We use the preprocessed data, model, and hyperparameters from Hu et al. (2019) which was the state-of-the-art domain adaptation based translation method for this dataset in 2019. The model is trained on 4 data sources: Acquis (legal text), EMEA (parliamentary proceedings), IT (IT assistance), and Koran (translations of the Quran). Evaluation is performed on the Acquis test set using sacrebleu to compute BLEU (Post, 2018).

To estimate the performance of models under varying data composition, we subsample up to 300,000 sentences from each data source, fit the estimators on 19 datasets of size less than 600,000 total sentences, and evaluate on 11 datasets of size 600,000 to 1,200,000. Since BLEU is a similarity measure and is penalized by reference ambiguity, we consider 50-BLEU to be the excess error. The rational function approximation was the only procedure to achieve a positive r^2 (0.83) among the methods. The difference in prediction accuracies is apparent when plotting predicted and observed log-loss (Figure 2). Especially notable is the fact that the overparametrized MLP model fits well on the training set, but completely fails to extrapolate. In contrast, the linear model has a low *training* set r^2, suggesting that the relationship between data composition and performance is fundamentally nonlinear in this case.

Multitask question answering Finally, we consider a multitask learning problem where some of the data sources are auxiliary tasks that may not directly be useful for the test time task. This breaks the covariate shift assumption that has been implicit throughout this paper. The target task is the BoolQ question answering dataset , and we train this model using a combination of 4 data sources: the MNLI entailment task (Williams et al. (2018), 50,000 examples subsampled), STS sentence similarity judgment task (Cer et al. (2017), 5749 examples), MRPC paraphrasing task (Dolan et al. (2004), 3668 examples), and the BoolQ training set (Clark et al. (2019) 9427 examples). We use the GLUE data with the Jiant package to train a multitask BERT based model for this task (Wang et al., 2019b). The Jiant package captures best-practices on the GLUE benchmark and is representative of how a large class of pre-trained model based classifiers operate.

The challenge with this task is that only the BoolQ training set provides direct supervision for the test-time task, and the other data sources provide weak supervision that may or may not be helpful in the downstream problem. The model performance estimates are fitted on 9 datasets with up to 26,000 total examples and evaluated on 14 datasets with more than 26,000 examples. The linear and MLP estimates do not seem to extrapolate well to the test set, and the datasize estimates provides only weak correlation

to the true errors. Although the gaps between the various methods are large, we note that the small sample size makes the confidence intervals relatively wide compared to the other experiments, and the 5 and 95% percentiles for the bootstrapped paired difference between the rational and datasize methods are (0.04, 0.80) respectively.

Overall, the rational function approximation is a promising approximation for model performance scaling even in challenging scenarios with neural models, non-accuracy metrics (such as BLEU), and multitask settings. Although linear approximations to generalization error can be effective (as seen in the dialogue task), this was not the case for the remaining three out of four tasks and the rational function approximation obtained substantial gains in those settings.

5. Discussion

In this work, we've proposed a new approach to predicting the performance of a prediction model as a function of training data composition that consists of measuring model accuracies for small n and a range of q and fitting a parametric model $V(n, q) := -\alpha \log(n) + \sum_{i=1}^{m} (\sum_{k=1}^{K} \lambda_{ik} q_k)^{-1}$. Our main contribution is to show that for a range of models, this nonlinear parametric model is a more natural approximation to generalization error than existing linear approximations. Empirical results on the Amazon sentiment regression task show that this approximation is accurate under a range of conditions, and experiments on neural models suggest that the method can continue to perform well in more realistic situations where the theory does not necessarily hold. Our work is a first step in going beyond closed-form estimates of model performance or additivity assumptions. It is an open question whether the same approach can scale to more extreme extrapolation settings or large numbers of data sources, and we hope to explore this in future work.

6. Acknowledgments

The authors would like to acknowledge helpful comments from the reviewers, as well as comments on an earlier draft of this work from researchers at Microsoft Semantic Machines, including Christopher Lin, Jayant Krishnamurty, Adam Pauls, and Steven Wegmann.

References

Agarwal, A., Dahleh, M., and Sarkar, T. A Marketplace for Data: An Algorithmic Solution. *arXiv preprint arXiv:1805.08125*, 2019.

Andreas, J., Bufe, J., Burkett, D., Chen, C., Clausman, J., Crawford, J., Crim, K., DeLoach, J., Dorner, L., Eisner, J., Fang, H., Guo, A., Hall, D., Hayes, K., Hill, K., Ho, D., Iwaszuk, W., Jha, S., Klein, D., Krishnamurthy, J., Lanman, T., Liang, P., Lin, C., Lintsbakh, I., McGovern, A., Nisnevich, A., Pauls, A., Petters, D., Read, B., Roth, D., Roy, S., Rusak, J., Short, B., Slomin, D., Snyder, B., Striplin, S., Su, Y., Tellman, Z., Thomson, S., Vorobev, A., Witoszko, I., Wolfe, J., Wray, A., Zhang, Y., and Zotov, A. Task-oriented dialogue as dataflow synthesis. *Transactions of the Association for Computational Linguistics (TACL)*, 8:556–571, 2020.

Belkhir, N., Dréo, J., Savéant, P., and Schoenauer, M. Per instance algorithm configuration of cma-es with limited budget. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '17, pp. 681–688, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349208. doi: 10.1145/3071178.3071343. URL https://doi.org/10.1145/3071178.3071343.

Ben-David, S., Lu, T., Luu, T., and Pal, D. Impossibility theorems for domain adaptation. In *Artificial Intelligence and Statistics (AISTATS)*, 2010.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. *International Workshop on Semantic Evaluation (SemEval)*, 2017.

Clark, C., Lee, K., Chang, M., Kwiatkowski, T., Collins, M., and Toutanova, K. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *North American Association for Computational Linguistics (NAACL)* , 2019.

Crammer, K., Kearns, M., and Wortman, J. Learning from multiple sources. In *Advances in Neural Information Processing Systems*, 2007.

Dolan, B., Quirk, C., and Brockett, C. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *International Conference on Computational Linguistics (COLING)*, 2004.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. In *Conference on Learning Theory (COLT)*, 2010.

Ghorbani, A. and Zou, J. Data shapley: Equitable valuation of data for machine learning. *arXiv preprint arXiv:1904.02868*, 2019.

Ghorbani, A., Kim, M., and Zou, J. A Distributional Framework for Data Valuation. *arXiv preprint arXiv:2002.12334*, 2020.

Hanneke, S. A bound on the label complexity of agnostic active learning. In *International Conference on Machine Learning (ICML)*, pp. 353–360, 2007.

Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., Patwary, M., Yang, Y., and Zhou, Y. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Hu, J., Xia, M., Neubig, G., and Carbonell, J. Domain adaptation of neural machine translation by lexicon induction. In *Association for Computational Linguistics (ACL)*, pp. 2989–3001, 2019.

Jia, R., Dao, D., Wang, B., Hubis, F. A., Hynes, N., Gurel, N. M., Li, B., Zhang, C., Song, D., and Spanos, C. Towards efficient data valuation based on the shapley value. *arXiv preprint arXiv:1902.10275*, 2019.

John, R. C. and Draper, N. R. D-optimality for regression designs: A review. *Technometrics*, 17(1):15–23, 1975.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Koehn, P. and Knowles, R. Six challenges for neural machine translation. In *The First Workshop on Neural Machine Translation*, pp. 28–39. Association for Computational Linguistics, 2017.

Kolachina, P., Cancedda, N., Dymetman, M., and Venkatapathy, S. Prediction of learning curves in machine translation. In *Annual Meeting of the Association for Computational Linguistics*, pp. 22–30, Jeju Island, Korea, 7 2012. Association for Computational Linguistics.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation with multiple sources. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1041–1048, 2009.

Ohrimenko, O., Tople, S., and Tschiatschek, S. Collaborative Machine Learning Markets with Data-Replication-Robust Payments. *arXiv preprint arXiv:1911.09052*, 2019.

Post, M. A call for clarity in reporting BLEU scores. In *The Third Conference on Machine Translation: Research Papers*, pp. 186–191, Belgium, Brussels, 2018. Association for Computational Linguistics.

Pukelsheim, F. *Optimal Design of Experiments*. Society for Industrial and Applied Mathematics, USA, 2006. ISBN 0898716047.

Rosenfeld, J., Rosenfeld, A., and Belinkov, Y. A constructive prediction of the generalization error across scales. In *International Conference on Learning Representations (ICLR)*, 2020.

van der Vaart, A. W. *Asymptotic statistics*. Cambridge University Press, 1998.

Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv*, 2010.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019a.

Wang, A., Tenney, I. F., Pruksachatkun, Y., Yeres, P., Phang, J., Liu, H., Htut, P., , Yu, K., Hula, J., Xia, P., Pappagari, R., Jin, S., McCoy, R., Patel, R., Huang, Y., Grave, E., Kim, N., Fevry, T., Chen, B., Nangia, N., Mohananey, A., Kann, K., Bordia, S., Patry, N., Benton, D., Pavlick, E., and Bowman, S. `jiant` 1.3: A software toolkit for research on general-purpose text understanding models. `http://jiant.info/`, 2019b.

Williams, A., Nangia, N., and Bowman, S. A broad-coverage challenge corpus for sentence understanding through inference. In *Association for Computational Linguistics (ACL)*, pp. 1112–1122, 2018.

Yoon, J., Arik, S., and Pfister, T. Data valuation using reinforcement learning. *arXiv preprint arXiv:1909.11671*, 2019.