Invariant representation learning for robust deep networks

Julian Salazar* Amazon AI
Davis Liang* Amazon AI
Zhiheng Huang Amazon AI
Zachary C. Lipton Amazon AI, CMU

Abstract

Deep neural networks are often brittle to superficial perturbations of their inputs; models that perform well offline on held-out data can still break under small amounts of naturally-occurring or adversarial shifts. We consider *invariant representation learning* (IRL), first proposed in the domain of speech recognition, as a simple, effective, and general extension to data augmentation. Rather than only presenting original and noisy inputs as having the same label, IRL also promotes similar intermediate representations for original examples and their noised counterparts. The approach penalizes the distance (typically L^2 and cosine distances) between their activations, at every layer above a chosen bottleneck. We motivate IRL from vicinal risk motivation and existing regularizers, formulate IRL for image classification, language modeling, speech recognition, and semi-supervised learning, and experimentally show improvements on these tasks in terms of accuracy and in robustness to synthetic, out-of-domain, and adversarial noise.

1 Introduction

In supervised learning, one has prior knowledge that a model's outputs should be invariant to minor changes in the inputs. Colloquially, we refer to this invariance as *robustness* and these minor changes as *noise*. During training, one can introduce this knowledge stochastically via *data augmentation* [1], or analytically via a related *regularizer* [2, 3, 4]. Deep learning provides more approaches: invariances can be promoted via *neural network design*, e.g., convolutional layers [5], or via effective *representation learning*, e.g., deep embeddings which cluster together transform-related inputs, like words with similar contexts [6].

In this paper, we show that *invariant representation learning* (IRL), introduced in [7] for noise robustness in sequence-to-sequence speech recognition, can be reinterpreted as a broad, effective synthesis of these methods for learning robust representations. For supervised learning, let $\mathcal{L}(\boldsymbol{x}, y; \boldsymbol{\theta})$ denote the per-sample loss on $\boldsymbol{x}$ given its training label y and model parameters $\boldsymbol{\theta}$. Then, replace $\mathcal{L}$ with:

$$\mathcal{L}_{\text{IRL}}(\boldsymbol{x}, y) = \alpha\mathcal{L}(\boldsymbol{x}, y) + \beta\mathcal{L}_{\text{noise}}(\boldsymbol{x}, y) + \gamma\mathcal{L}_{\text{dist}}(\boldsymbol{x}), \tag{1}$$

where α, β, γ are hyperparameters, and with $\mathcal{L}_{\text{noise}}, \mathcal{L}_{\text{dist}}$ (regularization) computed stochastically: Let $\nu_{\boldsymbol{x}}$ be our noise distribution model around $\boldsymbol{x}$. We sample K noisy versions $\tilde{\boldsymbol{x}}^{(1)}, \ldots, \tilde{\boldsymbol{x}}^{(K)}$ from $\nu_{\boldsymbol{x}}$ (data augmentation) then compute

$$\mathcal{L}_{\text{noise}}(\boldsymbol{x}, y) = \tfrac{1}{K}\textstyle\sum_{k=1}^{K}\mathcal{L}(\tilde{\boldsymbol{x}}^{(k)}, y), \quad \mathcal{L}_{\text{dist}}(\boldsymbol{x}) = \tfrac{1}{K}\textstyle\sum_{k=1}^{K}\sum_{\ell=1}^{L}\gamma_\ell\, d(\boldsymbol{a}_\ell, \tilde{\boldsymbol{a}}_\ell^{(k)}). \tag{2}$$

where $\boldsymbol{a}_\ell$ gives the model's activations at layer ℓ for the provided input, and γ_ℓ are hyperparameters that are non-zero at layers ℓ one expects the original and noisy samples to have similar representations at (network design). Hence, IRL expresses the inductive bias that a deep network's intermediate representations should also exhibit invariance to minor changes in the input. Our description of IRL is kept general to facilitate comparison with existing approaches. Most readily, $(\alpha, \beta, \gamma) = (1, 0, 0),\ (0, 1, 0)$ give regular training and data augmentation, respectively. Formulations can be very simple: IRL-E [7] supervises the encoded representations of an original and a noised speech sample; explicitly, $\alpha = \beta = \gamma = K = 1$, with $\gamma_\ell = 0$ for all but the last encoder layer $\ell = e$. There (and here), d is a weighted sum of squared L^2 distance and negative cosine similarity $d(\boldsymbol{u}, \boldsymbol{v}) = ||\boldsymbol{u} - \boldsymbol{v}||^2 - \lambda\frac{\boldsymbol{u}^\top\boldsymbol{v}}{||\boldsymbol{u}||||\boldsymbol{v}||}$, as this both reduces magnitudes and angles on the penalized layer. To ensure this persists, one can also penalize losses in all subsequent layers (IRL-C).

Related work: While IRL's ideas have been studied independently, beyond [7] we find none composing them into a single IRL-like procedure. While [7] compared IRL with other noise robustness strategies in speech, we view IRL

*Equal contribution. Presented at the workshop: *Integration of Deep Learning Theories*
32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

as a domain-independent regularizer and compare with thematically-related work. ***Joint representation invariance:*** Multi-task learning learns task-independent features implicitly, though one can further regularize the feature selection matrix [8]. In deep learning, [9, 10] used branch networks (classifier, autoencoder respectively); IRL does not require a branch or two steps. ***Pairing originals and noise:*** Denoising autoencoders [11] take inputs and corruptions to learn reconstruction. Per-sample pairing with original inputs for discriminative tasks was done with synthetic noise [12] and adversarial noise [13, 14, 15]. These consider $(\alpha, \beta, \gamma) = (1, 0, \gamma)$, $(\frac{1}{2}, \frac{1}{2}, 0)$, $(\frac{1}{2}, \frac{1}{2}, \gamma)$, $(1, 0, 0)$, respectively. All take $K = 1$, none consider sequence tasks or intermediate representations, and ones with $\gamma \neq 0$ only applied squared L^2 distance on the logits ($\ell = L$). ***Activations across layers:*** In the generative context, [16] supervised content and style losses (L^2 distances) between source images $\boldsymbol{x}_{\text{content}}, \boldsymbol{x}_{\text{style}}$ to improve a generated image $\boldsymbol{x}'$.

Our contributions: We extend IRL to general deep networks, motivate IRL from vicinal risk minimization, and show that IRL specializes to a number of known stochastic and analytic regularization procedures. We formulate IRL for convolutional architectures via image classification, for natural language via word-level language modeling, and extend [7] for speech recognition to the Wall Street Journal (WSJ) dataset. We observe consistent improvements over naive data augmentation for clean, noisy, and adversarial versions of the test sets. We also formulate IRL for semi-supervised learning, validating this strategy on a standard semi-supervised CIFAR setup and observing a 1-3% absolute improvement in accuracy. Finally, we show that presenting paired original and noised samples together is itself an effective regularizer, and that per-layer activation distances, along with the norms of Jacobians, are small with IRL.

2 Interpretations

In machine learning we hope to minimize *risk* $R(\boldsymbol{\theta}) = \int L(\mathrm{x}, \mathrm{y}; \boldsymbol{\theta}) \, \mathrm{d}P_{\text{data}}(\mathrm{x}, \mathrm{y})$, where L is the loss function and P_{data} is the data-generating distribution. *Empirical risk minimization* (ERM) [17] approximates P_{data} with $\hat{P}_{\text{data}} = \frac{1}{N} \sum_{i=1}^{N} (\delta_{\boldsymbol{x}^{(i)}} \times \delta_{y^{(i)}})$ where $\delta_{\boldsymbol{v}}$ is the Dirac measure at $\boldsymbol{v}$. Deep learning further substitutes L with surrogate losses $\mathcal{L}$ like cross entropy, while memory constraints lead to minibatching $\{(\boldsymbol{x}^{(b)}, y^{(b)})\}_{b=1}^{B}$ [18]. A typical deep learning optimization step can be seen as taking successive estimates of risk:

$$R(\boldsymbol{\theta}) \approx \int L(\mathrm{x}, \mathrm{y}; \boldsymbol{\theta}) \, \mathrm{d}\hat{P}_{\text{data}}(\mathrm{x}, \mathrm{y}) \approx \int \mathcal{L}(\mathrm{x}, \mathrm{y}; \boldsymbol{\theta}) \, \mathrm{d}\hat{P}_{\text{data}}(\mathrm{x}, \mathrm{y}) \approx \tfrac{1}{B} \textstyle\sum_{b=1}^{B} \mathcal{L}(\boldsymbol{x}^{(b)}, y^{(b)}; \boldsymbol{\theta}). \tag{3}$$

The confidence of empirical risk as an estimate is affected by a model family's capacity [17]. Memorization of examples, as observed in deep networks [19], minimizes empirical risk but generalizes poorly; hence we consider improvements:

Vicinal risk: One can take a generative viewpoint with *vicinal risk minimization* (VRM) [20], where $\mathrm{d}\hat{P}_{\text{data}}$ is interpreted as a density estimate, and use the noise model $\nu_{\boldsymbol{x}^{(i)}}$ to naturally incorporate our prior knowledge of invariances into VRM: $P_{\text{data}} \approx \hat{P}_{\text{est.}} = \frac{1}{N} \sum_{i=1}^{N} (\nu_{\boldsymbol{x}^{(i)}} \times \delta_{y^{(i)}})$. In practice, $\nu_{\boldsymbol{x}^{(i)}}$ might be chosen for coverage rather than accuracy, so one can take the mixture $\alpha\delta_{\boldsymbol{x}^{(i)}} + \beta\nu_{\boldsymbol{x}^{(i)}}$ instead. Working proportionally, we can write $\frac{\alpha}{\beta}\delta_{\boldsymbol{x}^{(i)}} + \nu_{\boldsymbol{x}^{(i)}}$ and view $\frac{\alpha}{\beta}$ as a pseudocount. During training, one approximates $\nu_{\boldsymbol{x}^{(i)}}$ by sampling noised versions (stochastic data augmentation). We present the original sample $\boldsymbol{x}$ and noised samples $\{\tilde{\boldsymbol{x}}^{(k)}\}_{k=1}^{K}$ together (scaled by α, β). We call this *IRL batching*, reducing the variance of the per-batch risk estimate by taking, for each sample $\boldsymbol{x}$,

$$\int \mathcal{L}(\mathrm{x}, \mathrm{y}; \boldsymbol{\theta}) \, \mathrm{d}[\alpha\delta_{\boldsymbol{x}} + \beta\nu_{\boldsymbol{x}}](\mathrm{x}) \, \mathrm{d}\delta_y(\mathrm{y}) \approx \alpha\mathcal{L}(\boldsymbol{x}, y; \boldsymbol{\theta}) + \tfrac{\beta}{K} \textstyle\sum_{k=1}^{K} \mathcal{L}(\tilde{\boldsymbol{x}}^{(k)}, y; \boldsymbol{\theta}), \tag{4}$$

which are exactly $\mathcal{L}$ and $\mathcal{L}_{\text{noise}}$ in Equation 1. We see that $K \to \infty$ allows for further variance reduction in estimated risk by the central limit theorem. However, this may hurt generalization as $\mathcal{L}_{\text{noise}}$ could stabilize while individual $\mathcal{L}(\tilde{\boldsymbol{x}}^{(k)}, y; \boldsymbol{\theta})$ can still be improved. The high capacity of neural networks motivates stochastic noising (versus pre-creating noisy versions) to avoid memorization when # of parameters $P \gg (K + 1)N$. However, in practice this can still result in memorizing the most recent $\mathcal{O}(P)$ inputs, a phenomenon known as *catastrophic forgetting* [21]. One idea is to balance the empirical risk of the latest batch with the risk of the model learned thus far. Specifically, let $f(\boldsymbol{x}; \boldsymbol{\theta})$ play the role of y and interpolate Equation 4 with a weighted term

$$\int (\cdots + \gamma\mathcal{L}(\mathrm{x}, f(\boldsymbol{x}; \boldsymbol{\theta}); \boldsymbol{\theta})) \, \mathrm{d}\nu_{\boldsymbol{x}}(\mathrm{x}) \, \mathrm{d}\delta_y(\mathrm{y}) \approx \cdots + \tfrac{\gamma}{K} \textstyle\sum_{k=1}^{K} \mathcal{L}_{\text{c.e.}}(f(\tilde{\boldsymbol{x}}^{(k)}; \boldsymbol{\theta}), f(\boldsymbol{x}; \boldsymbol{\theta})), \tag{5}$$

for cross-entropy loss, sharing the noise draws from Equation 4. This latter term corresponds to $\mathcal{L}_{\text{dist}}$, except IRL uses symmetric loss functions over multiple layers to enforce our inductive bias of invariant intermediate representations.

Structural risk: Another way to improve Equation 3 is to balance $\hat{P}_{\text{data}}$ with model complexity via *structural risk minimization* [17]. One approach modifies the objective $\mathcal{L}$ with a regularization term dependent on $\boldsymbol{\theta}$. [3] used an additive noise model $\tilde{\mathbf{x}} = \boldsymbol{x} + \boldsymbol{\xi}$, where $\boldsymbol{\xi}$ is independent of $\boldsymbol{x}$, to derive a regularizer for a neural network $f(\boldsymbol{x}; \boldsymbol{\theta})$. They take a component-wise Taylor expansion of $f = (f_1, \ldots, f_J)$ around $\boldsymbol{x}$, which gives $f_j(\tilde{\mathbf{x}}) = f_j(\boldsymbol{x} + \boldsymbol{\xi}) = f_j(\boldsymbol{x}) + [\nabla f_j(\boldsymbol{x})]^\top \boldsymbol{\xi} + \frac{1}{2}\boldsymbol{\xi}^\top [\nabla^2 f_j(\boldsymbol{x})]\boldsymbol{\xi} + O(||\boldsymbol{\xi}||^3)$. When $\mathcal{L}$ is mean squared error and under further assumptions

on $\boldsymbol{\xi}$ (e.g., zero mean, isotropic covariance $\eta^2 \boldsymbol{I}$), one can marginalize out $\boldsymbol{\xi}$ and retrieve (to $\mathcal{O}(\eta^2)$) the Tikhonov regularizer $\eta^2 ||\frac{\partial f}{\partial \boldsymbol{x}}(\boldsymbol{x}; \boldsymbol{\theta})||^2$. This argument has been expanded by others [22, 23, 15]. One can show that under the additive noise model, with d as pairwise L^2 distance, $\mathbb{E}[\boldsymbol{\xi}] = 0$, and $\mathcal{L} = \mathcal{L}_{\text{c.e.}}$ (cross-entropy), that $\mathcal{L}_{\text{IRL}}$ induces in expectation the analytic regularizer:

$$\mathcal{L}_{\text{IRL}}(\boldsymbol{x}, y) \approx (\alpha + \beta)\mathcal{L}_{\text{c.e.}}(\boldsymbol{x}, y) + \beta\Omega_R + \gamma\Lambda_R, \tag{6}$$

$$\Omega_R = -\tfrac{1}{2} \sum_{j=1}^{J_L} y_j (\nabla_{\boldsymbol{x}} \log f_j(\boldsymbol{x}))^\top \boldsymbol{\Sigma} (\nabla_{\boldsymbol{x}} \log f_j(\boldsymbol{x})), \quad \Lambda_R = \sum_{\ell=1}^{L} \gamma_\ell \sum_{j=1}^{J_\ell} (\nabla_{\boldsymbol{x}} (a_\ell)_j(\boldsymbol{x}))^\top \boldsymbol{\Sigma} (\nabla_{\boldsymbol{x}} (a_\ell)_j(\boldsymbol{x})), \tag{7}$$

where $\boldsymbol{\Sigma} = \text{Cov}(\boldsymbol{\xi})$ and $\boldsymbol{a}_\ell$ are ℓ-th layer activations as before. One can estimate $\boldsymbol{\Sigma}$ as isotropic [3], diagonal [22], or with a running estimate if the noise distribution evolves [15] as observed with adversarial examples. In fact, by furthermore removing $\gamma\Lambda_R$ (corresponding to $\mathcal{L}_{\text{dist}}$) and dividing by $\alpha+\beta$, we retrieve the analytic regularizers validated in these respective works as performing well against synthetic and adversarial noise.

3 Experiments

Image classification: We trained Wide ResNet-28-10 [24] on CIFAR-10/100 [25]. We view their augmentation of random crops, flips, and mean-variance normalization as global invariances (instead of minor 'noise') to be applied consistently to the original sample and its noised counterparts. To ablate the effects of IRL, we define the following noise distributions $\nu_{\boldsymbol{x}}$. ***In-domain (in.):*** We apply brightness and contrast jitters (scales[2] are uniform from $[-0.5, 0.5]$), then add PCA noise with a scale of 0.1 [26]. ***Out-of-domain (out.):*** We apply hue and saturation jitters (scales are uniform from $[-0.5, 0.5]$). ***Adversarial (adv.):*** We dynamically create white-box adversarial examples on the normalized images using the fast gradient sign method (FGSM) [13], choosing ϵ s.t. 2ϵ is near where the baseline performs close to random.

Here and in later experiments, we use $\alpha = \beta = 0.5$ and $K = 1$ as validated on a 40K-10K split of CIFAR-10's train set using ResNet-20-v2 [27]; our search was consistent with Section 2's proposal to use a mixed distribution and the reduced generalization as $K \to \infty$. Our results are listed in Table 1. For IRL we take $\gamma_\ell = 1$ on the last 4 blocks and the logits. We realize a 5-8% relative improvement over the baseline using the IRL procedure with in-domain noise, realized incrementally through data augmentation, IRL batching (Section 2), and IRL loss. We also saw similar improvement with adversarial examples; in particular, we took $(\alpha, \beta) = (0.8, 0.2)$ after observing the expected overfitting on the adversarial distribution (e.g., adversarial training [13] performed better on 0.5ϵ than on the original test set).

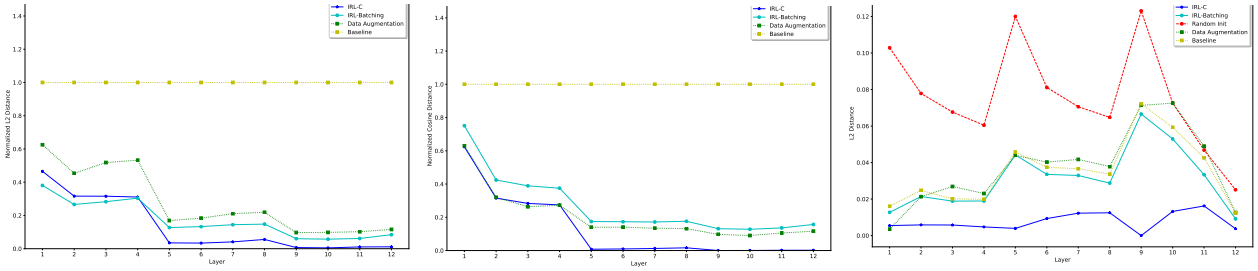
Table 1: Test error of Wide ResNet-28-10 on CIFAR datasets. Baselines from [24].

			CIFAR-10						**CIFAR-100**				
(α, β, γ)	**Method**	**Noise**	none	in.	out.	ϵ=0.2	ϵ=0.8	**Noise**	none	in.	out.	ϵ=0.1	ϵ=0.4
$(1, 0, 0)$	Baseline	std.	3.89	–	–	–	–		18.85	–	–	–	–
$(1, 0, 0)$	(ours)	std.	3.73	5.43	9.24	40.62	82.22		18.77	23.23	39.70	55.87	96.96
$(0, 1, 0)$	Data aug.	in.	3.79	4.67	9.48	42.58	86.19		18.68	20.33	38.69	57.57	96.88
$(0.5, 0.5, 0)$	IRL bat.	in.	3.66	4.55	9.50	42.94	86.98		18.30	20.18	**38.54**	57.60	97.57
$(0.5, 0.5, 1)$	IRL loss	in.	**3.60**	**4.46**	**8.90**	48.43	87.72		**17.64**	**19.78**	38.89	57.70	97.65
$(0.5, 0.5, 0)$	Adv. trn.	ϵ=0.4	4.92	7.74	9.42	4.79	69.51	ϵ=0.2	23.83	31.48	47.51	20.90	83.46
$(0.8, 0.2, 0)$	IRL bat.	ϵ=0.4	5.14	8.70	11.45	5.04	71.61	ϵ=0.2	20.52	25.60	39.85	21.06	**80.40**
$(0.8, 0.2, 1)$	IRL loss	ϵ=0.4	4.55	7.81	10.32	**4.63**	**69.47**	ϵ=0.2	22.49	28.84	45.20	**20.26**	88.63

Activation and gradient magnitudes: For our CIFAR-10 models we plot L^2 and cosine distances between the hidden layer activations on the original and in-domain noised test set in Figure 1. For both metrics regular data augmentation already induces hidden representations close in distance, which is further improved by IRL. Furthermore, in Figure 1c we observe Jacobians $\frac{\partial a_\ell}{\partial \boldsymbol{x}}(\boldsymbol{x})$ with smaller norm through IRL, validating our analytic analysis of IRL as a regularizer on component-wise activation gradients (wrt. inputs) across layers (Equation 7). This is also suggestive of smoothness in each layer's intermediate landscape, as opposed to the fitting of noise warned against in Section 2.

Semi-supervision: Leaving unlabeled data unused is wasteful, as one often has prior distributional knowledge that can be leveraged [28]. Here, one has a noise model $\nu_{\boldsymbol{x}}$ that real inputs should satisfy. In the VRM framework, [20] noted that for unlabeled $\boldsymbol{x}$, one can replace the unknown 'true' y with the current best estimate, $f(\boldsymbol{x}; \boldsymbol{\theta})$, to give $\int \mathcal{L}(\mathbf{x}, f(\boldsymbol{x}; \boldsymbol{\theta}); \boldsymbol{\theta})\, \mathrm{d}\nu_{\boldsymbol{x}}(\mathbf{x})$. We saw this "model risk" in Equation (5), which we corresponded to $\mathcal{L}_{\text{dist}}$. This motivates *semi-supervised IRL*, where for unlabeled data, one applies noising and computes only $\mathcal{L}_{\text{dist}}$ (i.e., $(\alpha, \beta, \gamma) = (0, 0, \gamma)$. This selective use of loss terms is similar to [10], where for unlabeled data they restrict to the autoencoder path in their joint-training network. We use a standard semi-supervised "reduced CIFAR" setup [29] (take 4,000 labeled samples

[2] `https://mxnet.incubator.apache.org/tutorials/python/types_of_data_augmentation.html`



(a) Normalized average L^2 distance (b) Normalized average cosine distance (c) Average Frobenius norm of $\frac{\partial a_\ell}{\partial \boldsymbol{x}}(\boldsymbol{x})$

Figure 1: Average (a) L^2 and (b) cosine distance between original and noised test data (normalized to baseline), and (c) the Jacobian norm per layer on original data, versus models (series) and across layers (x-axis) on CIFAR-10 models.

and discard the labels on the other 46,000). We get 21.5% and 60.7% error by applying IRL on the 4K samples for CIFAR-10/100. Semi-supervision gives 18.2% and 59.7% respectively, an absolute accuracy improvement of 1-3%.

Language modeling: We consider word-level, recurrent language models [30], training standard LSTM models on Penn Treebank (PTB) and WikiText-2 [31]. We selected this task because neural language models can be architecturally simple (e.g., two LSTM layers) and yet prone to overfitting [32]; also, data augmentation for neural language models is an underserved topic of study [33]. Let $w_{<t}$ denote the preceding context for a word w_t during model training. [33] considered these two noise types. ***Unigram:*** For each word in $w_{<t}$, with probability p we replace the word with a draw from the unigram distribution. ***Blank:*** Instead, replace with a placeholder token "__".

Table 2: Test perplexities of an LSTM model on PTB and WikiText-2. Baselines from [33].

			PTB		**WikiText-2**	
(α, β, γ)	**Method**	**Noise**	val.	test	val.	test
$(1, 0, 0)$	Baseline	none	81.6	77.5	–	–
$(1, 0, 0)$	(ours)	none	80.5	77.5	96.9	92.1
$(0, 1, 0)$	Data aug.	uni.	85.8	82.9	106.7	100.0
$(0.5, 0.5, 0)$	IRL bat.	uni.	80.2	77.4	97.5	92.6
$(0.5, 0.5, 1)$	IRL loss	uni.	77.2	74.4	99.2	93.7
$(0, 1, 0)$	Data aug.	blank	78.8	75.3	98.6	92.3
$(0.5, 0.5, 0)$	IRL bat.	blank	80.3	76.7	97.7	93.8
$(0.5, 0.5, 1)$	IRL loss	blank	**75.1**	**71.8**	**94.0**	**88.6**

Here, we take $p = 0.2$ and a two-layer, 1500 hidden-unit LSTM model with a final dense layer, our model and training regime matching [33]. We supervise on the logits and the hidden states before ($\gamma_{L-1} = \gamma_L = 1$) with L^2 distance only. The results are in Table 2. We saw cumulative improvement due to data augmentation, IRL batching, and IRL loss. IRL is particularly essential for these models, as augmentation and IRL batching often gave no improvement over baseline, with jumps in improvement only occurring for IRL loss. With blank, we outperform variational dropout and its Monte Carlo variant (71.8 vs. 75.0, 73.4) on PTB, and variational dropout on WikiText-2 (88.6 vs 96.3) [34].

Table 3: Test character error rates (CER) on WSJ and LibriSpeech under noise. LibriSpeech for a 4+4-layer model [7].

		WSJ (eval92)						**LibriSpeech (test-clean)**					
(α, β, γ)	**Method**	none	RIR	speech	v.up	v.down	tel.	none	RIR	speech	v.up	v.down	tel.
$(1, 0, 0)$	Regular	**11.2**	67.5	130.0	33.8	**28.0**	70.0	6.5	24.1	91.5	6.5	6.5	14.2
$(1, 1, 0)$	IRL bat.	13.0	**45.9**	83.5	29.7	29.4	**64.7**	6.4	21.0	32.0	6.4	6.3	12.2
$(1, 1, 1)$	IRL loss	12.3	48.2	**58.3**	**29.6**	30.0	73.0	**3.3**	**13.8**	**14.1**	**3.5**	**3.5**	**6.4**

Speech recognition: We applied the experiment of [7] to the Wall Street Journal (WSJ) dataset [35]. Our added noises come from the MUSAN dataset [36]. Our out-of-domain noise are room impulse responses (RIR), overlapping speech drawn from the WSJ dataset at 6 SNRdB (speech), volume modulation by doubling/halving amplitude (vol. up/vol. down), and telephony resampling to 8 kHz (tel.) from 16 kHz, all as in [7]. Here, we take a 4-layer encoder, 4-layer decoder LSTM model with a final dense layer. We supervise on the encoder, four decoding layers and logits ($\gamma_\ell = 1$ for $\ell \geq e$). The results for WSJ are presented in Table 3 and contrasted to past LibriSpeech results. We found that IRL was more effective when trained on the LibriSpeech dataset than on the WSJ dataset for the same parameters; however, we still saw great improvement against overlapping speech with an absolute improvement of 25.2%.

References

[1] Henry S. Baird. Document image defect models. In *Structured Document Image Analysis*, pages 546–556. Springer, 1992.

[2] Patrice Simard, Bernard Victorri, Yann LeCun, and John Denker. Tangent Prop - a formalism for specifying selected invariances in an adaptive network. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 895–903, 1992.

[3] Chris M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, 1995.

[4] Laurens Maaten, Minmin Chen, Stephen Tyree, and Kilian Weinberger. Learning with marginalized corrupted features. In *International Conference on Machine Learning (ICML)*, pages 410–418, 2013.

[5] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[6] 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 (NeurIPS)*, pages 3111–3119, 2013.

[7] Davis Liang, Zhiheng Huang, and Zachary C. Lipton. Learning noise-invariant representations for robust speech recognition. In *Proceedings of IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018.

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

[9] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. *International Conference of Machine Learning (ICML)*, 2015.

[10] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision (ECCV)*, pages 597–613. Springer, 2016.

[11] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research (JMLR)*, 11(Dec):3371–3408, 2010.

[12] Stephan Zheng, Yang Song, Thomas Leung, and Ian Goodfellow. Improving the robustness of deep neural networks via stability training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4480–4488, 2016.

[13] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *International Conference of Learning Representations (ICLR)*, 2015.

[14] Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow. Adversarial logit pairing. *CoRR*, abs/1803.06373, 2018.

[15] Kevin Roth, Aurélien Lucchi, Sebastian Nowozin, and Thomas Hofmann. Adversarially robust training through structured gradient regularization. *CoRR*, abs/1805.08736, 2018.

[16] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2414–2423, 2016.

[17] Vladimir Vapnik. Principles of risk minimization for learning theory. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 831–838, 1992.

[18] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep Learning*, volume 1. MIT press, 2016.

[19] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *International Conference of Learning Representations (ICLR)*, 2017.

[20] Olivier Chapelle, Jason Weston, Léon Bottou, and Vladimir Vapnik. Vicinal risk minimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 416–422, 2001.

[21] Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

[22] Todd K. Leen. From data distributions to regularization in invariant learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 223–230, 1995.

[23] Salah Rifai, Xavier Glorot, Yoshua Bengio, and Pascal Vincent. Adding noise to the input of a model trained with a regularized objective. *CoRR*, abs/1104.3250, 2011.

[24] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *British Machine Vision Conference (BMVC)*, 2016.

[25] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, CiteSeerX, 2009.

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, pages 630–645. Springer, 2016.

[28] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *CoRR*, abs/1804.09170, 2018.

[29] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning augmentation policies from data. *CoRR*, abs/1805.09501, 2018.

[30] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Conference of the International Speech Communication Association*, 2010.

[31] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *CoRR*, abs/1609.07843, 2016.

[32] Stephen Merity, Nitish Shirish Keskar, and Richard Socher. Regularizing and optimizing LSTM language models. *International Conference of Learning Representations (ICLR)*, 2018.

[33] Ziang Xie, Sida I. Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y Ng. Data noising as smoothing in neural network language models. *International Conference of Learning Representations (ICLR)*, 2017.

[34] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1019–1027, 2016.

[35] John S. Garofolo et al. CSR-I (WSJ0) complete LDC93S6A. Philadelphia: Linguistic Data Consortium, 1993.

[36] David Snyder, Guoguo Chen, and Daniel Povey. MUSAN: A music, speech, and noise corpus. *CoRR*, abs/1510.08484, 2015.