IMCLR: IMPLICIT CONTRASTIVE LEARNING FOR IMAGE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Contrastive learning is an effective method for learning visual representations. In most cases, this involves adding an explicit loss function to encourage similar images to have similar representations, and different images to have different representations. In this paper, we introduce a clever construction for Implicit Contrastive Learning (ImCLR), primarily in the supervised setting: there, the network can implicitly learn to differentiate between similar and dissimilar images. Furthermore, this requires almost no change to existing pipelines, which allows for easy integration and for fair demonstration of effectiveness on a wide range of well-accepted benchmarks. Namely, there is no change to loss, no change to hyperparameters, and no change to general network architecture. We show that ImCLR improves the test error in the supervised setting across a variety of settings, including 3.24% on Tiny ImageNet, 1.30% on CIFAR-100, 0.14% on CIFAR-10, and 2.28% on STL-10. We show that this holds across different number of labeled samples, maintaining approximately a 2% gap in test accuracy down to using only 5% of the whole dataset. We further show that gains hold for robustness to common input corruptions and perturbations at varying severities with a 0.72% improvement on CIFAR-100-C, and in the semi-supervised setting with a 2.16% improvement with the standard benchmark Π-model. We demonstrate that ImCLR is complementary to existing data augmentation techniques, achieving over 1% improvement on CIFAR-100 by combining ImCLR with CutMix over either baseline, and 2% by combining ImCLR with AutoAugment over either baseline. Finally, we perform an extensive ablation study to better understand the proposed algorithm.

1 INTRODUCTION

In the last decade, numerous innovations in deep learning for computer vision have substantially improved results on many benchmark tasks (Krizhevsky et al., 2012; He et al., 2016; Zagoruyko & Komodakis, 2016; Huang et al., 2017). These innovations include architecture changes, training procedure improvements, data augmentation techniques, regularization strategies, among many others. While the supervised setting remains the gold-standard, pre-training and fine-tuning has emerged as a powerful paradigm. In recent years, major advancements in unsupervised representation learning (Chen et al., 2020b;c; He et al., 2019; Khosla et al., 2020) and semi-supervised learning (Chapelle & Scholkopf, 2006; Lee, 2013; Tarvainen & Valpola, 2017; Berthelot et al., 2019) have allowed neural networks to leverage vast amounts of unlabeled data.

Contrastive learning has shown to be one of the leading ideas in this regard (Becker & Hinton, 1992; Hadsell et al., 2006). Generally, contrastive learning encourages images with similar semantics to have similar representations, while images with dissimilar semantics to have dissimilar representations. This form of representation learning is most often implemented with an explicit loss function (Chen et al., 2020a;b;c; He et al., 2019; Khosla et al., 2020). However, this requires changes to existing training pipelines (Chen et al., 2020b;c; He et al., 2019; Khosla et al., 2020), such as using larger batches, the decision over the choice of negative sampling method and the introduction of additional hyperparameter tuning. Some unsupervised representation learning methods also require significantly greater training times; Momentum Contrast (He et al., 2019) trains a ResNet-50 for 1.25M training steps, which takes 6 days using 64 GPUs.

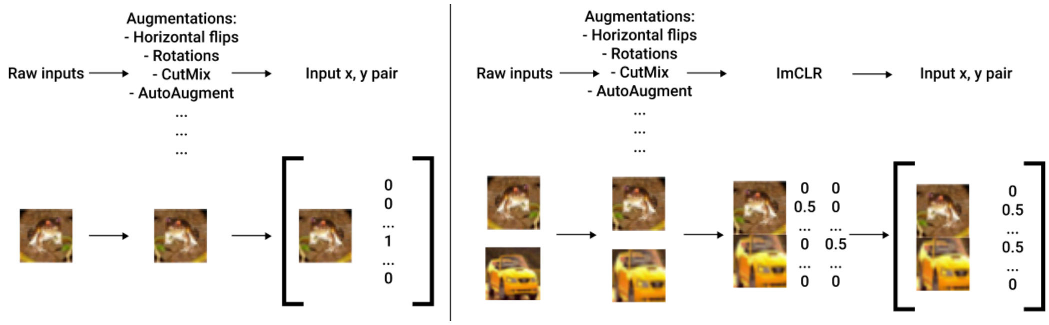

Figure 1: The Implicit Contrastive Learning Framework. Left: The standard one-hot training. Right: ImCLR with two images. Top: Abstract pipeline. Bottom: Concrete example.

In this paper, we consider the "online" supervised setting and introduce Implicit Contrastive Learning (ImCLR) for image classification. ImCLR proposes to input images to the network by simply concatenating images which then allow the neural network to implicitly learn the similarity and differences between images. We train the networks by presenting each input as a concatenation of two (or more) images, and thereby using a multi-hot vector as the label. In essence, for each sample the neural network is presented multiple images at once and is required to implicitly learn the semantics present in the images. We show ImCLR works well with existing tuned hyperparameters, has no change to existing losses or general network architecture, and is complementary to data augmentation techniques which allows for easy adoption and integration into modern deep learning pipelines.

Our contributions are as follows:

- We propose Implicit Contrastive Learning (ImCLR), a construction that allows neural networkss to implicitly learn the similarity and dissimilarity between images.
- ImCLR improves the test performance on existing image classification tasks, including by 3.24% on Tiny ImageNet with ResNet-56 (He et al., 2016), 1.40% on CIFAR-100 with VGG-16 (Simonyan & Zisserman, 2014), 0.64% on CIFAR-100 with PreAct ResNet-18, 2.28% on STL-10 with Wide-ResNet 16-8 (Zagoruyko & Komodakis, 2016), and 0.14% on CIFAR-10 with ResNet-20.
- Improvements carry over to the robustness setting, where we measure robustness to nineteen of the most common input corruptions and perturbations at five degrees of severity, and the semi-supervised learning setting, with 1% test error improvement on CIFAR-100-C (Hendrycks & Dietterich, 2019) with VGG-16 (Simonyan & Zisserman, 2014) and 2% test error improvement on CIFAR-10, with all but 4000 labeled samples with the Π-model (Laine & Aila, 2017).
- We demonstrate that ImCLR is complementary to existing data augmentation techniques, achieving over 1% test error improvement on CIFAR-100 with VGG-16, by combining ImCLR with state-of-the-art data augmentation method CutMix (Yun et al., 2019), as compared to either ImCLR or CutMix baseline alone. Furthermore, we achieve over 2% test error improvement on CIFAR-100 with PRN-18, by combining ImCLR with state-of-the-art augmentation method AutoAugment (Cubuk et al., 2018), as compared to either ImCLR or AutoAugment baseline alone.

2 The Implicit Contrastive Learning Framework

In contrast to recent advances in contrastive learning which requires the use of an explicit additional loss during unsupervised pretraining, we introduce the Implicit Contrastive Learning framework (ImCLR) where we aim to learn semantic relationships implicitly. The primary idea of ImCLR is to allow the network to implicitly learn the similarity and dissimilarity between images. In particular, we alter the input to the network to be a concatenation of two images, and the output to be a two-hot vector of 0.5 and 0.5; see Figure 1. The choice of 0.5 and 0.5 is a result of the Cross Entropy loss, and 1 and 1 can be explored for the Binary Cross Entropy loss. With such a construction on each input, the network will be forced to implicitly identify both images, and be encouraged to leverage

Table 1: Model Parameters for each experiment.

Experiment short name	one-hot	ImCLR	% Difference
RN56-TINYIMAGENET	$1,865,768$	$2,070,568$	10.9%
VGG16-CIFAR100	$15,038,116$	$15,300,260$	1.7%
PRN18-CIFAR100	$11,222,244$	$11,222,244$	0.0%
RN20-CIFAR10	$570,602$	$573,162$	0.4%
WRN-STL10	$11,002,330$	$11,048,410$	0.4%
VGG16-CIFAR100-C	$15,038,116$	$15,300,260$	1.7%
WRN-CIFAR10-SSL	$1,467,610$	$1,467,610$	0.0%
VGG16-CIFAR100-CUTMIX	$15,038,116$	$15,300,260$	1.7%
PRN18-CIFAR100-AA	$11,222,244$	$11,222,244$	0.0%
RN20-CIFAR10	$570,602$	($k=2$) $573,162$	0.4%
		($k=3$) $575,722$	0.9%
		($k=5$) $580,842$	1.8%
VGG16-CIFAR100-N	$15,038,116$	($k=2$) $15,300,260$	1.7%
		($k=3$) $15,562,404$	3.4%
		($k=5$) $16,086,692$	6.9%

the similarity and dissimilarity between images and thereby perform the contrastive learning task implicitlly.

This construction can be directly plugged into any existing image classification training pipeline, with the only typical changes being the sizes of the first and last layers of the network. The change in parameters is generally insignificant (e.g., $< 1\%$ for ResNet-20 on CIFAR10, or 0% for PreAct ResNet-18 on CIFAR100, due to average pool, with the exception being ResNet-56 on Tiny ImageNet; see Table 1). To ensure fairness in comparisons, we tune hyperparameters in the original standard one-hot supervised setting –including epochs to ensure performance has saturated– and we then apply the **exact same hyperparameters** to ImCLR. We note that for testing we concatenate the same image twice, with the one-hot vector used as the ground truth label.

2.1 Implementation and synergy with existing data augmentation

In the traditional setting, a batch size of k is defined by having k inputs per batch, where each of the k inputs is typically the result after data augmentation. For consistency with data augmentation techniques, which combine two or more images such as Mixup (Zhang et al., 2017), we define an input vector as a vector after the concatenation. In particular, and for simplicity of presentation, for each input, we assume we perform the following motions:

(a) Sample two images.
(b) Apply existing data augmentation to each image individually.
(c) Concatenate the two images as a single input vector.
(d) Rescale each label vector to sum to 0.5, and add them element-wise to produce the multi-hot label.

This paradigm can be easily extended to k-fold concatenation of images, where each label vector is rescaled to $1/k$, and then summed element-wise. We explore $k > 2$ in Section 3.5. For clarity, we present this procedure as well in Algorithm 1, where $k = 1$ is the standard one-hot training procedure, and $k = 2$ is the primary focus of this paper.

Algorithm 1 The ImCLR training framework. Produces one sample. For concatenating two images, we set $k = 2$. To recover the standard one-hot supervised training, we set $k = 1$.

Inputs: Samples $\{x_i, y_i\}_{i=0}^{k}$; x_i are inputs and y_i are one-hot labels; stochastic transformation T; number of images to concatenate k.
1. Compute $x_i = T(x_i)$.
2. Concatenate as $x = \texttt{concat}\left(\{x_i\}_{i=0}^{k}\right)$
3. Compute output $y = \frac{1}{k}\left(\sum_{i=0}^{k} y_i\right)$

return x, y

3 Results

We provide experimental results for supervised image classification, test error robustness against image corruptions and perturbations, semi-supervised learning, combining ImCLR with strong aug-

Table 2: Summary of experimental settings. SL = supervised learning; SSL = semi-supervised learning

Experiment short name	Model	Dataset	Setting
`RN56-TINYIMAGENET`	RN-56	Tiny ImageNet	SL
`VGG16-CIFAR100`	VGG-16	CIFAR100	SL
`PRN18-CIFAR100`	PreActResNet-18	CIFAR100	SL
`RN20-CIFAR10`	ResNet-20	CIFAR10	SL
`WRN-STL10`	Wide ResNet 16-8	STL10	SL
`VGG16-CIFAR100-C`	VGG-16	CIFAR100	robustness
`WRN-CIFAR10-SSL`	Wide ResNet 28-2	CIFAR10	SSL
`VGG16-CIFAR100-CUTMIX`	VGG-16	CIFAR100	augmentation
`PRN18-CIFAR100-AA`	PreActResNet-18	CIFAR100	augmentation
`RN20-CIFAR10-N`	ResNet-20	CIFAR10	ablation
`VGG16-CIFAR100-N`	VGG-16	CIFAR100	ablation

Table 3: Generalization error of experiments with and without ImCLR in the supervised setting.

Experiment	without ImCLR	with ImCLR	Absolute Improvement
`RN56-TINYIMAGENET`	42.03%	**38.79%**	$\downarrow$ 3.24%
`VGG16-CIFAR100`	27.80%	**26.50%**	$\downarrow$ 1.30%
`PRN18-CIFAR100`	25.93%	**25.29%**	$\downarrow$ 0.64%
`RN20-CIFAR10`	7.65%	**7.51%**	$\downarrow$ 0.14%
`WRN-STL10`	17.26%	**14.98%**	$\downarrow$ 2.28%
`VGG16-CIFAR100-C`	48.50%	**47.78%**	$\downarrow$ 0.72%

mentation, and an ablation study. A summary of experimental settings are give in Table 2 and comprehensively detailed in each section. We tuned the hyperparameters of the standard one-hot setting to achieve the performance of the original papers and of the most popular public implementations. We then used the *exact same hyperparameters and pipeline* for ImCLR for fairness.

3.1 Supervised Image Classification

In this section, we explore improving the performance of well-known baselines in the supervised learning setting. We add ImCLR to five model-dataset pairs, and lastly observe the performance with and without ImCLR across a varying number of supervised samples in the CIFAR100 setting. See Table 3 for results.

`RN56-TINYIMAGENET`. ResNet-56 (He et al., 2016) is a deep ResNet architecure with 56 layers. Tiny ImageNet is a dataset with 110,000 images of size $64 \times 64 \times 3$ and 200 classes. The test/train split is 100,000/10,000. We trained the model for 80 epochs with momentum SGD (step size set as $\eta = 0.1$, and momentum parameter $\beta = 0.9$), Cross Entropy loss, decaying by a factor of 0.1 at 40 and 60 epochs, using a batch size of 64. We applied the standard image augmentation (He et al., 2016) of centering, squishing min/max to -1/1, horizontal flips, and height/width shift range of 0.125. By adding ImCLR, the absolute generalization error was reduced by 3.24%, from 42.03% to 38.79%. By observing Figure 2 (Left), we see that while the two methods are initially comparable, adding ImCLR reduces the error in the later stages of training. The plateau of the ImCLR curve and the resistance to overfitting suggests implicit contrastive learning may also have an implicit regularization effect.

`VGG16-CIFAR100`. VGG-16 (Simonyan & Zisserman, 2014) is a 16 layer CNN with many 3x3 convolutional filters. CIFAR100 is a dataset with 100 classes and 500 samples per class in the training set, and 100 samples per class in the test set. Images are of size $32 \times 32 \times 3$. We adapted the original VGG-16 for the CIFAR100 dataset. We trained the model for 300 epochs with momentum SGD (step size set as $\eta = 0.1$, and momentum parameter $\beta = 0.9$), Cross Entropy loss, decaying learning rate schedule by a factor of 0.1 at 150 and 225 epochs, using a batch size of 128. We applied the standard image augmentation of centering, squishing min/max to -1/1, horizontal flips, and height/width shift range of 0.1. By adding ImCLR, the absolute generalization error was reduced by 1.30% from 27.80% (comparable to (Sankaranarayanan et al., 2018)) to 26.50%. Contrary to

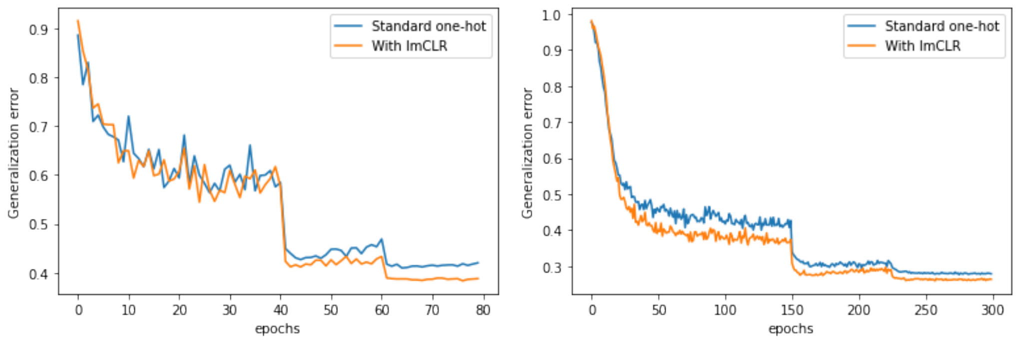

Figure 2: Generalization error for supervised learning. Left: `RN56-TINYIMAGENET`. Right: `VGG16-CIFAR100`.

`RN56-TINYIMAGENET`, we observe in 2 (Right) that ImCLR already improves in the early stages of training. It is generally typical in neural network training to see the gap closed in the first learning rate decay when there exists a gap early on in training, but here ImCLR maintains an improvement.

`PRN18-CIFAR100`. We utilize `VGG16-CIFAR100` as a base for several other experiments, so we include another architecture on CIFAR100 as supporting evidence. PreActResNet-18 (PRN-18) (He et al., 2016) is a variation of ResNet with a different residual block. Following popular implementations, we trained this network for 200 epochs with momentum SGD (step size set as $\eta = 0.1$, and momentum parameter $\beta = 0.9$) and decaying learning rate schedule by a factor 0.2 at 60, 120, and 180 epochs. The rest of the settings follow that of `VGG16-CIFAR100`. Similarly, we see an improvement for test error of 0.64%; such improvements, while minor, are observed on already fine-tuned scenarios, which indicates the effectiveness of our technique.

`RN20-CIFAR10`. ResNet20 (He et al., 2016) is a 20 layer deep residual neural network for image classification. CIFAR10 is the 10 class version of CIFAR100. Namely, there are 60,000 $32 \times 32 \times 3$ with a 50,000/10,000 train/test split, and 10 classes. The model was trained for 300 epochs with momentum SGD (step size set as $\eta = 0.08$, and momentum parameter $\beta = 0.9$), Cross Entropy loss, decaying learning rate schedule by a factor of 0.1 at 150 and 225 epochs, using a batch size of 128. Data augmentation follows `VGG16-CIFAR100`. This is a particularly challenging task to improve upon due to the model architecture where doubling the number the parameters and increasing the depth results in only minor gains in performance (He et al., 2016). In such a setting, adding ImCLR achieves a small gain of 0.15%.

`WRN-STL10`. Here we employ a Wide ResNet 16-8 (Zagoruyko & Komodakis, 2016), a 16 layer deep ResNet architecture with 8 times the width. STL-10 comprises 1300 images of size $96 \times 96 \times 3$ with a 500/800 train/test split and 10 classes. This is a more challenging dataset than CIFAR10 due to the number of training samples and size of images. We trained the WRN model for 100 epochs with momentum SGD (step size set as $\eta = 0.1$, and momentum parameter $\beta = 0.9$), Cross Entropy loss, decaying learning rate by a factor of 0.1 at 50 and 75 epochs, using a batch size of 64. Data augmentation follows `VGG16-CIFAR100`. A 2.28% absolute test error is gained here.

Understanding performance with varying epochs. ImCLR performs well in the above supervised settings, and we further explore performance in the low sample regime. In particular, we select the `VGG16-CIFAR100` setting, and decrease the number of samples in each class proportionally. We use the exact same training setup as in the full `VGG16-CIFAR100` case, and tabulate results in Table 4. We perform 3 runs since low-sample settings produce higher variance results. The improvement for the full dataset setting hold with lower samples at roughly 2% generalization error.

Table 4: Generalization error (%) for VGG16-CIFAR100 with varying number of proportional samples in each class.

samples%	100	50	30	20	10	5
CE	27.80 ±.10	34.88 ±.20	42.52 ±.34	50.41 ±.38	71.91 ±.57	86.03 ±.12
ImCLR	**26.50 ±.11**	**33.61** ±.21	**40.40** ±.34	**48.19** ±.52	**68.71** ±.87	**85.61** ±.40

Table 5: Generalization error of Π-model on the standard benchmark of CIFAR10, with all but 4,000 labels removed.

Experiment	without ImCLR	with ImCLR	Absolute Improvement
`WRN-CIFAR10-SSL`	17.31%	**15.15**%	↓ 2.16%

3.2 Robustness

We investigate the impact of ImCLR on robustness. In particular, we select a corrupted dataset as test set and reevaluate models trained with and without ImCLR on the uncorrupted training set, following standardized procedures (Hendrycks & Dietterich, 2019).

`VGG16-CIFAR100-C`. The CIFAR100-C (Hendrycks & Dietterich, 2019) dataset is a test set for CIFAR100 of 10,000 images, where each image is corrupted at 5 different severities, resulting in a test set of size 50,000. Nineteen of the most popular corruptions are selected, including various noise, blur, weather, digital, and other corruptions. These corruptions are performed individually, and the average test error across all corruptions and corruption levels is given in the last row of Table 3, where ImCLR reduces test error by 0.72%. Hendrycks & Dietterich (2019) advocates a mean Corruption Error which is calculated as the mean of the proportions to the performance of AlexNet for each corruption type, and while we did not benchmark AlexNet, the improvement in mean Corruption Error is expected to be larger.

3.3 Semi-supervised Learning

Thus far, ImCLR is applied under the Cross Entropy loss. While varying the number of samples is helpful in understanding the impact of ImCLR under different settings, we explore if ImCLR can be directly applied to improve Semi-Supervised Learning (SSL), where the network processes both labeled and unlabeled samples. We select a popular and practical subset of SSL, which involves adding a loss function for consistency regularization. Consistency regularization is similar to contrastive learning in that it tries to minimize the difference in output between similar samples. In particular, we select the classic and standard benchmark of the Π-model (Laine & Aila, 2017).

The Π model adds a loss function for the unlabeled samples of following form:

$$d(f_\theta(x), f_\theta(\hat{x})),$$

where d is typically the Mean Square Error, f_θ is the output of the neural network, and $\hat{x}$ is a stochastic perturbation of x. Minimizing this loss enforces similar output distributions of an image and its perturbation. A coefficient is then applied to the SSL loss as a weight with respect to the Cross Entropy loss. By adding this additional loss function, the unlabeled samples are evaluated with the SSL loss, while the labeled samples are evaluated with Cross Entropy.

`WRN-CIFAR10-SSL`. We follow the standard setup in Oliver et al. (2018) for the CIFAR10 dataset, where 4000 labeled samples are selected, and remaining samples are unlabeled. We use a WRN 28-2 architecture (Zagoruyko & Komodakis, 2016), training for 200,000 iterations with a batch size of 200, of which 100 are labeled and 100 are unlabeled. The Adam optimizer is used ($\eta = 3e-4, \beta_1 = 0.9, \beta_2 = 0.999$), decaying learning rate schedule by a factor of 0.2 at 130,000 iterations. Horizontal flips, random crops, and gaussian noise are used as data augmentation. A coefficient of 20 is used for the SSL loss. By adding ImCLR, we reduce the test error by 2.16%.

3.4 ImCLR is complementary to existing data augmentation

Data augmentation is critical in training neural network models. Recently, stronger forms of data augmentation (Zhang et al., 2017; DeVries & Taylor, 2017; Yun et al., 2019; Cubuk et al., 2018) have provided substantially improved results on a variety of benchmarks. Here, we select state-of-the-art data augmentation method CutMix, an effective technique where a section of one image are pasted onto another and labels are correspondingly weighted, and AutoAugment, a reinforcement learning approach to choosing effective data augmentations. We apply CutMix (AutoAugment) to produce samples prior to ImCLR, and treat each sample post-CutMix (post-AutoAugment) as an input sample to ImCLR (in Algorithm 1).

Table 6: Generalization error (%) of `VGG16-CIFAR100` with CutMix.

Experiment	Standard	CutMix	ImCLR	ImCLR + CutMix
`VGG-CIFAR100-CUTMIX`	27.80 $\pm$ .10	27.20 $\pm$ .11	26.50 $\pm$ .11	**25.49** $\pm$ .13

Table 7: Generalization error (%) of `PRN18-CIFAR100` with AutoAugment.

Experiment	Standard	AA	ImCLR	ImCLR + AA
`PRN18-CIFAR100-AA`	25.93	23.87	25.29	**21.51**

`VGG16-CIFAR100-CUTMIX`. We follow the same experimental settings as in the supervised setting previously described, and include previous results in Table 6. CutMix improves on the standard training setup of horizontal flips and other weaker augmentations. However, combining ImCLR with CutMix results in almost 2% better absolute error than just CutMix. This suggests that ImCLR is complementary to existing data augmentation techniques, and not a replacement. This strengthens the notion that it can be easily plugged directly into existing pipelines.

`PRN18-CIFAR100-AA`. We follow experimental settings in `PRN18-CIFAR100`. We use existing AutoAugment policies for the CIFAR datasets, and following Cubuk et al. (2018) for CIFAR, we apply AutoAugment after other augmentations, and before normalization and ImCLR. AutoAugment improves 2% over standard augmentation, and adding ImCLR improves by another 2% (see Table 7); again, suggesting a complementary behavior and easy incorporation into existing pipelines.

3.5 Ablation study

Throughout this paper, we have studied ImCLR in the setting of the concatenation of two images ($k = 2$ in Algorithm 1). We now perform an ablation study to determine how far this framework can be pushed. Namely, we increase the value of k, and observe the test error in the setting of `VGG16-CIFAR100` and `RN20-CIFAR100`. We fix the hyperparameters as used previously, with results given in Table 8 and Figure 3.

The error deteriorates immediately after $k = 2$, where the case $k = 3$ returns to the error of the standard one-hot case, and further increasing k typically increases the error further. This behavior is clearer in the case of `VGG16-CIFAR100`. We can see in Figure 3 that the choice of k has limited impact in the early stages of training, but affects the final test error, where performance begins to deteriorate after the first learning rate decay.

Furthermore, we also explore the concatenation of the same image. The semantic meaning of concatenation of the same image is that the network must recognize both parts of the image are of the same class. However, this performs on par with the standard one-hot case and performs worse than the concatenation of two different images. First, this results in a sanity check that the ImCLR construction is identical (with respect to performance) to the one-hot vector classification constructions. Second, worse performance in ImCLR when the same image is concatenated twice indicates that the network learns less, as compared to the concatenation of two images: this further strengthens the implicit regularization that ImCLR brings during training.

4 Related Work

Contrastive learning (Hadsell et al., 2006; Becker & Hinton, 1992) fundamentally relates to the idea that similar images should have similar representations, and dissimilar images should have

Table 8: Generalization error for `VGG16-CIFAR100` and `RN20-CIFAR10` with varying number of images concatenated.

k	1 (standard)	2 (same image)	2	3	5
`VGG16-CIFAR100`	27.80%	27.69%	**26.50%**	27.35%	29.35%
`RN20-CIFAR10`	7.65%	7.73%	**7.51%**	8.13%	7.89%

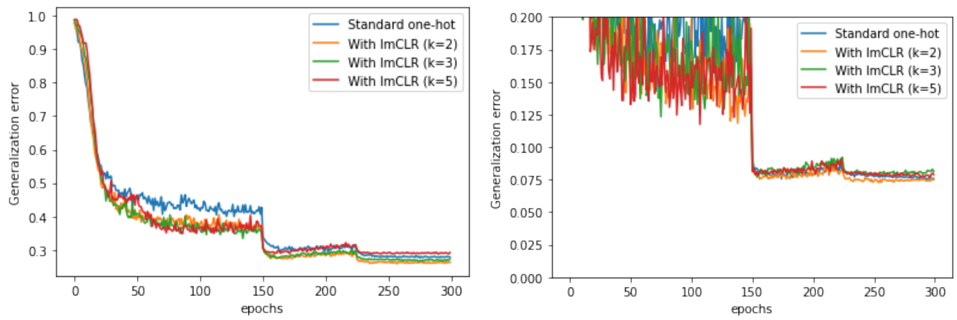

Figure 3: Left: Generalization error for `VGG16-CIFAR100`. Right: Generalization error for `RN20-CIFAR10`. Varying number of images concatenated.

dissimilar representations. This idea has been substantially and effectively explored in the recent self-supervised and unsupervised representation learning literature (Chen et al., 2020b;c; He et al., 2019; Sermanet et al., 2018; Tian et al., 2019; Wu et al., 2018; Hénaff et al., 2019; Hjelm et al., 2019). Contrastive learning in the self-supervised/unsupervised setting has been thoroughly explored and implemented using triplet loss functions (Schroff et al., 2015). These contrastive losses (Hadsell et al., 2006) are considered to influence and encourage similarities and dissimilarities in learned representation, such as by constructing a loss which increases the cosine similarity or reduces the euclidean distance between similar images.

This topic is tightly related to other methods including negative sampling/contrastive estimation (Mikolov et al., 2013; Smith & Eisner, 2005), which relies on implicit negative evidence which exists in other unlabeled samples. Contrastive learning is also related to consistency regularization in semi-supervised learning (Chapelle & Scholkopf, 2006), where the focus of consistency regularization is in consistency on the output distribution with respect to stochastic perturbations to the input (Lee, 2013; Laine & Aila, 2017; Berthelot et al., 2019; Chen et al., 2020a). In addition, there are ties to pre-text tasks (Zhang et al., 2016; Doersch et al., 2015; Kolesnikov et al., 2019; Noroozi & Favaro, 2016), such as rotation prediction (Gidaris et al., 2018), where the network learns a representation of the data by performing an unsupervised task, which then aids learning of the supervised task.

In supervised learning, several ideas have been recently introduced that significantly boosts the performance in supervised learning. These techniques can be added to the label, such as label smoothing (Sukhbaatar et al., 2014), or directly to the data, using data augmentation (Zhang et al., 2016; Cubuk et al., 2018; DeVries & Taylor, 2017; Yun et al., 2019), or both (Zhang et al., 2017).

In machine learning, there are two related frameworks that output multiple labels from a single image, namely ensembles (Dietterich, 2000) and multiple choice learning (Guzman-Rivera et al., 2012). Both ensembles and multiple choice learning aim to output multiple labels fom the same input; ensembles utilize multiple models to obtain multiple predictions from the same input, while multiple choice learning predicts multiple labels from the same model. Recent literature in ensemble learning have explored improving an ensemble of neural networks (Hansen & Salamon, 1990) with random initialization (Lakshminarayanan et al., 2017), attention (Kim et al., 2018), information theoretic objectives (Sinha et al., 2020), among others. ImCLR is strictly different from both ensembles and multiple choice learning as our aim is to predict multiple outputs from multiple inputs.

5 Conclusion

We introduce Implicit Contrastive Learning (ImCLR) for image classification. ImCLR encourages neural networks to implicitly learn the similarity and difference between images by concatenating multiple images per sample with a multi-hot vector as target. ImCLR can directly be plugged into existing pipelines with minimal changes, and works well with no change in loss, hyperparameters, or general network architecture. ImCLR improves the performance of supervised image classification in a variety of standard benchmarks including Tiny ImageNet, CIFAR-10, CIFAR-100, and STL-10. Furthermore, ImCLR improves robustness to corruptions, semi-supervised learning, and is complementary to existing data augmentations. ImCLR is simple to implement and we hope is useful both practically and as the subject of future research.

REFERENCES

Suzanna Becker and Geoffrey Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 1992.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Avital Papernot, Nicolas Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.

Olivier Chapelle and Bernhard Scholkopf. Semi-supervised learning. *MIT Press*, 2006.

John Chen, Vatsal Shah, and Anastasios Kyrillidis. Negative sampling in semi-supervised learning. *ICML*, 2020a.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020b.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020c.

Ekin Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc Le. Autoaugment: Learning augmentation policies from data, 2018.

Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout, 2017.

Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pp. 1–15. Springer, 2000.

Carl Doersch, Abhinav Gupta, and Alexei Efros. Unsupervised visual representation learning by context prediction. *ICCV*, 2015.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.

Abner Guzman-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, pp. 1799–1807, 2012.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. *CVPR*, 2006.

Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *ICLR*, 2019.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding, 2019.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 736–751, 2018.

Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019.

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

Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pp. 6402–6413, 2017.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. *ICML Workshop on Challenges in Representation Learning*, 2013.

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

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *European Conference on Computer Vision*, 2016.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *arXiv preprint arXiv:1804.09170*, 2018.

Swami Sankaranarayanan, Arpit Jain, Rama Chellappa, and Ser Nam Lim. Regularizing deep networks using efficient layerwise adversarial training. *arXiv preprint arXiv:1705.07819*, 2018.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time contrastive networks: Self-supervised learning from video. *ICRA*, 2018.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Samarth Sinha, Homanga Bharadhwaj, Anirudh Goyal, Hugo Larochelle, Animesh Garg, and Florian Shkurti. Dibs: Diversity inducing information bottleneck in model ensembles. *arXiv preprint arXiv:2003.04514*, 2020.

Noah A Smith and Jason Eisner. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 354–362. Association for Computational Linguistics, 2005.

Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels, 2014.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, 2017.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding, 2019.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *International Conference on Computer Vision (ICCV)*, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Pas. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. *European conference on computer vision*, 2016.