BarlowMatch: Combining Redundancy Reduction based Self-Supervision and Consistency Regularization based Semi-Supervision

Rajashekar Vasantha [*] **Surabhi Ranjan** **Sumit Mamtani**

Abstract

This paper introduces BarlowMatch, a semi-supervised learning method that combines the power of redundancy reduction based self-supervised learning and consistency regularization based semi supervised learning. BarlowMatch consists of two stages: (1) self-supervised pre-training based on redundancy reduction and (2) semi-supervised fine-tuning based on augmentation consistency regularization and pseudolabeling. This paper also presents an active learning method based on diversity sampling to select images to be labeled. Using BarlowMatch we obtain a Top-1 accuracy of 35.22% using 5% labeled data. This accuracy increases to 36.69% when we add an extra 2.5% labels obtained using diversity maximization. The code is available at https://github.com/rajashekarv95/DLSP-2021-Project

1. Introduction & Related Work

Self-supervised learning (SSL) models achieve scalability at a low cost and are useful in learning more subtle, less common representations of the input data without relying on human annotations. Models pre-trained using SSL have been shown to yield higher performance than when solely trained in a supervised manner as demonstrated in natural language processing (Radford et al., 2019), speech recognition (Rivière et al., 2020), and computer vision tasks (Goyal et al., 2021)

Within computer vision, methods based on autoencoders (Ranzato et al., 2007), clustering (Caron et al., 2018), instance-level discrimination (Bojanowski & Joulin, 2017) have outperformed supervised learning in many downstream tasks (He et al., 2020; Grill et al., 2020) and they demonstrate the benefits of pre-training in the field of computer vision. However, many of these methodologies have surpassed supervised learning within the limited scope of datasets originally created for supervised or weakly supervised learning (Henaff, 2020). Some research using uncurated data (Joulin et al., 2016) have trained SSL models on random data using datasets consisting of a few million images and ongoing research (Goyal et al., 2021) indicates that self-supervised learning is indeed a promising way forward.

Within SSL, some methodologies aim to learn input data representations which are invariant to distortions in the image (also known as data augmentations). These methods maximize the similarity between representations obtained from different distortions of an image by passing this image through a Siamese network. (Zbontar et al., 2021; Sohn et al., 2020). To prevent trivial solutions from data augmentations, methods like Simsiam (Chen & He, 2020) update network architecture and parameters in an asymmetric fashion.

Contrastive methods compute pairwise comparisons between 'positive' and 'negative' samples (Grill et al., 2020) eg. in SimCLR (Chen et al., 2020), a data point is augmented to two different views which are sent to two deep ReLU networks having identical weights and the output from each is sent to a contrastive loss function.

Clustering methods use distorted samples to calculate a 'target' for the loss and a different distorted version of the sample to predict this target using k-means in DEEPCLUSTER (Caron et al., 2018) or non-differentiable operators in SWAV (Caron et al., 2020).

In this paper, we have combined self-supervised pre-training with semi-supervised fine-tuning to solve an 800-way classification problem using 5% labeled images.

2. Method

BarlowMatch consists of two parts - (a) self-supervised pre-training based on redundancy reduction - Barlow Twins (Zbontar et al., 2021) and (b) semi-supervised fine-tuning based on augmentation consistency regularization and pseudo-labeling - FixMatch (Sohn et al., 2020). The architecture diagram of BarlowMatch is shown in Figure 1.

[*] Equal contribution . Correspondence to: Rajashekar Vasantha <rv2138@nyu.edu>.

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

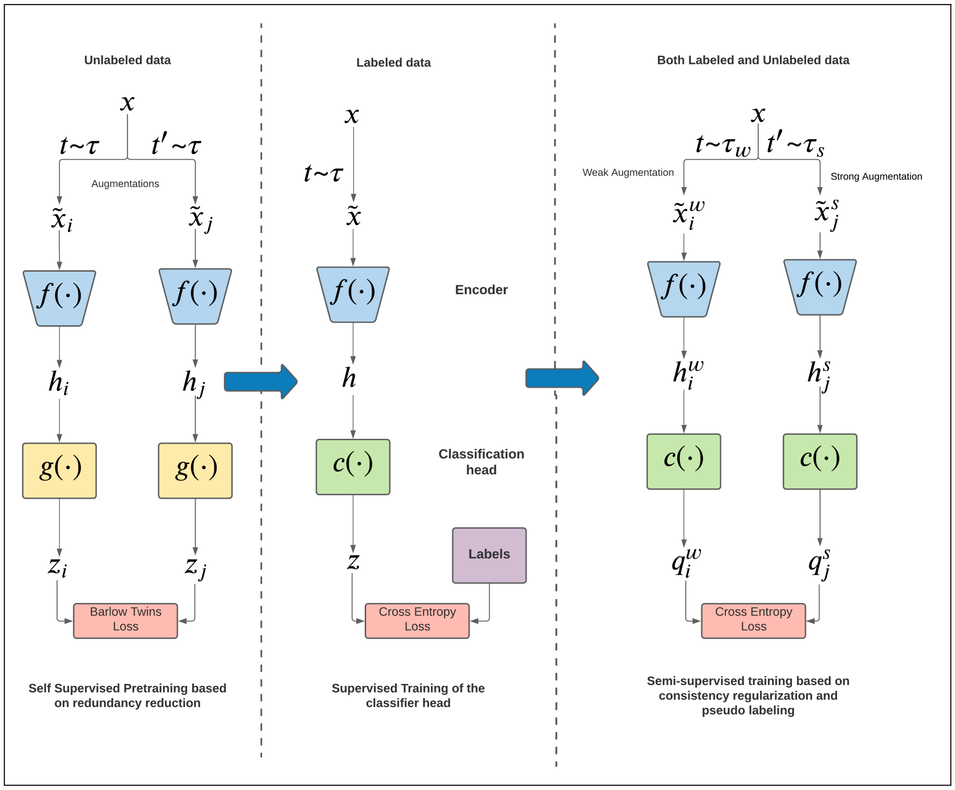

Figure 1. Architecture diagram of BarlowMatch which includes three stages. (1) Self supervised pre-training using Barlow Twins, (2) Supervised training using labeled data and (3) Semi-Supervised fine tuning using FixMatch

We use Barlow Twins for self-supervised pre-training. In this method, through redundancy reduction, twin representations of an unlabelled image are obtained by passing two different augmentations of the same image through the same network. The network is then trained to make the empirical cross correlation matrix of the twin representations to be as close to the identity matrix as possible. This makes the representation invariant to the distortions and also de-correlates the different vector components of the representation. We minimize the following loss in this stage of learning.

$$\mathcal{L}_{BT} = \sum_{i} (1 - C_{ii}) + \lambda \sum_{i} \sum_{i \neq j} C_{ij}^2 \tag{1}$$

where,

$$C_{ij} = \frac{\sum_b z^A_{b,i} z^B_{b,j}}{\sqrt{\sum_b (z^A_{b,i})^2}\sqrt{\sum_b (z^B_{b,j})^2}} \tag{2}$$

where b indexes the batch and i, j index the vector dimensions of the network's output. After pre-training on Barlow Twins, a classifer was trained on top of the backbone using supervised learning on the labeled data.

Post supervised training, the model was finetuned using FixMatch. FixMatch combines consistency regularization and pseudo-labeling. Consistency regularization utilizes unlabeled data by relying on the assumption that the model should output similar predictions when fed perturbed versions of the same image. Pseudo-labeling leverages the idea of using the model itself to obtain artificial labels for unlabeled data. An artificial label was computed for each unlabeled example which was then used in a standard cross-entropy loss. The artificial label was computed using a weakly augmented image and its strongly augmented form was used to predict this pseudo-label.

The loss function for FixMatch consists of two cross-entropy loss terms: a supervised loss l_l applied to labeled data and an unsupervised loss l_u. Specifically, l_l is just the standard cross-entropy loss on weakly augmented labeled examples x_l:

$$l_s = \frac{1}{B} \sum_{b=1}^{B} H(p_l, p(y|\tau(x_l))) \tag{3}$$

FixMatch computes an artificial label for each unlabeled example which is then used in a standard cross-entropy loss. To obtain an artificial label, we first compute the model's predicted class distribution given a weakly-augmented version of a given unlabeled image: $q_u = p(y|\tau(x_u))$. Then, we use $\hat{q}_u = \arg\max(q_u)$ as a pseudo-label, except we enforce the cross-entropy loss against the model's output for a strongly-augmented version of x_u:

$$l_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbb{1}(\max q_u \geq t) H(\hat{q}_u, p(y|\tau'(x_u))) \tag{4}$$

where t is a scalar hyperparameter denoting the threshold above which we retain a pseudo-label. Combining these using λ_u, which is a hyperparameter denoting the relative weight of the unlabeled loss, we get:

$$\mathcal{L}_{FM} = l_l + \lambda_u l_u \tag{5}$$

3. Implementation Details

3.1. Barlow Twins

3.1.1. Image Augmentations

We adopt the augmentation pipeline similar to Barlow Twins (Zbontar et al., 2021).The image augmentation pipeline consists of the following transformations: random cropping, resizing to 96 × 96, horizontal flipping, color jittering, converting to grayscale, Gaussian blurring, and solarization. The first two transformations (cropping and resizing) are always applied, while the last five are applied randomly, with some probability. This probability is different for the two distorted views in the last two transformations (blurring and solarization).

3.1.2. Architecture

The encoder f consists of a modified Wide-ResNet-50 network (Zagoruyko & Komodakis, 2016). To deal with lower resolution images, the first convolution layer was modified to have filters of size 3 instead of 7 which was used in the original implementation. The final fully connected layer was also removed and a projector network g made up of three linear layers was added. Each linear layer consisted of 8192 output units. The first two layers of the projector are followed by a batch normalization layer and rectified linear units.

3.1.3. Optimization

We use the LARS optimizer (You et al., 2017) and train for 400 epochs with a batch size of 512. The learning rate starts at 0 and is linearly increased to 0.05 during the first 5 epochs of training, after which it is decreased to 0.002 using a cosine decay schedule (Loshchilov & Hutter, 2016). We use $\lambda = 5 \times 10^{-6}$ and a weight decay parameter of 1.5×10^{-6} as suggested by the Barlow Twins paper (Zbontar et al., 2021).

3.2. Classifier

3.2.1. Architecture

The classifier c consists of four linear layers with 8192 output each with the final layer producing 800 dimensional logits. The linear layers of the classifier are followed by a batch normalization layer, rectified linear units and dropouts. We observed that a classifier deeper than four layers led to a drop in performance.

3.2.2. Optimization

The backbone and the classifier was trained via supervised learning using Adam Optimizer and cross entropy loss. We used a learning rate of 10^{-5} on the backbone model parameters and a learning rate of 10^{-2} on the classifier parameters. During training, a dropout of 0.1 and weight decay of 0.001 was applied.

3.3. Fixmatch

3.3.1. Image Augmentations

We leverage the weak and strong augmentations as described in FixMatch (Sohn et al., 2020). For weak augmentation, we use random cropping, resizing to 96×96 followed by random horizontal flip. For strong augmentation, we use RandAugment (Cubuk et al., 2019) and CTAugment followed by CutOut.

3.3.2. Optimization

We use a standard SGD with momentum to optimize the FixMatch loss. For a learning rate schedule, we use a cosine learning rate decay (Loshchilov & Hutter, 2016) which sets the learning rate to $\eta \cos \frac{7\pi k}{16K}$ where η is the initial learning rate, k is the current training step, and K is the total number of training steps. We run the optimization for 40 epocs where one epoch is defined as one pass over the unlabeled dataset. We use a batch size $B = 64$ and $\mu = 7$. This means that for every optimization step, we use 7 times the unlabeled data compared to labeled data. We set the threshold $t = 0.9$ so as to consider only the unlabeled images for which our model is highly confident. We set $\lambda_u = 1$ so that both labeled and unlabeled loss are given equal weight.

4. Label Request Methodology

We make use of diversity based sampling to determine the images that we requested labels for. The idea was to find a subset of unlabeled images that are most far away from the labeled images in the representation space. Using the representations from the self-supervised pre-trained model, the labeled images were clustered using the KMeans algo-

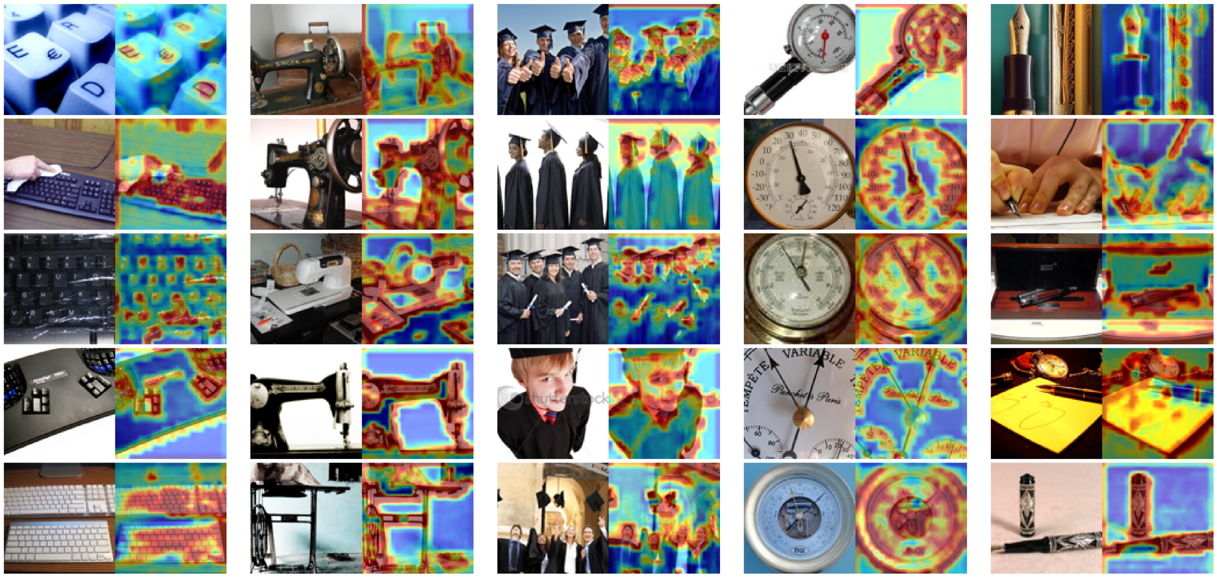

Figure 2. Guided GradCAM visualizations of a few images from the dataset.

rithm. We chose the number of clusters to be 800 as we were solving a 800 label classification problem. For each unlabeled image, we computed the distance to the closest cluster among the 800 clusters created. 12800 images with the greatest minimum distance were chosen to request labels for. This meant that we obtained labels for parts of the representation space that were least represented by the current labeled dataset.

5. Results

As shown in the table below, using only Barlow Twins gave us an accuracy of 32.80% and adding Fixmatch to it increased accuracy by 1%. With extra labels, we observed that semi-supervised finetuning augments the self supervised pre training and led to an overall increase in accuracy by 3 %.

Table 1. Top-1 Accuracies

Model	Train Accuracy	Validation Accuracy	Validation Accuracy with extra labels
Barlow + Classifier	97.89%	32.80%	35.22%
Barlow + Classifier + Fixmatch	99.23%	33.70%	36.69%

We visualize in Figure 2 some of the images using GradCAM (Selvaraju et al., 2016). This uses the gradients of any target concept (say ‘dog’ in a classification network) flowing into the final convolutional layer to produce a coarse localization map highlighting the important regions in the image for predicting the concept. This method combines guided backpropagation and gradient-weighted class activation mapping. The red areas correspond to high contribution towards a particular class score.

We notice some interesting behaviour looking at the visualizations. The model is able to learn visual features and extrapolate it to new images. In the first column, fourth image - whose class is a computer keyboard - the model correctly recognizes a keyboard which is of a different shape compared to usual keyboards. In the third column - whose class is most probably related to graduation gown - the model learns to correctly recognize the shape of the graduation cap and the patterns present on the graduation gown. The model also recognizes the caps thrown in the air in the last image. In the right-most column, we see images belonging to the class of fountain pen. The model learns to correctly identify the nib of a fountain pen - as can be seen in the second image from the top. In this image, although the entire pen is not visible, the model is able to recognize a fountain pen just by its nib.

This shows that the pre-training using redundancy reduction has indeed led to learning useful visual features.

6. Discussion and Future Work

In this paper, we proposed a combination of self-supervised learning framework based on redundancy reduction and semi-supervised learning fine-tuning based on consistency regularization and pseudo-labeling. This model can be further improved by refining the loss function used in Barlow Twins (Zbontar et al., 2021).

We also realize that the Barlow Twins backbone was under trained as the loss curve did not saturate. Training for longer should improve the representation learnt by a large extent.

For the label request, we can refine the diversity-based sampling method further by incorporating the diversity among the unlabeled samples as well. We impose another constraint to ensure that the chosen unlabeled images are also far from each other in the representation space. This prevents similar unlabeled images from being chosen and further increases the diversity of the chosen images.

References

Bojanowski, P. and Joulin, A. Unsupervised learning by predicting noise. In *International Conference on Machine Learning*, pp. 517–526. PMLR, 2017.

Caron, M., Bojanowski, P., Joulin, A., and Douze, M. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149, 2018.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Chen, X. and He, K. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. Randaugment: Practical data augmentation with no separate search. *CoRR*, abs/1909.13719, 2019. URL http://arxiv.org/abs/1909.13719.

Goyal, P., Caron, M., Lefaudeux, B., Xu, M., Wang, P., Pai, V., Singh, M., Liptchinsky, V., Misra, I., Joulin, A., et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Henaff, O. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pp. 4182–4192. PMLR, 2020.

Joulin, A., Van Der Maaten, L., Jabri, A., and Vasilache, N. Learning visual features from large weakly supervised data. In *European Conference on Computer Vision*, pp. 67–84. Springer, 2016.

Loshchilov, I. and Hutter, F. SGDR: stochastic gradient descent with restarts. *CoRR*, abs/1608.03983, 2016. URL http://arxiv.org/abs/1608.03983.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Ranzato, M., Huang, F. J., Boureau, Y.-L., and LeCun, Y. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition*, pp. 1–8. IEEE, 2007.

Rivière, M., Joulin, A., Mazaré, P.-E., and Dupoux, E. Unsupervised pretraining transfers well across languages. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7414–7418. IEEE, 2020.

Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

Sohn, K., Berthelot, D., Li, C.-L., Zhang, Z., Carlini, N., Cubuk, E. D., Kurakin, A., Zhang, H., and Raffel, C. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

You, Y., Gitman, I., and Ginsburg, B. Scaling SGD batch size to 32k for imagenet training. *CoRR*, abs/1708.03888, 2017. URL http://arxiv.org/abs/1708.03888.

Zagoruyko, S. and Komodakis, N. Wide residual networks. *CoRR*, abs/1605.07146, 2016. URL http://arxiv.org/abs/1605.07146.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.