MixMatch: A Holistic Approach to Semi-Supervised Learning - A Replication

Annie B Rapheal rapheal@kth.se

Aditya D Udapudi udapudi@kth.se

Filippa Modigh fmodigh@kth.se

Raghav Bongole bongole@kth.se

Abstract

Semi supervised learning comes into the picture to take advantage of both supervised and unsupervised ML. MixMatch is a semi-supervised learning method that is an extension to MixUp for supervised learning using both labeled and unlabeled data. In this project the MixMatch algorithm is explored for the CIFAR-10 dataset. To understand the components that contribute to the performance, an ablation study is conducted. The results obtained indicates that MixMatch results in comparable performance to a supervised model.

1 Introduction

During the last few years a substantial amount of work is being done in the field of machine learning and artificial intelligence, most of which involve training large and complex neural networks. But such networks often need a huge amount of data for training and testing purposes. It is quite challenging to find labelled data suited for all the different types of tasks and experiments. While obtaining unlabelled data is quite cheap and less time consuming, this type of data can also be exploited using the Semi Supervised Learning(SSL) methods. Semi supervised learning as defined in (2) is a method which is halfway between supervised and unsupervised learning. In addition to unlabeled data, the algorithm is provided with some supervision information – but not necessarily for all examples.

One such method is MixMatch method (1) which implements a stochastic data augmentation technique to an unlabelled image data set and finally classified into one of the pre-defined classes. Being able to use the vast amount of unlabeled data to improve a model would give significant advantages. Therefore, we aim to study this method by reproducing the experiments and results obtained in (1) on various parameters involved in the algorithm. The replication can be accessed via a GitHub repository with [1].

2 Related Work

This replication of MixMatch is based on the work by Berthelot, et al. (2019) (1) in which the authors combined different state-of-the-art techniques for semi-supervised learning to produced the new algorithm with significantly reduced the error rate. In this replication, we have followed the MixMatch procedure very closely. More technical details will be presented in Section 4 but a brief description of the MixMatch elements in (1) can be summarized as follows:

(a) Data augmentation: a common regularization technique that deforms or add noise to the input images. The idea is that the model should output the same class distribution for an augmented input example even after it has been augmented.

[1] https://github.com/filippad/DeepLearning21-Project

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

(b) Label guessing using sharpening: a model is used to predict the labels of the augmented unlabelled images. Those guesses will then be "sharpened" with a temperature hyperparamter. These sharpened guesses are later used in the unsupervised loss term.

(c) MixUp: Both labeled and unlabeled examples and their labels are then concatenated and shuffled to create a source for MixUp(4). Each labeled and unlabeled example and target is then mixed up with this source. The result is added to a collection that finally can be used for traning.

3 Data

We evaluate the models using the **CIFAR-10** dataset. It contains 60000 color images, i.e 32x32x32 RGB inages, with 10 classes with 6000 images per class. There are 50000 training images and 10000 test images. We have not performed any preprocessing to this dataset in this project.

According to a tracking record by *Papers with Code* [2], the best classification method on CIFAR10 is currently EffNet-L2 by Google Research with an accuracy rate of 99.70%. This technique does not seek to minimize the commonly used loss training (e g. cross-entropy) but seeking for the parameters that lie in neighbourhoods that have uniformly low loss.

4 Methods

The Mixmatch algorithm incorporates various techniques to enhance the state-of-the-art work done in the field of semi-supervised learning. Specifically, along with data augmentation, techniques such as label-guessing, sharpening and mix-up have been incorporated. These techniques can ensure consistency regularization and can help achieve entropy minimization. The subsections following the Data Augmentation subsection are done on a per-batch basis.

4.1 Data Augmentation

The Mixmatch algorithm uses augmentations that are performed on both labeled and unlabeled data. In slight contrast to the original algorithm, we perform these augmentations before splitting the data into batches. This, however, is not a new idea and has been carried out in some implementations of the algorithm. Each labeled sample is augmented once to obtain an augmented label set and each unlabeled sample is augmented 'K' times to obtain another set. The augmentations consist mainly of padding, cropping or randomly flipping an image.

4.2 Label Guessing

The set obtained after augmentation of the unlabeled data is then utilized to guess labels of each unlabeled sample point. Specifically, if u_i is an unlabeled sample then its label g_i is given by,

$$g_i = \frac{1}{k}\sum_{k=1}^{K} P(u_i) \tag{1}$$

where P denotes the application of the neural network model which outputs a vector of probabilities of each class. This is done during each update step (per batch) in the algorithm.

4.3 Sharpening

This step is done mainly to achieve entropy minimization. Here, the output of the label guessing function is taken and sharpened according to this equation:

$$Sharpen(g_{ij}, T) = \frac{g_{ij}^{\frac{1}{T}}}{\sum_j g_{ij}^{\frac{1}{T}}} \tag{2}$$

where j denotes the index of each element in the vector of label guessed probabilities g_i.

[2] https://paperswithcode.com/sota/image-classification-on-cifar-10

4.4 Mixup

The unlabeled images along with their corresponding sharpened labels guessed probabilities together with the labeled images along with their targets are shuffled together to form a set W. This set is "mixed up" with the augmented label set and the augmented unlabeled set. More formally, let L be the labeled set and T be its corresponding target values. Let U denote the unlabeled set and G denote the sharpened targets. We follow these steps:

$$
\begin{gathered}
all\_data = Concat(L, U) \\
all\_targets = Concat(T, G) \\
W\_data = Shuffle(All\_data) \\
W\_targets = Shuffle(All\_targets) \\
\lambda = Beta\_distribution(\alpha, \alpha) \\
\lambda = max(\lambda, 1 - \lambda) \\
data^{'} = \lambda * all\_data + (1 - \lambda) * W\_data \\
targets^{'} = \lambda * all\_targets + (1 - \lambda) * W\_targets
\end{gathered}
$$

Here, we set the hyper-parameter α value to 0.75.

4.5 Model

We use a Wide-Resnet 28 model for our experiments following (3). Let P denote the model. We denote $mixed\_labeled\_inputs$ as the part of $targets'$ that was mixed up with L and $mixed\_unlabeled\_inputs$ as the part mixed up with U. We also gather the sets $mixed\_labeled\_targets$ and $mixed\_unlabeled\_targets$ correspondingly. Then,

$$
\begin{gathered}
log\_prob\_x = P(mixed\_labeled\_inputs) \\
log\_prob\_u = P(mixed\_unlabeled\_inputs)
\end{gathered}
$$

4.6 Loss

We use a semi-supervised loss defined as follows.

$$
\begin{gathered}
Lx, Lu, w = SemiLoss(log\_prob\_x, \\
mixed\_labeled\_targets, log\_prob\_u, \\
mixed\_unlabeled\_targets, epoch + \frac{batch}{n\_batches}) \\
TotalLoss = Lx + w * Lu
\end{gathered}
$$

where $SemiLoss$ implements,

$$
\begin{gathered}
prob\_u = SoftMax(log\_prob\_u) \\
Lx = CrossEntropyLoss(SoftMax(log\_prob\_x, mixed\_labeled\_targets) \\
Lu = SquaredErrorLoss(mixed\_unlabeled\_targets, prob\_u) \\
w = LinearRampup(epoch + \frac{batch}{n\_batches})
\end{gathered}
$$

Here, $LinearRampup$ is a function that clips the value of its argument to 1.

5 Experiments and Results

5.1 Implementation details

The “Wide ResNet-28” model is used in all the experiments. The hyperparameters from (1) are used for the experiments. An exponential moving average model is used for evaluation with a decay rate of 0.999. The model parameters with the best accuracy is preserved and saved to checkpoint in every ten epochs. All the experiments are run for 50 epochs due to restriction in computational resources.

5.2 Semi-Supervised Learning

In our experiments, we evaluate the results on the benchmark dataset CIFAR-10. The aim of semi-supervised methods like MixMatch is to learn with minimal amount of labeled data. The experiments are conducted with 500 and 4000 labelled images and by considering rest of the data as unlabeled. The train and validation sets are split distributing the labels proportionally into labeled, unlabeled and validation sets.

5.3 Baseline Methods

The MixUp discussed in section 4.4 was designed as a regulariser for supervised learning. This is used as one of the baseline models without the unlabeled data. A supervised model is trained with all the labeled images. The results of these two supervised models are compared with the performance of semi-supervised model with MixMatch.

5.4 Ablation Study

The MixMatch algorithm is a combination of different methods based on supervised learning extended to semi-supervised learning. To understand the impact of each of the component an ablation study is conducted with and without the addition of some of the components.

- K is the number of augmentations over which mean class distribution is calculated. K values 2 and 1(without the mean distribution) are explored.
- Changing the temperature sharpening values.
- MixUp is performed with only labeled examples.

5.5 Results

The error rates from all above experiments are displayed in Table 1. Although we did not achieve as high accuracy rates as the results in the MixMatch paper, there are similar patterns in the our results. The more number of training examples, the better accuracy the model achieved.

$K = 2$ gives lower error rates than $K = 1$ in all cases. The difference is larger in case of larger number of labeled training examples.

Our testing for MixMatch without temperature sharpening ($T = 1$) surprisingly gives better results than the one with ($T = 0.5$). This does not necessarily imply that sharpening with temperature has a negative effect but this might due to the random initialization of the model parameters. Our results are obtained from a single trial, therefore the final number can vary. Furthermore, we observed that the model is not very sensitive to varying temperature values.

Experiments	250 labels	4000 labels	45000 labels
Supervised learning	-	-	6.68
MixMatch with MixUp on labeled only	42.32	28.41	-
MixMatch, (K=1, T=0.5)	24.36	12.18	-
MixMatch, (K=2, T=0.2)	23.31	9.81	-
MixMatch, (K=2, T=0.5)	23.72	9.27	-
MixMatch, (K=2, T=0.75)	22.18	8.78	-
MixMatch without temperature sharpening (K=2, T=1)	20.91	9.22	-

Table 1: Ablations experiment results where the values are test error rates on CIFAR10

6 Conclusion

We successfully re-implemented MixMatch, investigated and replicated a few important experiments. We observed similar trends as obtained in the original paper for most of the experiments performed. Mixmatch with just 4000 labelled images performs with formidable accuracy which is comparable to supervised training. With varying number of augmentations, we find that a higher value of K=2 proved better. We also observed the effects of varying the temperature value in the sharpening step. We found that MixMatch can circumvent the data scarcity problem well by incorporating different semi-supervised learning improvements.

References

[1] MixMatch: A Holistic Approach to Semi-Supervised Learning David Berthelot and Nicholas Carlini and Ian Goodfellow and Nicolas Papernot and Avital Oliver and Colin Raffel, 2019, NeurIPS

[2] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-Supervised Learning. MIT Press, 2006.

[3] Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In Advances in Neural Information Processing Systems, pages 3235–3246, 2018.

[4] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017.