INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Semi-supervised Audio Classification with Consistency-Based Regularization

Kangkang Lu, Chuan-Sheng Foo, Kah Kuan Teh, Huy Dat Tran, Vijay Ramaseshan Chandrasekhar

Institute for Infocomm Research, A*STAR, Singapore

{luk, foo_chuan_sheng, teh_kah_kuan, hdtran, vijay}@i2r.a-star.edu.sg

Abstract

Consistency-based semi-supervised learning methods such as the Mean Teacher method are state-of-the-art on image datasets, but have yet to be applied to audio data. Such methods encourage model predictions to be consistent on perturbed input data. In this paper, we incorporate audio-specific perturbations into the Mean Teacher algorithm and demonstrate the effectiveness of the resulting method on audio classification tasks. Specifically, we perturb audio inputs by mixing in other environmental audio clips, and leverage other training examples as sources of noise. Experiments on the Google Speech Command Dataset and UrbanSound8K Dataset show that the method can achieve comparable performance to a purely supervised approach while using only a fraction of the labels.

Index Terms: audio classification, semi-supervised learning, data interpolation, data augmentation

1. Introduction

State-of-the-art deep learning methods typically require large amounts of labeled data to obtain high predictive performance. For audio classification in particular [1, 2], datasets need to include variations caused by the uncontrollable nature of audio sources, thus incurring increased data acquisition and more importantly, data labeling costs. Therefore, building a robust audio classification engine with limited labeled data is an important and practical problem that we are going to address in this paper.

Semi-supervised approaches that utilize small amounts of labeled data together with larger amounts of unlabeled data have previously been explored as a way to mitigate this data labeling burden, as unlabeled audio data is typically easier and cheaper to obtain in practice. This approach has been explored for audio event classification [3, 4] and speech recognition [5]. These works typically adopt a self-training approach; the iterative process starts by first training a classifier on the small amount of labeled data, then using it to predict labels for the unlabeled data. Confidently predicted labels (with associated data) are then included as part of the training data and the classifier is re-trained on this larger labelled dataset. The process is repeated for several iterations. An alternative approach applied to music instrument recognition first trains a Gaussian mixture model with the labeled data, then continues to improve the model using an iterative EM-algorithm on unlabeled samples [6].

Motivated by the recent successes of consistency-based semi-supervised learning methods in computer vision [7, 8, 9], we investigate their applicability to audio data. We first briefly review this class of methods that encourage model predictions to be consistent on unlabeled samples. Specifically, on unlabeled samples, the model is trained to have predictions that are consistent with those on perturbed inputs and different model parameters. This is achieved by including a consistency loss between the student and teacher models on the unlabeled samples, in addition to the usual classification loss for the labeled samples. In [7], this consistency loss is computed between a noisy student model and a clean teacher model, where random noise is added to the latent features in the student model. During model training, the teacher model then tries to denoise the corrupted latent features and minimize the difference between the features in the student and teacher models. Laine and Aila [8] proposed the π model where the student and teacher models share the same weights but use different data augmentation techniques and dropout. They also introduced temporal ensembling that uses the exponential moving average (EMA) of the student model predictions as the outputs of teacher model. Instead of a prediction ensemble, Tarvainen and Valpola [9] used the EMA of the student model's weights as the weights of teacher model and achieved state-of-the-art performance on image datasets. However, despite their success on image datasets, these methods have not been applied to audio data to the best of our knowledge.

The specific perturbations applied to the inputs during training play an important role in consistency-based methods; in some sense they enforce smoothness of the classifier along the data manifold. However, perturbations commonly used for images may not be suitable for audio data. Moreover, some perturbations also are specific to audio data, for instance, mixing with environmental noise. There is a special category of audio in the Google Speech Commands Dataset that is background noise (e.g., "doing the dishes", "exercise bike", "running tap"). This background noise was collected to enable the training of more robust classifiers that can identify a command in spite of the noise.

Besides adding environmental noise directly to the training samples, we can also mix different samples together as a form of perturbation. Zhang et al. [10] proposed to train a neural network with convex combination of pairs of samples. This mixup method has been shown to be an effective and simple way of data augmentation and can improve the generalization of networks. Verma et al., [11] further extended mixup to semi-supervised learning. The consistency loss is computed between predictions on interpolated samples from the student model and the interpolated predictions from teacher model. Instead of using small perturbations like noise or data augmentation, the interpolated samples mostly lie in the low-density regions along the decision boundaries. Adding small perturbations to these samples may easily push them to the other side of the decision boundary. By training on these samples, they achieved state-of-the-art semi-supervised learning (SSL) performance.

In this work, we apply Mean Teacher to two different audio datasets – the Google Speech Commands Dataset [12] and UrbanSound8K Dataset [13], and show that the consistency-based SSL methods also perform well on audio data. We show that using environment noise as perturbation further improves performance. We also explore the effect of mixing samples and find that encouraging interpolated predictions from the student

Copyright © 2019 ISCA

http://dx.doi.org/10.21437/Interspeech.2019-1231

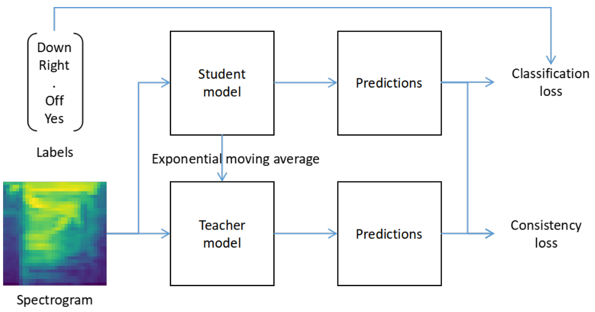

Figure 1: *Architecture of Mean Teacher model.*

and teacher models to be consistent improves performance. Our work establishes consistency-based methods as a strong baseline for semi-supervised audio classification.

2. Methods

In this section, we describe how Mean Teacher, environmental noise and sample mixing are applied to audio datasets.

2.1. Mean Teacher method

The Mean Teacher method attempts to enforce classifier smoothness in the face of small perturbations on the data and model parameters. In the method, we have a student and teacher model, both assumed to be neural networks that share the same architecture (f). The weights of the teacher model (W) are the EMA of the student model's weights (w). Formally, the teacher weights at step t are computed as

$$W_t = \alpha * W_{t-1} + (1 - \alpha) * w_t, \tag{1}$$

where α is the smoothing constant that effectively controls the averaging time period. Given a labeled set of samples $D_l = \{(x_i, y_i)\}_{i=1}^{N_L}$ and an unlabeled set $D_u = \{x_i\}_{i=1}^{N_U}$, the classification loss function is the usual cross-entropy (CE) loss between predictions of labeled samples and corresponding ground truth $L_{cls} = \sum_{(x,y)\in D_l} CE(f(x; w), y)$. The consistency loss L_{con} is defined as follows, between predictions on the same sample from the student and teacher models

$$L_{con} = \sum_{x\in D_c} \|f(x; w) - f(x; W)\|^2 . \tag{2}$$

where $D_c = \{x \mid (x, y) \in D_l\} \cup D_u$, so that this is computed over samples from both the labeled and unlabeled data. The total loss is the sum of classification and consistency losses

$$L_{total} = L_{cls} + \lambda L_{con}, \tag{3}$$

where λ is a parameter controlling the importance of consistency loss. When applying Mean Teacher, the consistency loss is computed on both labeled and unlabeled data.

A schematic of the method is shown in Figure 1. During training, each batch contains both labeled and unlabeled samples. We feed these batches of data into both the student and teacher models but with different perturbations; we discuss these in detail in Section 2.2. These perturbations may include translations, addition of Gaussian noise, flips, and dropout. The consistency loss is computed as the mean-square distance between the predictions from the student and teacher models. The classification loss is evaluated as the cross-entropy loss between the softmax output of the student model and the target labels. During optimization, the gradients are not backpropagated to the teacher model, instead, only the weights of the student model are updated.

2.2. Perturbations on audio data

We modify the perturbation settings in [9], and the details are as following.

2.2.1. *Time and frequency shifts*

Random time and frequency shifts are natural perturbations on audio data. Translated to spectrogram images, these perturbations correspond to horizontal translations for time shifts and vertical translations for frequency shifts.

2.2.2. *Gaussian noise*

Additive Gaussian noise is also used as a possible perturbation as there might be noise when recording and transforming the audio data.

2.2.3. *Environment noise as perturbations*

In the Google Speech Commands dataset, we have six classes of background noise, one sample per class. Owing to the linearity of the spectrogram, we can simply add the spectrogram of the audio sample with that of the background noise to mimic the situation when the command audio is recorded with the noise. Given the original spectrogram as S_{ori} and the environment noise spectrogram as S_{env}, the final sample (S_{in}) fed into the network is:

$$S_{in} = (1 - \beta) * S_{ori} + \beta * S_{env}, \tag{4}$$

where β is used to control the environment noise level. Compared to adding random Gaussian noise, real environmental noise is a more realistic form of noise, and can potentially enable classifiers to be robust to such noise in the input. As there are only limited number of noise samples in the speech commands dataset, we also use the unrelated UrbanSounds8K urban audio dataset to enrich the quantity and diversity of environment audio data.

2.2.4. *Consistency regularization with mixed samples*

Another possible source of perturbations are the original samples themselves. Mixed samples can be regarded as the case when different commands are given at the same time. Given

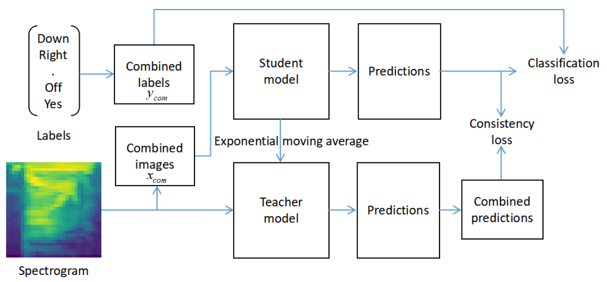

Figure 2: *Architecture of Mean Teacher model with mixed samples as input.*

two samples as (x_a, x_b) and their corresponding labels (y_a, y_b) (one-hot encoding), the generated pair of samples are:

$$x_{mix} = (1 - \gamma) * x_a + \gamma * x_b \tag{5}$$

$$y_{mix} = (1 - \gamma) * y_a + \gamma * y_b \tag{6}$$

The classification loss will be computed as:

$$L_{cls\_mix} = CE(f(x_{mix}; w), y_{mix}) \tag{7}$$

For mixed unlabeled data (X_a, X_b), their predictions from the student and teacher models are:

$$P_{stu} = f(X_{mix}; w) \tag{8}$$

$$P_{tch} = (1 - \gamma) * f(X_a; W) + \gamma * f(X_b; W), \tag{9}$$

where $X_{mix} = (1 - \gamma) * X_a + \gamma * X_b$. And the consistency loss is:

$$L_{con\_mix} = ||P_{stu} - P_{tch}||^2 \tag{10}$$

In the case where we utilized mixed samples as perturbations, the training workflow is different as shown in Figure 2. The samples are mixed before being fed into the network. For mixed samples with corresponding combined labels, they contribute to the classification loss. The predictions of original samples from teacher model are combined as P_{tch}. The consistency loss is the mean-square distance between combined predictions from teacher model and predictions of mixed samples from student model as noted in Equation 10.

3. Experiments

3.1. Datasets

We perform experiments on two datasets: the Google Speech Commands Dataset [12] and the UrbanSound8K Dataset [13]. In both cases, during training, the entire training set is considered as the unlabeled dataset, while labeled samples are randomly selected from the training set in each run.

Google Speech Commands Dataset: This dataset includes one-second long utterances of 30 different speech commands such as "Yes", "No", "Up" and "Down", spoken by many different people. We split the dataset into a 57886-sample training set and a 6835-sample testing set according to the official partition script. We convert the waveform audio into 32×32 mel-spectrogram images for training.

UrbanSound8K Dataset: This is a collection of short audio clips (1 to 4 seconds long) from 10 different classes collected from urban acoustic environments. Classes include sounds of car horns, dogs barking, and jackhammers. There are 8732 audio samples in total; we used the training set including 5434 labeled samples from the version hosted on Kaggle[1]. We split the labeled data into training and test sets. We transform the audio into mel-spectrogram images with the same sampling rate. As the audio clips have varying lengths, the images are in different shapes. To facilitate training, we zero-pad them to the same size 32×128.

3.2. Experimental setup

For our classifiers (both the supervised baseline and in Mean Teacher), we use the same 13-layer convolutional neural network proposed in [9]. Briefly, the network consists of 3 blocks of 3 convolution layers followed by a single pooling layer; the last block is then connected to a fully-connected layer that computes the prediction logits. The network also includes a dropout layer after each of the first two pooling layers. Batch normalization and weight normalization are applied to the convolution and fully-connected layers. The model is trained using the ADAM optimizer with a batch size of 100 for 80000 iterations on Google Speech Commands Dataset and 8000 iterations on UrbanSound8K Dataset.

During the first few epochs of training, the optimization relies more on the labeled data and classification loss. As the model starts to converge, the consistency loss plays a more important role. As described in [9], a ramp-up on λ in Equation 3 is applied to adapt the importance of the consistency loss as training progresses. Specifically, both the learning rate and λ ramp up to their maximum value in the first 40000 or 4000 steps. The learning rate also ramps down to 0 in the last 25000 or 2500 steps for better convergence. For details on the ramp-up and ramp-down procedure, please refer to the experimental setup section in the appendix of [9]. The maximum learning rate is set to 0.003. All experiments are repeated five times with the same series of random seeds, and we report the average classification accuracy along with standard deviations.

3.3. Evaluation of the Mean Teacher method

We first compare the performance of the semi-supervised Mean Teacher method to a purely supervised convolutional neural network with the same architecture (see Section 3.2) that only uses the labeled data during training. For these experiments we only include random time and frequency shifts as well as Gaussian noise as perturbations. We report classification accuracy on the Google Speech Commands Dataset (Table 1) and UrbanSound8K Dataset (Table 2).

On the Google Speech Commands Dataset, Mean Teacher outperforms the supervised baseline when not all labeled data is used. The performance gain increases as less labeled data is used – from 5.41% when only 1% of labels are used to 0.54% when 25% of labels are used in training. Mean Teacher achieves comparable performance to the supervised baseline using all labels, with only 25% (even 10%) of the labels.

The UrbanSound8K dataset is more challenging as it has fewer training samples. We also observe that Mean Teacher achieves performance gains over the supervised baseline across the board. Here, gains are significant even when almost a third of the samples are labeled (3.02% gain when 1500 labels are used). With only 12% of the labels (600 labels), Mean Teacher achieves a large improvement in accuracy of 8.76% over the supervised baseline. On this dataset it is not possible to achieve comparable accuracy to a supervised baseline using all labels with less labeled data. However, Mean Teacher manages to obtain 85.93% accuracy using only 30% of the labels.

Table 1: *Classification Accuracy (%) of Mean Teacher on the Google Speech Commands Dataset (57886 training samples)*

Labels	Supervised	Mean Teacher
600	87.04 ± 0.62	92.45 ± 0.39
3000	93.21 ± 0.24	95.08 ± 0.19
6000	94.46 ± 0.12	95.73 ± 0.14
15000	95.64 ± 0.13	96.18 ± 0.04
57886	96.62 ± 0.07	96.65 ± 0.06

[1]https://www.kaggle.com/pavansanagapati/urban-sound-classification

Table 2: *Classification Accuracy (%) of Mean Teacher on UrbanSound8K (4892 training samples)*

Labels	Supervised	Mean Teacher
60	34.62 ± 0.19	33.70 ± 6.26
300	60.04 ± 3.07	65.27 ± 3.41
600	66.41 ± 4.93	75.17 ± 1.52
1500	82.91 ± 1.92	85.93 ± 0.98
4892	93.37 ± 0.71	93.41 ± 0.32

3.4. Incorporating environment noise as perturbations

3.4.1. Background noise from Google Speech Dataset

To add the environment noise, before each iteration, we randomly pick one category of environment noise, crop it to a 32×32 patch and add it to the training batch. We do a grid search for best β in Equation 4 in range of $[0.1, 0.5]$ with step size of 0.1. A randomly selected β in range of $[0.1, 0.5]$ is also evaluated. After validating on 600 labeled samples, the best performance is achieved with $\beta = 0.1$. From Table 3, we can see that, with this new perturbation, the accuracy is almost the same. This lack of significant improvement could be due to the lack of diversity in the background noise samples (6 samples, 40 seconds each).

3.4.2. Urban noise from UrbanSound8K

To see if a more diverse set of environment noise could provide performance benefits, we also consider the Urban Sound Dataset [13] as a source of noise, as it spans more classes and includes more samples. We use the same noise setting as found on the Google Speech Commands Dataset in our experiments.

From the results shown in Table 3, we can see that using urban noise yields a clear performance improvement. Compared with using only the six noise samples in the Google Speech Commands Dataset, urban noise is much more effective. With 600 labels, while the command noise barely affect the accuracy, incorporating urban noise provides a larger boost of 1.86%. With only 15000 (around 25%) labels, performance surpasses that of the supervised baseline using all the labels (96.62%). We conclude that the quantity and variety of the noise greatly influences classification performance.

However, these results also raise the question: is the improvement due to the new noise perturbation or simply due to a data augmentation effect. To clarify this, we apply environmental noise to supervised training as a way of data augmentation. The results are shown in Table 4. We can see that, except the improvement in the case where 600 labels are used, there is no clear improvement when using urban noise as part of a data augmentation approach. These results indicate the effectiveness of introducing environmental noise as a new form of perturbation as part of the Mean Teacher method.

Table 3: *Classification Accuracy (%) with environmental noise on Google Speech Commands Dataset*

Labels	Mean Teacher	MT+noise	MT+urban noise
600	92.45 ± 0.39	92.44 ± 0.27	94.31 ± 0.27
3000	95.08 ± 0.19	95.07 ± 0.13	95.91 ± 0.30
6000	95.73 ± 0.14	95.76 ± 0.14	96.35 ± 0.16
15000	96.18 ± 0.04	96.18 ± 0.10	97.01 ± 0.10
57886	96.65 ± 0.06	96.68 ± 0.01	97.65 ± 0.07

Table 4: *Classification Accuracy (%) of using environmental noise as data augmentation on Google Speech Commands Dataset*

Labels	Supervised	Urban noise augmentation
600	87.04 ± 0.62	87.26 ± 0.44
3000	93.21 ± 0.24	93.22 ± 0.25
6000	94.46 ± 0.12	94.60 ± 0.16
15000	95.64 ± 0.13	95.74 ± 0.12
57886	96.62 ± 0.07	96.63 ± 0.06

Table 5: *Classification Accuracy (%) incorporating sample mixing on Google Speech Commands Dataset*

Labels	Mean Teacher	MT+sample mixing
600	92.45 ± 0.39	94.38 ± 0.23
3000	95.08 ± 0.19	95.78 ± 0.18
6000	95.73 ± 0.14	96.32 ± 0.16
15000	96.18 ± 0.04	96.63 ± 0.06
57886	96.65 ± 0.06	97.17 ± 0.04

3.5. Mixing samples as perturbations

Finally, we evaluate the effect of mixing samples as perturbations. We do not include environment noise in these experiments to isolate the effects of the mixing. Results are shown in Table 5. With mixed samples, instead of introducing more perturbations, we are augmenting the dataset with more training samples near the decision boundary of different target labels. The performance improvements are small (0.5-2%), similar to when we use urban noise as perturbations. We note that the greatest improvement was achieved when only 1% of the labels were used (600 labels), where it provides a 1.93% boost in accuracy over the vanilla Mean Teacher method.

4. Conclusion

In this paper, we demonstrate the effectiveness of the Mean Teacher method for audio classification on two very different audio classification datasets – speech and urban audio. We introduce the addition of environment noise as a new effective perturbation as part of the Mean Teacher method, showing improvements using background noise from the UrbanSound8K dataset. Our results show the importance of using diverse collections of noise as perturbations and that the improvement cannot simply be achieved using a similar data augmentation strategy. Finally, we use mixed samples and labels as a source of perturbations. In all cases, we show that Mean Teacher can achieve comparable results to fully supervised training using only 25% (or even 10%) of the labels. We also observe significant accuracy boosts over the supervised baseline when only 1% of the labels are used. These promising results indicate the feasibility of applying the Mean Teacher method, environment noise and sample mixing on audio classification tasks. We believe that these methods can be adapted and improved for other audio datasets and tasks, and leave this as an interesting direction for future work.

5. References

[1] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2017.

[2] A. Jansen, J. F. Gemmeke, D. P. Ellis, X. Liu, W. Lawrence, and D. Freedman, "Large-scale audio event discovery in one million YouTube videos," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2017.

[3] Z. Zhang and B. Schuller, "Semi-supervised learning helps in sound event classification," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2012.

[4] W. Han, E. Coutinho, H. Ruan, H. Li, B. Schuller, X. Yu, and X. Zhu, "Semi-supervised active learning for sound classification in hybrid learning environments," *PLoS ONE*, 2016.

[5] D. Hakkani-Tur, G. Tur, M. Rahim, and G. Riccardi, "Unsupervised and active learning in automatic speech recognition for call classification," 2004.

[6] A. Diment, T. Heittola, and T. Virtanen, "Semi-supervised learning for musical instrument recognition," in *21st European Signal Processing Conference (EUSIPCO 2013)*, Sep. 2013, pp. 1–5.

[7] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *NIPS*, 2015.

[8] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," *International Conference on Learning Representations (ICLR)*, 2017.

[9] A. Tarvainen and H. Valpola, "Weight-averaged consistency targets improve semi-supervised deep learning results," in *NIPS*, 2017.

[10] Y. N. D. D. L.-P. Hongyi Zhang, Moustapha Cisse, "mixup: Beyond empirical risk minimization," *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=r1Ddp1-Rb

[11] V. Verma, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz, "Interpolation Consistency Training for Semi-Supervised Learning," *arXiv e-prints*, p. arXiv:1903.03825, Mar 2019.

[12] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," 2018.

[13] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *22nd ACM International Conference on Multimedia (ACM-MM'14)*, Orlando, FL, USA, Nov. 2014, pp. 1041–1044.