INN: A Method Identifying Clean-annotated Samples via Consistency Effect in Deep Neural Networks

D.Kim [1] and Y.Choi [2] and K.Kim [2] and Y.Kim [3]

Speaker : Dongha Kim

[1]Department of Statistics, Sungshin Women's University, South Korea

[2]Department of Statistics, Seoul National University, South Korea

[3]College of Data Science, Seoul National University, South Korea

May 27-28, 2021

Outline

1 Introduction

2 Our proposed method

3 Empirical analysis

4 References

Outline

1 Introduction

2 Our proposed method

3 Empirical analysis

4 References

Noisy label problem

- A lot of training data is needed to train classification deep networks.
- In many domains related to the AI,
  - **labeling** procedure can be done only **manually**→**high cost**.
- Alternative plans: internet search engines or hashtags.
  - Easy to collect massive labels but relatively **inaccurate**.

✓ **Noisy label problem**: classification tasks with **noisy labels**.

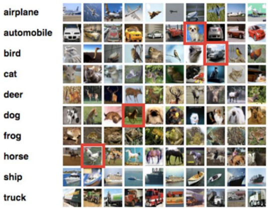


Figure: Data with noisy labels.

Noisy label problem

Bottleneck

- Deep learning models have large model complexity.
- Easy to over-fit training data, even data with noisy labels.

Goal

- To identify clean labeled data from training data
- To train deep networks robust to noisy labeled data.

Conventional solving strategy

Memorization effect (ME) [2, 3]

1. Deep networks memorize data **clean-labeled data first**,
2. and memorize noisy-labeled ones after.

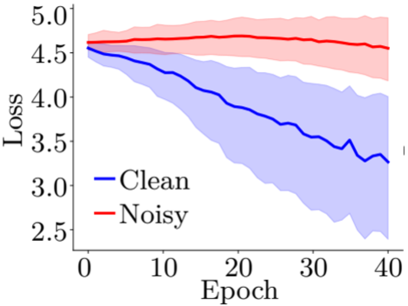


Figure: Memorization effect

Conventional solving strategy

- To train deep networks of high-performance, we need to measure the score of cleanness for each training sample in advance.
- Many existing learning methods dealing with noisy labels rely on the **small-loss strategy** to identify clean-labeled data, utilizing the ME ([7] and references therein).

- Small-loss strategy:
  1. Train a deep network for a few training epochs.
  2. Calculate per-sample loss values.
  3. Choose small-loss-valued samples and treat them as clean-labeled data.

Limitations of the small-loss strategy

- The However, the small-loss strategy has several weaknesses.

1. The performance of the small-loss strategy heavily depends on the selection of the training epoch.
2. When the labels are imbalanced or heavily polluted, the small-loss strategy may fail to separate clean-labeled data.

Our contributions

- We find a new observation, called the **consistency effect** (CE), which helps to identify clean-labeled data well.
- Based on the finding, we propose a new algorithm, called the integration with the nearest neighborhoods (INN), to identify clean-labeled samples.
- We demonstrate the INN overcomes the limitations of the small-loss method successfully.
- Combining ours with other learning frameworks handling noisy labels, we success to enhance the performance of prediction networks.

Outline

1 Introduction

2 Our proposed method

3 Empirical analysis

4 References

Notations

- $f(\mathbf{x};\theta) : \mathcal{X} \to \mathcal{S}^K$: K-class prediction model
  - $f_k(\mathbf{x};\theta) = p(Y = k|\mathbf{x};\theta)$, $k = 1, \ldots, K$
  - $\mathcal{S}^K$: K-dimensional simplex
  - θ: parameter
- $\mathcal{D}^{tr} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$: training data
  - $\mathbf{x} \in \mathcal{X}$: input
  - $y \in \{1, \ldots, K\}$: observed label
  - $y^* \in \{1, \ldots, K\}$: ground-truth label (not observed)
- $\mathcal{C}^{tr} = \{(\mathbf{x}, y) \in \mathcal{D}^{tr} : y = y^*\}$
- $\mathcal{N}^{tr} = \{(\mathbf{x}, y) \in \mathcal{D}^{tr} : y \neq y^*\}$

- Our goal is to estimate the clean labeled set $\mathcal{C}^{tr}$ accurately.

Motivation: Consistency effect (CE)

- For a given sample $(\mathbf{x}, y)$, let $\mathbf{x}^m = \frac{\mathbf{x}+\tilde{\mathbf{x}}}{2}$, where $\tilde{\mathbf{x}}$ is the nearest neighbor training input of $\mathbf{x}$ with a certain dissimilarity.
- We analyze how the values $f_y(\mathbf{x};\theta)$ and $f_y(\mathbf{x}^m;\theta)$ behave with respect to the label cleanness of $\mathbf{x}$.
- At each training epoch, we calculate the followings and compare them.

$$
\begin{aligned}
E_{\text{cor}} &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{C}^{\text{tr}}}\left[f_y(\mathbf{x})\right], \\
E_{\text{inc}} &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{N}^{\text{tr}}}\left[f_y(\mathbf{x})\right], \\
E^{\text{m}}_{\text{cor}} &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{C}^{\text{tr}}}\left[f_y(\mathbf{x}^{\text{m}})\right], \\
E^{\text{m}}_{\text{inc}} &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{N}^{\text{tr}}}\left[f_y(\mathbf{x}^{\text{m}})\right],
\end{aligned}
$$

- E_{cor} and E_{inc}: average of prediction values on the training data.
- $E^{\text{m}}_{\text{cor}}$ and $E^{\text{m}}_{\text{inc}}$: average of prediction values on the **neighbor region** of the training data.

Motivation: Consistency effect (CE)

- $E_{\text{cor}} - E_{\text{inc}}$ decreases as the training epoch increases (**ME**).
- However, $E^{\text{m}}_{\text{cor}} - E^{\text{m}}_{\text{inc}}$ does not vanish (**New observation**).
- The prediction values at a **neighbor region** of each sample are informative to identify clean labeled data.
  $\rightarrow$ **Consistency effect (CE)**

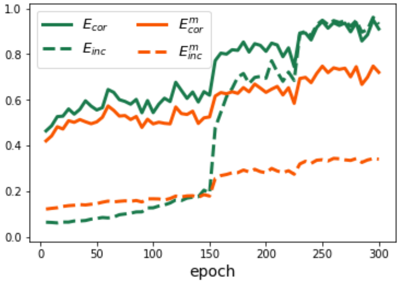


Figure: The ME and CE

INN

- $f(\cdot;\hat{\theta})$: an estimated model.
    - We found that using the MixUp loss function is the best.
- The INN score of $\mathbf{z} = (\mathbf{x}, y) \in \mathcal{D}^{tr}$ is

$$I(\mathbf{z}) = f_y\left(\frac{\mathbf{x}+\tilde{\mathbf{x}}}{2};\hat{\theta}\right),$$

  where $\tilde{\mathbf{x}}$ is the nearest training input of $\mathbf{x}$ with a certain dissimilarity.
    - The dissimilarity measure we choose is the Euclidean distance on the feature space of the function.
- The larger the score $I(\mathbf{x})$, the more we could regard $\mathbf{z}$ as being cleanly labeled.

INN

✓ Modification 1.

- We found that the CE occurs on a wide range between $\mathbf{x}$ and $\tilde{\mathbf{x}}$.
- To exploit the CE to the best, we modify the INN score by integrating the prediction function along from $\mathbf{x}$ to $\tilde{\mathbf{x}}$:

$$I(\mathbf{z}) = \int_0^1 f_y\left(\alpha\mathbf{x} + (1-\alpha)\tilde{\mathbf{x}}; \hat{\theta}\right),$$

INN

✓ Modification 2.

- Using multiple neighbors helps identify clean labeled data more accurately:

$$I(\mathbf{z}) = \frac{1}{L}\sum_{l=1}^{L}\int_0^1 f_y\left(\alpha\mathbf{x} + (1-\alpha)\tilde{\mathbf{x}}_l;\hat{\theta}\right),$$

where $\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_L$ are the L nearest neighbors of $\mathbf{x}$.

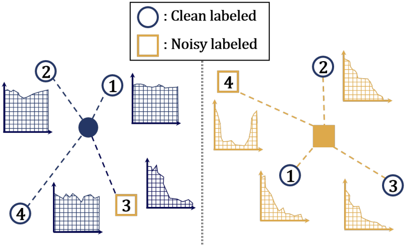


Figure: An illustration of calculating the INN score

Application of INN

- The INN scores can improve the performance of existing deep network learning frameworks solving the noisy label problem.
- In this study, we construct deep networks of high-performance by combining ours to the DivideMix method [4], one of the state-of-the-art methods.
- We make this by simply replacing the small-loss strategy part in the DivideMix with the INN method.

Outline

1 Introduction

2 Our proposed method

3 Empirical analysis

4 References

Performance test of INN

- Data sets:
  - CIFAR10&100.
- Architectures:
  - PreActResNet18.
- Optimizers:
  - SGD

Stability and superiority of INN

- We use CIFAR10&100, and we artificially impose label noise to the data sets with various noise rates from 10% to 30%.
- At each 50 training epochs, We measure the clean/noisy classification AUC value on the training data set.
- Baselines are the small-loss methods with two loss functions (CE and CE+NE).

Stability and superiority of INN

- For all cases, the INN method constantly outperforms other baselines for all considering training epochs.
- That means, the INN identifies clean-labeled data stably and accurately.

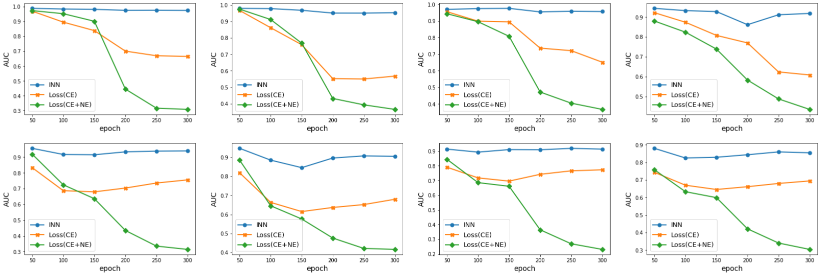


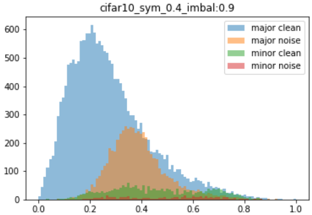
Figure: Comparison of AUC values for clean/noisy sample classification between the *INN* and other methods.

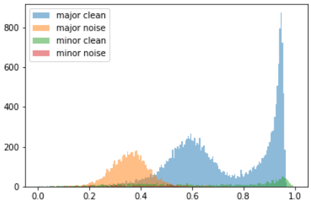
Imbalanced data analysis

- From CIFAR10, we randomly split ten classes into two groups so that each group has five.
- The training data set is composed of all images in the first group and 10% randomly sampled images from the second group.
- For each sample, we give noise to its label with a probability of 0.4.
- We compare the AUC of clean/noisy classification of the INN with that of the small-loss method.

Imbalanced data analysis

- The small-loss method fails to identify the minor samples, while the INN does successfully.





Heavily contaminated data analysis

- We train deep prediction models by applying the INN to the DivideMix.
- Then, we calculate test accuracies and compare them with those of other baselines learning frameworks.
- Our method enhances the state-of-the-art performances with large margins.

Data set	CIFAR10	CIFAR100	
Noise type	Symm.	Symm.	Asym.
Methods / Noise rate	90%	80%	40%
Cross-Entropy	42.7	19.9	42.7
Forward T [6]	42.9	19.19	-
M-correction [1]	69.1	48.2	-
MLNT [5]	58.7	42.4	-
DivideMix [4]	76.0	60.2	59.26
INN-DivideMix	**81.51**	**61.81**	**63.02**

Summary

- We proposed a new approach, called INN, to identify clean labeled samples from training data with noisy labels.
- The INN is based on the new finding called the consistency effect that discrepancies of predictions at neighbor regions are helpful to identify clean-labeled data well.
- We empirically demonstrated that the INN is stable and powerful even when the labels are imbalanced or heavily contaminated.


thank you
danke
謝謝
ngiyabonga
teşekkür ederim
gracias
tapadh leat
спасибо
dank je
bedankt
dziękuje
obrigado
sukriya
kop khun krap
go raibh maith agat
mochchakkeram
arigatō
dakujem
grazie
terima kasih
merci
감사합니다
мерси
хвала
рахмат
Баярлалаа
maururu
hvala
sagolun
takk
taiku
ευχαριστώ
ありがとう
mersi
barka
welalin
tack
misaotra
matondo
paldies
grazzi
mahalo
blagodaram
vinaka
faafetai lava
kiitos
dankie
dhanyavad
nanni
nandri
gracie
bayarlalaa
koszonom
enkosi
sobodi
dekuji
mesi
didi madloba
najis tuke
kam sah hamnida
rahmat
chnorakaloutioun
gratias ago
gracies
sulpay
tanemirt
rahmet
diolch
dhanyavadagalu
shukriya
merce
xiexie
trugarez
mamnun
asante
manana
obrigada
murakoze
tenki
djiere dieuf
tau
dankon
aciū
akun
chokrane
дякую
kia ora
spas

Outline

1 Introduction

2 Our proposed method

3 Empirical analysis

4 References

References I

[1] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness. Unsupervised label noise modeling and loss correction. In *36th International Conference on Machine Learning*, pages 312–321, 2019.

[2] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pages 233–242. PMLR, 2017.

[3] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pages 2304–2313. PMLR, 2018.

[4] J. Li, R. Socher, and S. C. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *8th International Conference on Learning Representations*, 2020.

[5] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli. Learning to learn from noisy labeled data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5051–5059, 2019.

References II

[6] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: A loss correction approach. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2233–2241, 2017.

[7] H. Song, M. Kim, D. Park, and J.-G. Lee. Learning from noisy labels with deep neural networks: A survey. *arXiv preprint arXiv:2007.08199*, 2020.