

Representer Point Selection for Explaining Deep Neural Networks

Chih-Kuan Yeh, Joon Sik Kim,
Ian E.H. Yen , Pradeep Ravikumar
CMU

NIPS-2018

what's about

for a given test point prediction, select **representer** points in
the training set which can explain the predictions

Representer point

Φ : model

Θ : parameter of model

x_t : testing data

L: loss

f_i : feature of training data

f_t : feature of testing data

Prediction of test data

$$\Phi(\mathbf{x}_t, \Theta^*) = \sum_i^n k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i),$$

$$\alpha_i = \frac{1}{-2\lambda n} \frac{\partial L(\mathbf{x}_i, \mathbf{y}_i, \Theta)}{\partial \Phi(\mathbf{x}_i, \Theta)}$$

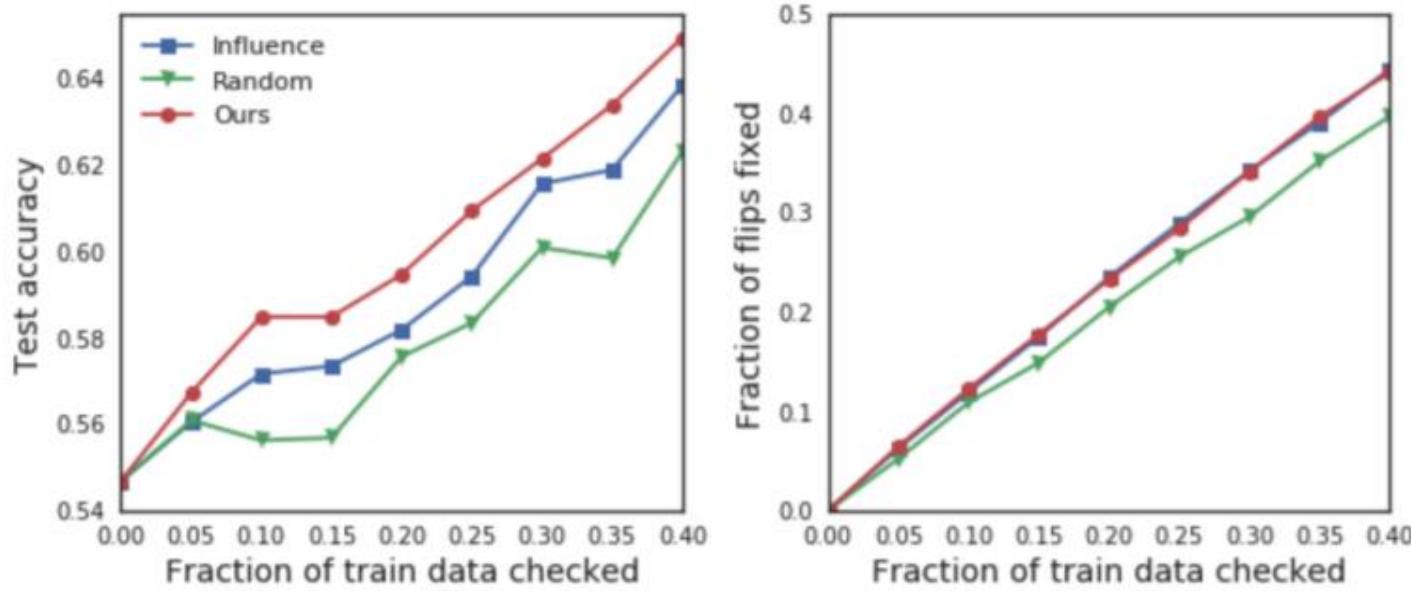
$$k(\mathbf{x}_t, \mathbf{x}_i, \alpha_i) = \underbrace{\alpha_i \mathbf{f}_i^T \mathbf{f}_t}_{\text{Feature similarity}}$$

Decomposition by theorem

α_i : importance of i-th data on the parameter

$k(x_t, x_i, \alpha_i)$: representer value for x_i given x_t , k_j represents x_i 's impact on class j

Dataset Debugging



40 percent of the data has the label flipped.
find mislabeled data, retrain.

Excitatory(positive) and Inhibitory(negative) Examples

test id3092
grizzly bear predicted as
grizzly bear

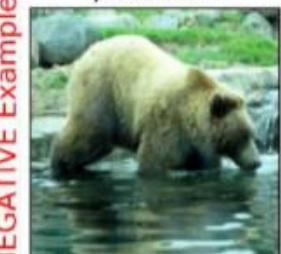


train id13033
grizzly bear predicted as
grizzly bear



POSITIVE Example

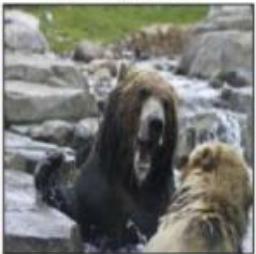
train id21249
polar bear predicted as
polar bear



NEGATIVE Example

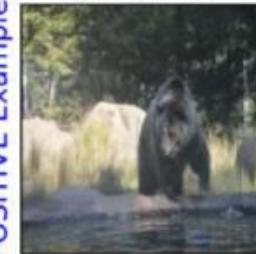
Ours

train id12728
grizzly bear predicted as
grizzly bear



POSITIVE Example

train id12742
grizzly bear predicted as
grizzly bear



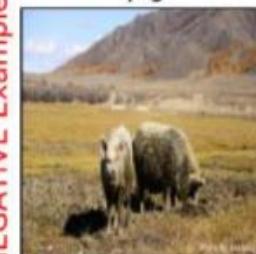
POSITIVE Example

train id1228
beaver predicted as
beaver



NEGATIVE Example

train id20730
pig predicted as
pig



NEGATIVE Example

train id12866
grizzly bear predicted as
grizzly bear



POSITIVE Example

train id21249
polar bear predicted as
polar bear



NEGATIVE Example

Influence Function

train id13033
grizzly bear predicted as
grizzly bear



POSITIVE Example

train id13155
grizzly bear predicted as
grizzly bear



POSITIVE Example

train id2556
buffalo predicted as
buffalo



NEGATIVE Example

Grizzly bear:
灰熊

beaver: 河狸

Positive: similar and same label

Negative: similar but different label

Excitatory and Inhibitory Examples

test id5727
rhinoceros predicted as
rhinoceros



犀牛

train id23304
rhinoceros predicted as
rhinoceros



POSITIVE Example

train id8471
elephant predicted as
elephant



NEGATIVE Example

Ours

train id23687
rhinoceros predicted as
rhinoceros



POSITIVE Example

train id29490
zebra predicted as
zebra



NEGATIVE Example

train id23336
rhinoceros predicted as
rhinoceros



POSITIVE Example

train id8518
elephant predicted as
elephant



NEGATIVE Example

train id19539
ox predicted as
cow



POSITIVE Example

train id19293
ox predicted as
ox



NEGATIVE Example

Influence Function

train id19684
ox predicted as
cow



POSITIVE Example

train id4642
cow predicted as
ox



NEGATIVE Example

train id19525
ox predicted as
cow



POSITIVE Example

train id19611
ox predicted as
ox



NEGATIVE Example

0x: 牛

Misclassified Examples

test id7
predicted as deer
true label is antelope



train id29372
predicted as zebra
true label is zebra



train id688
predicted as deer
true label is antelope



train id8090
predicted as elephant
true label is elephant



train id29208
predicted as zebra
true label is zebra



羚羊被标记为鹿

All of these pictures have deer,
but labeled different class

Example selection for dictionary learning

Tomoki Tsuchida ,Garrison W. Cottrell
UCSD

ICLR-2015

Dictionary Learning

Given the input dataset $X = [x_1 \dots x_K]$, $x_i \in R^d$, we wish to find a dictionary $D \in R^{d \times n}$: $D = [d_1, d_2 \dots d_n]$, and a representation $R = [r_1, r_2 \dots r_k]$, $r_i \in R^n$ such that both $\|x_i - Dr_i\|_F^2$ is minimized and the representations r_i are sparse enough.

This can be formulated as the following optimization problem:

$$\operatorname{argmin}_{\mathbf{D} \in \mathcal{C}, r_i \in \mathbb{R}^n} \sum_{i=1}^K \|x_i - \mathbf{D}r_i\|_2^2 + \lambda \|r_i\|_0$$

We want to use less data to learn dictionary

Score function

$$\text{Err: } g_j(r^{(i)}, x^{(i)}) = \left\| \widehat{D}r^{(i)} - x^{(i)} \right\|_1$$

$$\text{Grad: } g_j(r^{(i)}, x^{(i)}) = \left\| \widehat{D}r^{(i)} - x^{(i)} \right\|_1 * r_j^{(i)}$$

$$\text{SNR: } g_j(r^{(i)}, x^{(i)}) = \frac{\|x^i\|_2^2}{\|\widehat{D}r^i - x^i\|_2^2} * r_j^{(i)}, \text{ 信噪比}$$

$$\text{SUN: } g_j(r^{(i)}, x^{(i)}) = r_j^{(i)}$$

$$\text{SalMap: } g_j(r^{(i)}, x^{(i)}) = \text{SaliencyMap}(x^{(i)})$$



From other papers

Selector function

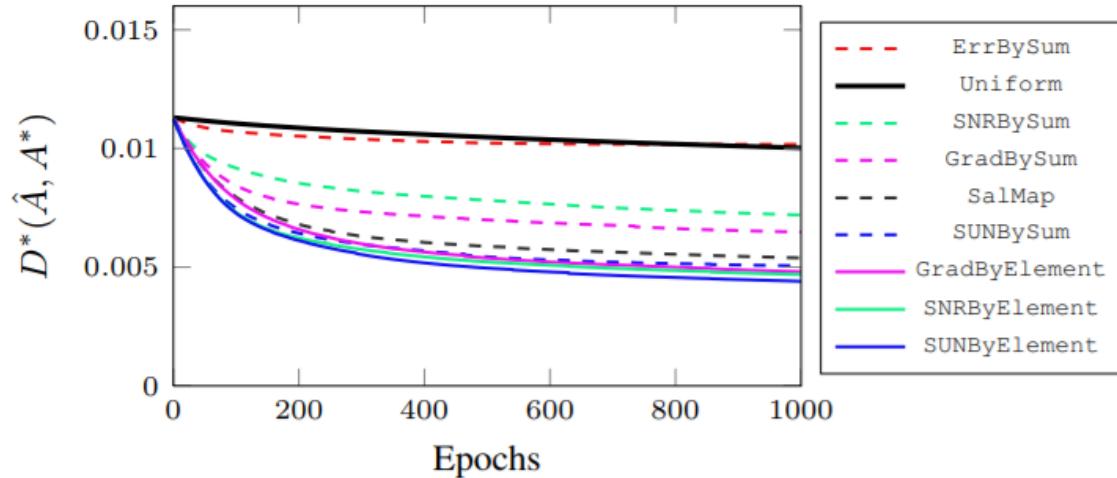
BySum: $f_{sel} = \text{top } n \text{ elements of } \sum_{j=1}^n g_j^i$

计算每个element的分值，最后加总，选出总分最高的样本

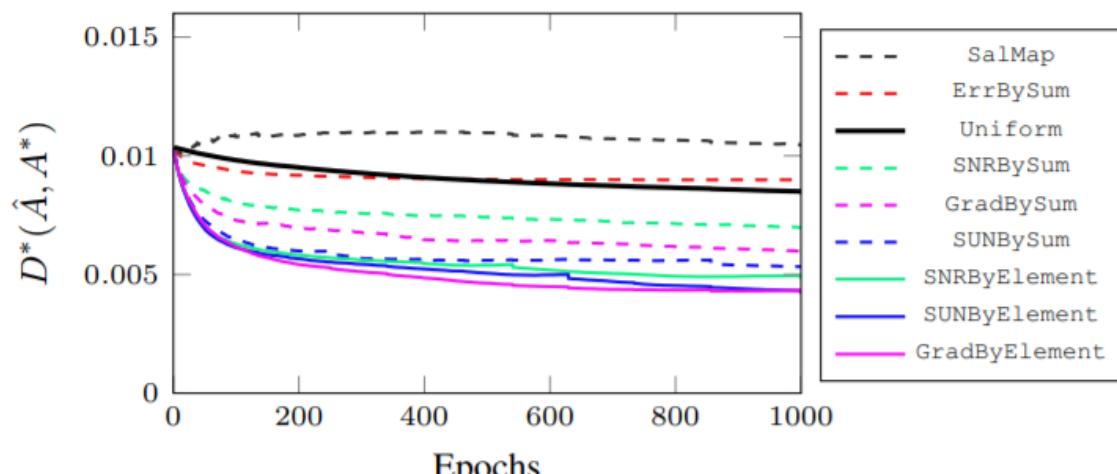
ByElement: $f_{sel} = \{\text{top } k \text{ elements of } g_j^i | j \in 1 \dots n\}$

针对每个element计算分值，选出该element分值最高的几个样本

Distance from true dictionary



(a) Gabor dictionary



(b) Alphanumeric dictionary

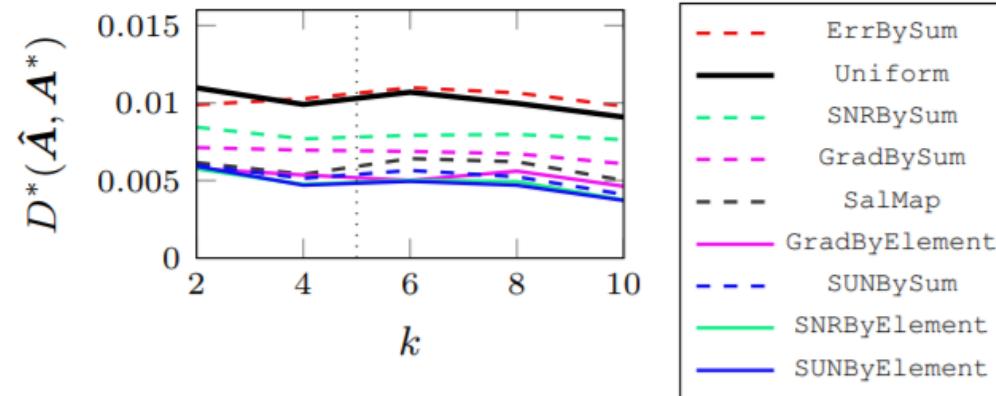
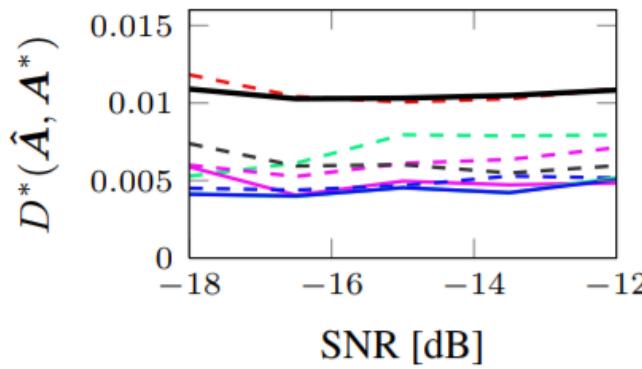
robustness

SNR: signal to noise ratio $(10 \log_{10}(2\lambda^2/\sigma_\epsilon^2))$

k: number of nonzero elements

n/N: ratio of selected examples to the original training set

K: number of dictionary elements



Algorithm is robust

