metric learning

1 basic knowledge

Definition

LOGO

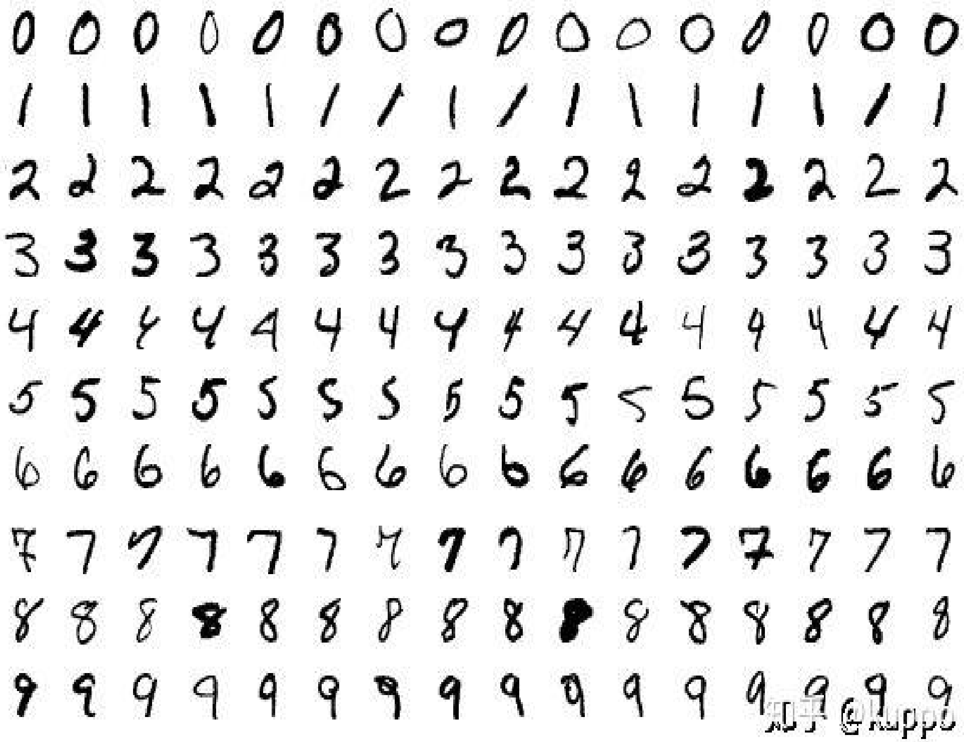
Metric learning aims to measure the similarity among samples while using an optimal distance metric for learning tasks.

在机器学习中，对高维数据进行降维的主要目的是希望找到一个合适的低维空间，在此空间中进行学习能比原始空间性能更好。事实上，每个空间对应了在样本属性上定义的一个距离度量，而寻找合适的空间，实质上就是在寻找一个合适的距离度量。那么，为何不直接尝试“学习”出一个合适的距离度量呢?这就是度量学习的基本动机。

Problem in classification

1、 fixed class number

perfect model：



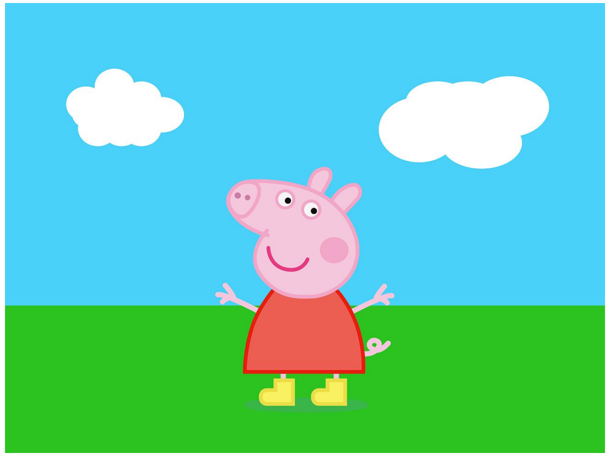
New demand：

Problem in classification

2、 differences between train dataset and test dataset

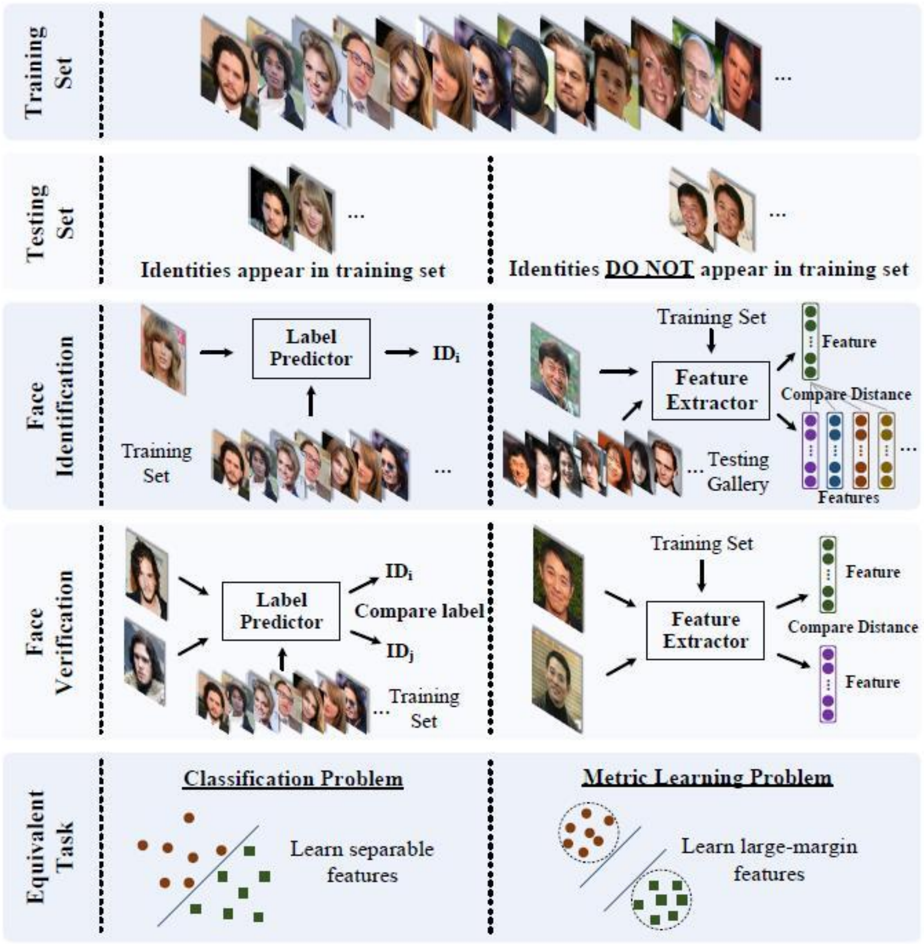
Your training set：

Your testing set：



Open-set V.S Closed-set

Closed-set problem



Open-set problem

Metric Learning in tradintional ML

LOGO

1、method without Learnable parameters：

Linear method：KNN、PCA

Nonlinear method：KPCA

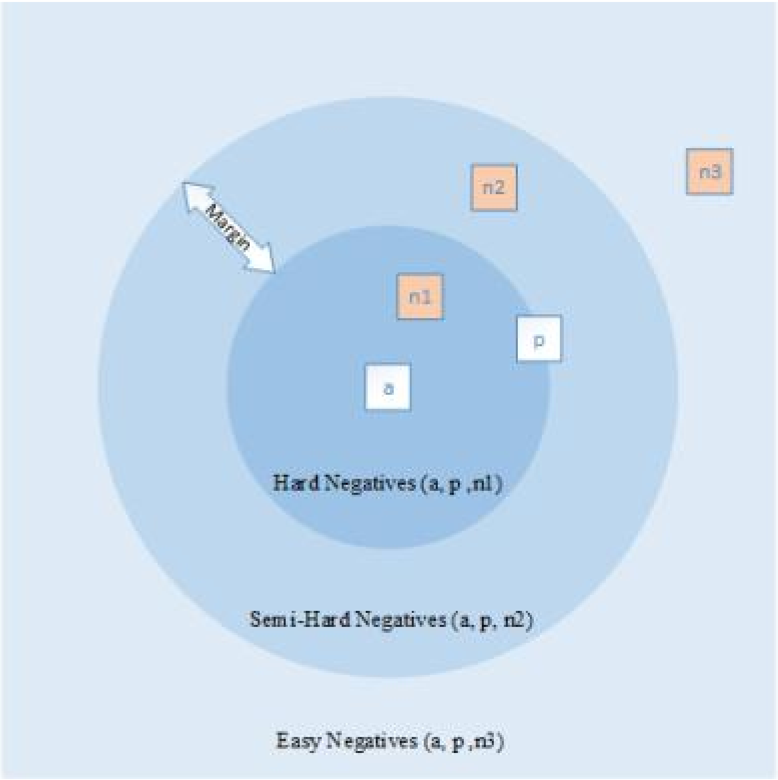
2、method without Learnable parameters：

Mahalanobis distance：

$$D_M(x, y) = \sqrt{(x-y)^T \Sigma^{-1} (x-y)}$$

Metric Learning in Deep Learning

1、 Data mining



Hard Negative Mining

$d(a,n) < d(a,p)$

Semi-Hard Negative Mining

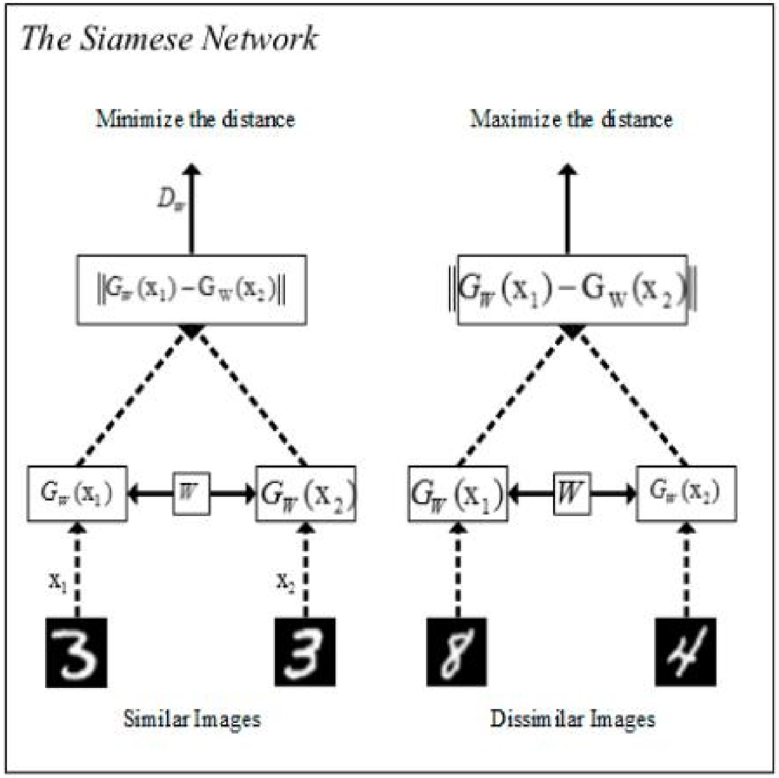
$d(a,p) < d(a,n) < d(a,p) + margin$

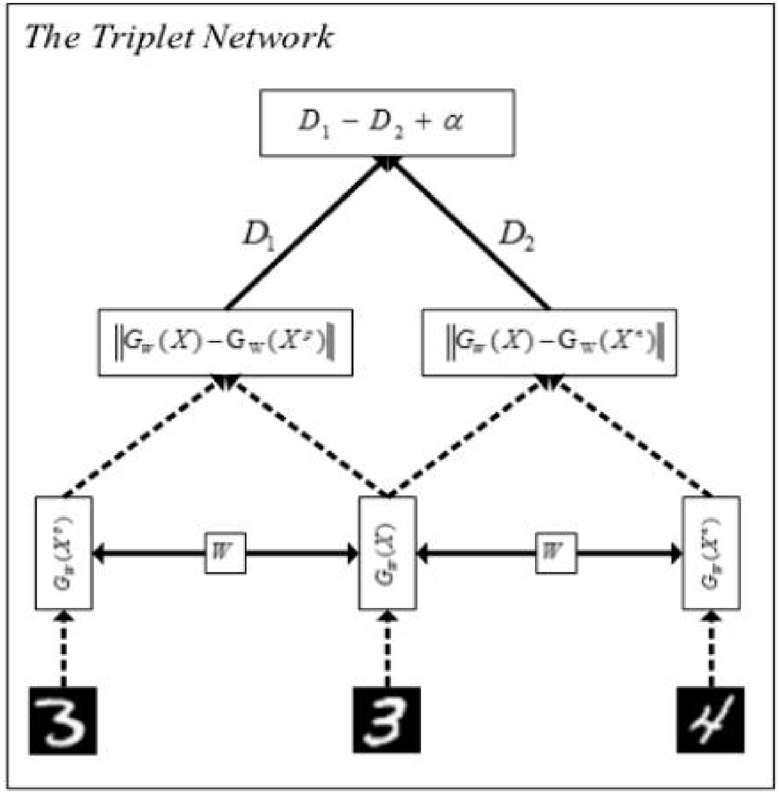
Easy Negative Mining

$d(a,p) + margin < d(a,n)$

Definition

2、Network Structure

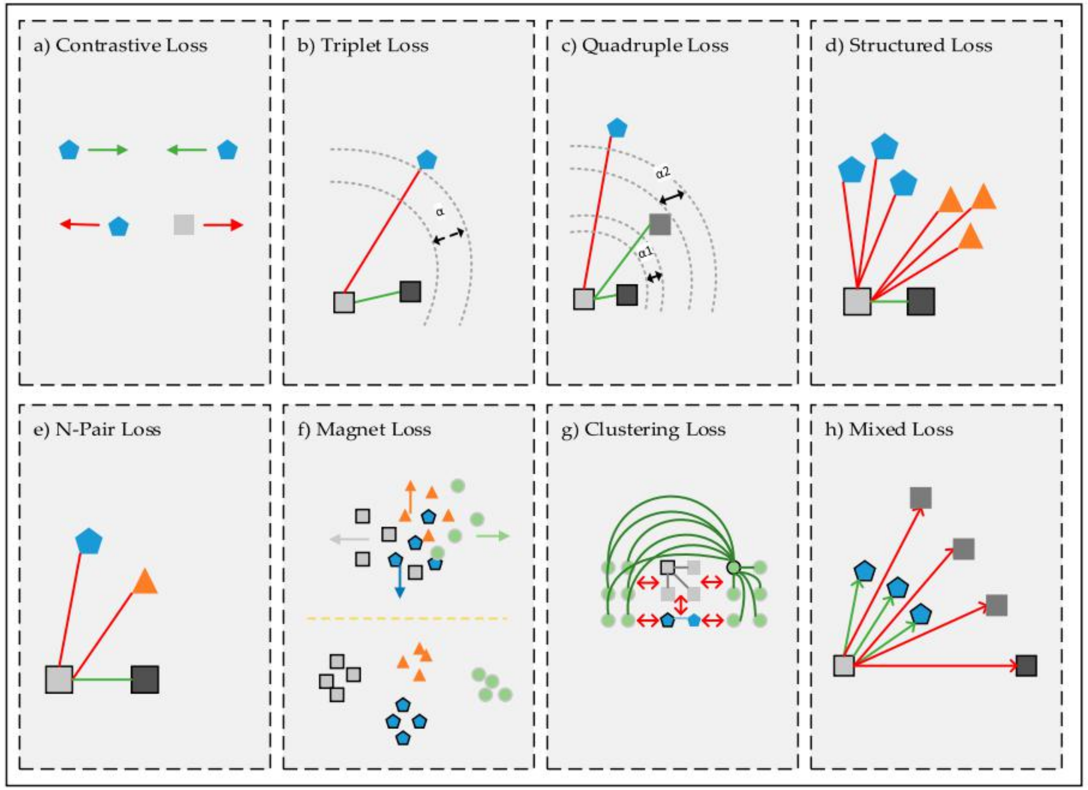




Metric Learning in Deep Learning

LOGO

3、Loss function



$$L_{Contrastive} = (1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{\max(0, m - D_W)\}^2$$

$$L_{Triplet} = \max(0, \|G_W(X) - G_W(X^p)\|_2 - \|G_W(X) - G_W(X^n)\|_2 + \alpha)$$

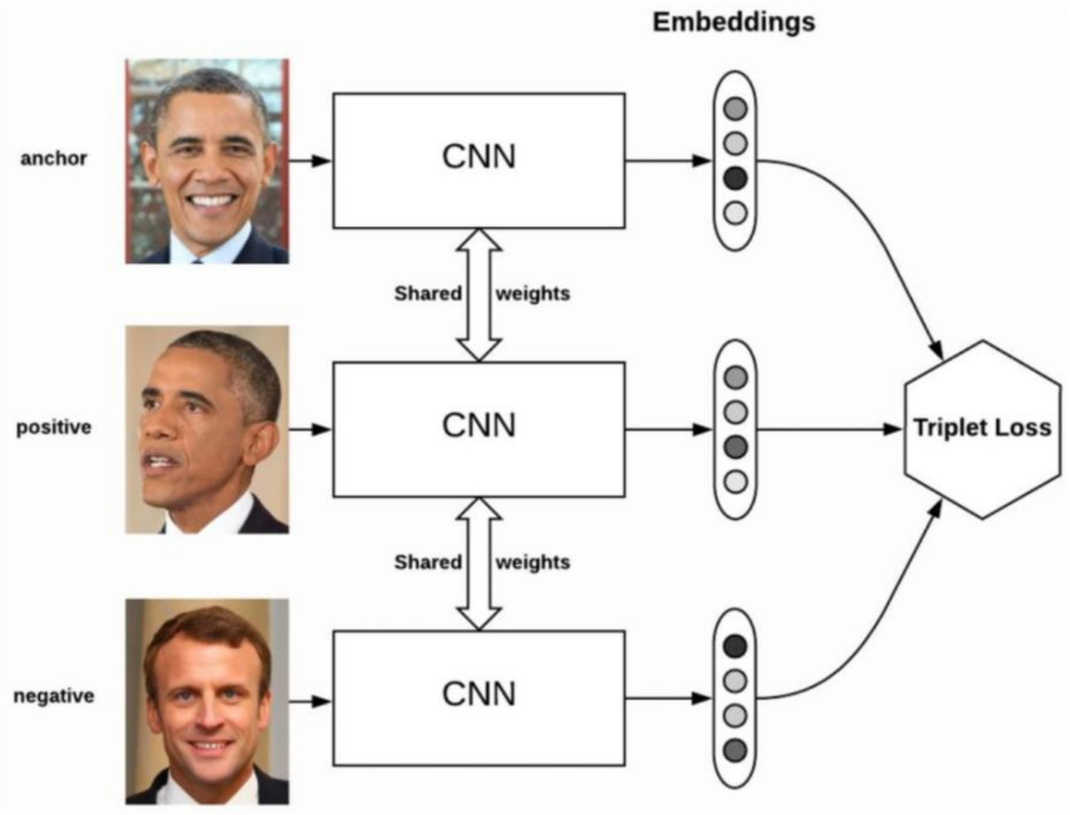
2 metric learning in FR

Metric learning in FR

LOGO

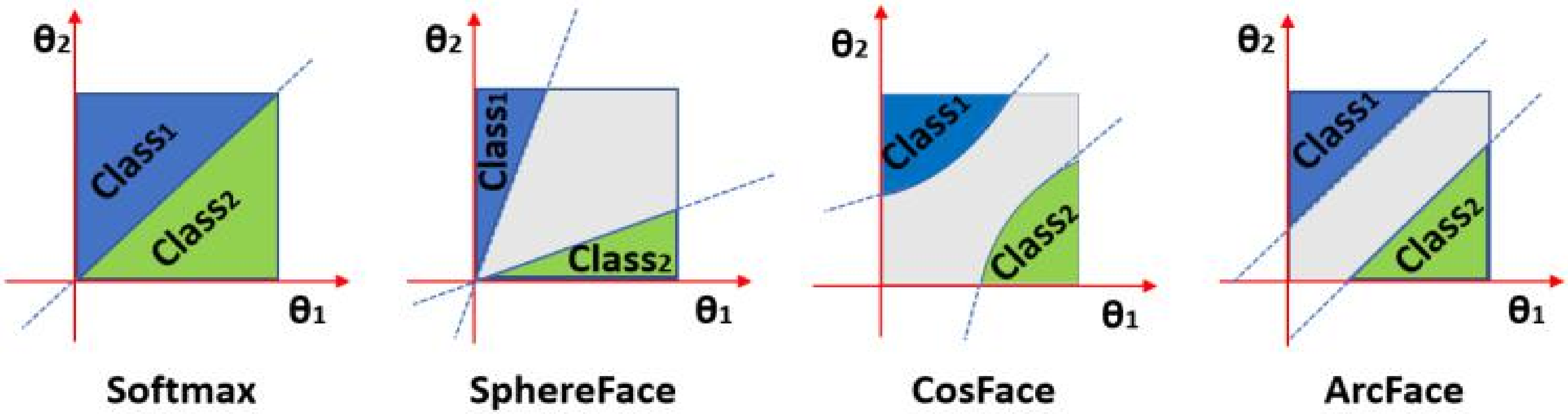
1、Euclidean distance：

$$L_{Triplet} = \max(0, \|G_W(X) - G_W(X^p)\|_2 - \|G_W(X) - G_W(X^n)\|_2 + \alpha)$$



Metric learning in FR

2、Angular Margin：



SphereFace： $\|x\|(\cos m\theta_1 - \cos\theta_2) = 0$

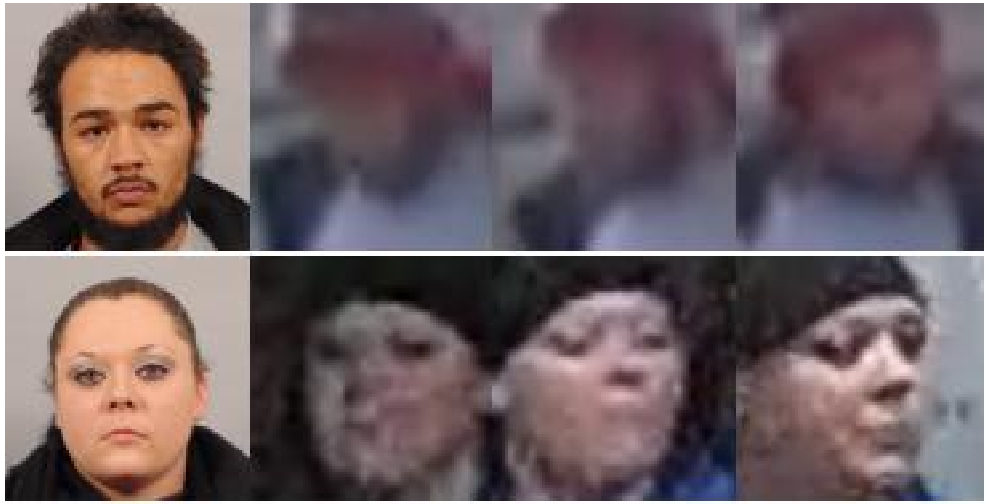
CosFace： $s(\cos\theta_1 - \mathrm{m} - \cos\theta_2) = 0$

ArcFace： $s(\cos(\theta_1 + \mathrm{m}) - \cos\theta_2) = 0$

Probabilistic Face Embeddings

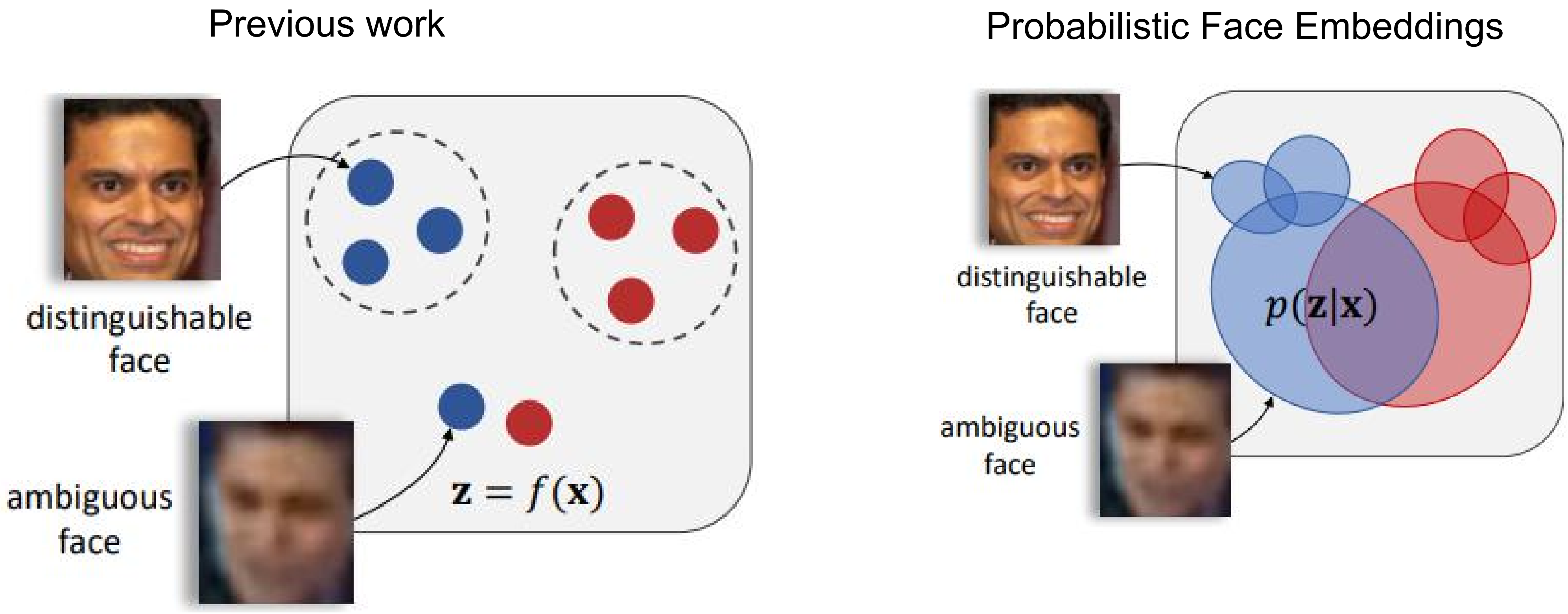
LOGO

IJB-A



IJB-S

Probabilistic Face Embeddings



Probabilistic Face Embeddings

Uncertainty corresponds to image quality/noise

MEGVII 旷视

Large for large pose / blurred / occluded samples

Small for clear / frontal samples

Photos with the same ID sampled from MS-Celeb-1M dataset

Probabilistic Face Embeddings

Given the PFE representations of a pair of images(x_i, x_j), we can directly measure the "likelihood" of them belonging to the same person (sharing the same latent code): $p(z_i = z_j)$, where $z_i \sim p(z|x_i)$ and $z_j \sim p(z|x_j)$.

$$p(\mathbf{z}_i = \mathbf{z}_j) = \int p(\mathbf{z}_i|\mathbf{x}_i)p(\mathbf{z}_j|\mathbf{x}_j)\delta(\mathbf{z}_i - \mathbf{z}_j)d\mathbf{z}_i d\mathbf{z}_j.$$

Probabilistic Face Embeddings

In practice, we would like to use the log likelihood instead, whose solution is given by

$$\begin{aligned} s(\mathbf{x}_i, \mathbf{x}_j) &= \log p(\mathbf{z}_i = \mathbf{z}_j) \\ &= -\frac{1}{2} \sum_{l=1}^{D} \left(\frac{(\mu_i^{(l)} - \mu_j^{(l)})^2}{\sigma_i^{2(l)} + \sigma_j^{2(l)}} + \log(\sigma_i^{2(l)} + \sigma_j^{2(l)}) \right) \\ &\quad - const, \end{aligned} \tag{3}$$

where $const = \frac{D}{2} \log 2\pi$, $\mu_i^{(l)}$ refers to the l^{th} dimension of $\boldsymbol{\mu}_i$ and similarly for $\sigma_i^{(l)}$.

Probabilistic Face Embeddings

LOGO

$$s(\mathbf{x}_i, \mathbf{x}_j) = \log p(\mathbf{z}_i = \mathbf{z}_j)$$
$$= -\frac{1}{2}\sum_{l=1}^{D}\left(\frac{(\mu_i^{(l)} - \mu_j^{(l)})^2}{\sigma_i^{2(l)} + \sigma_j^{2(l)}} + \log(\sigma_i^{2(l)} + \sigma_j^{2(l)})\right)$$
$$- \, const, \qquad (3)$$

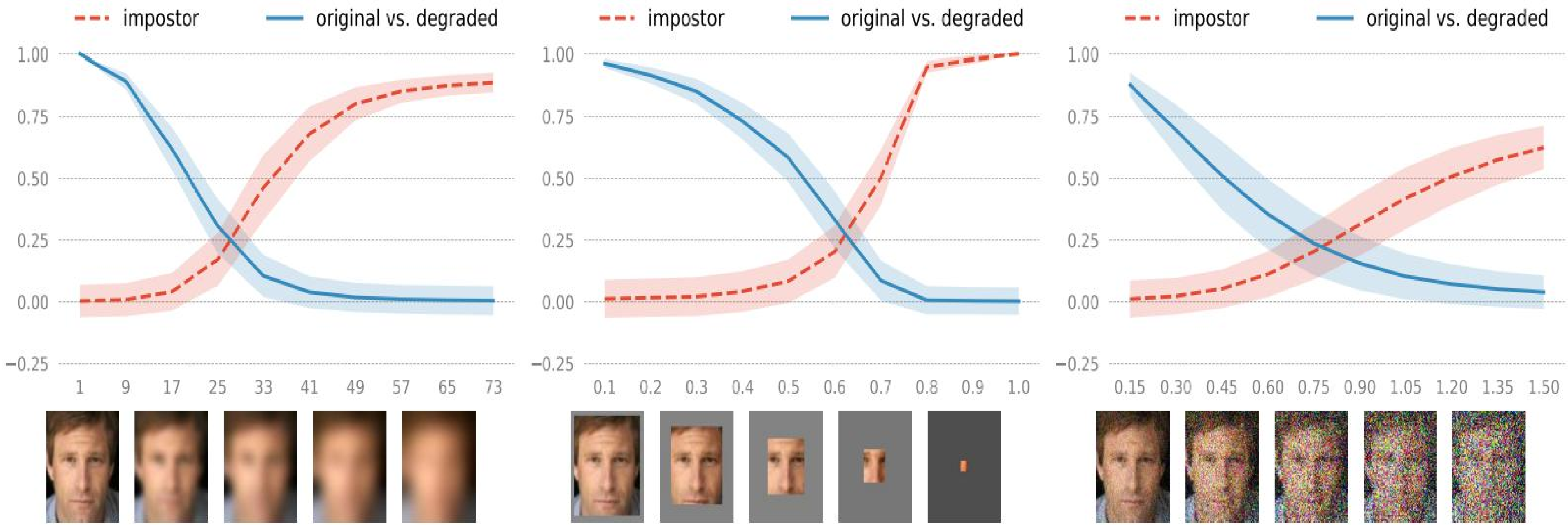
1. Attention mechanism: the first term in the bracket can be seen as a weighted distance which assigns larger weights to less uncertain dimensions.

2. Penalty mechanism: the second term in the bracket in can be seen as a penalty term which penalizes dimensions that have high uncertainties.

3. If either input xi or xj has large uncertainties, MLS will be low (because of penalty) irrespective of the dis_x0002_tance between their mean.

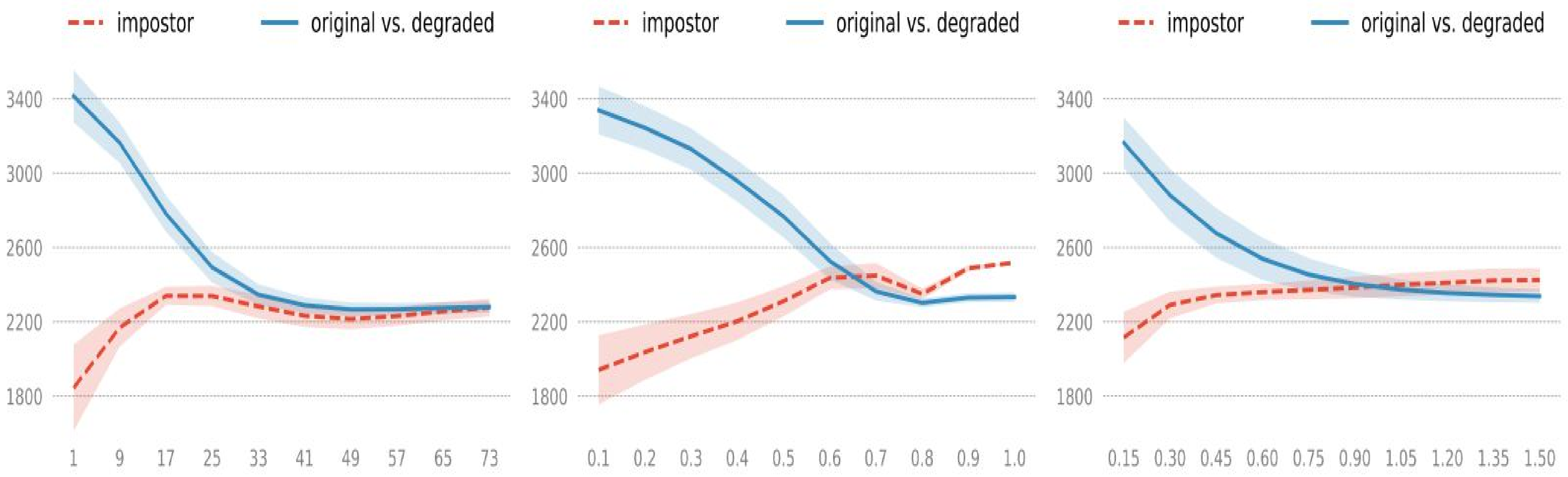
4. Only if both inputs have small uncertainties and their means are close to each other, MLS could be very high.

Probabilistic Face Embeddings



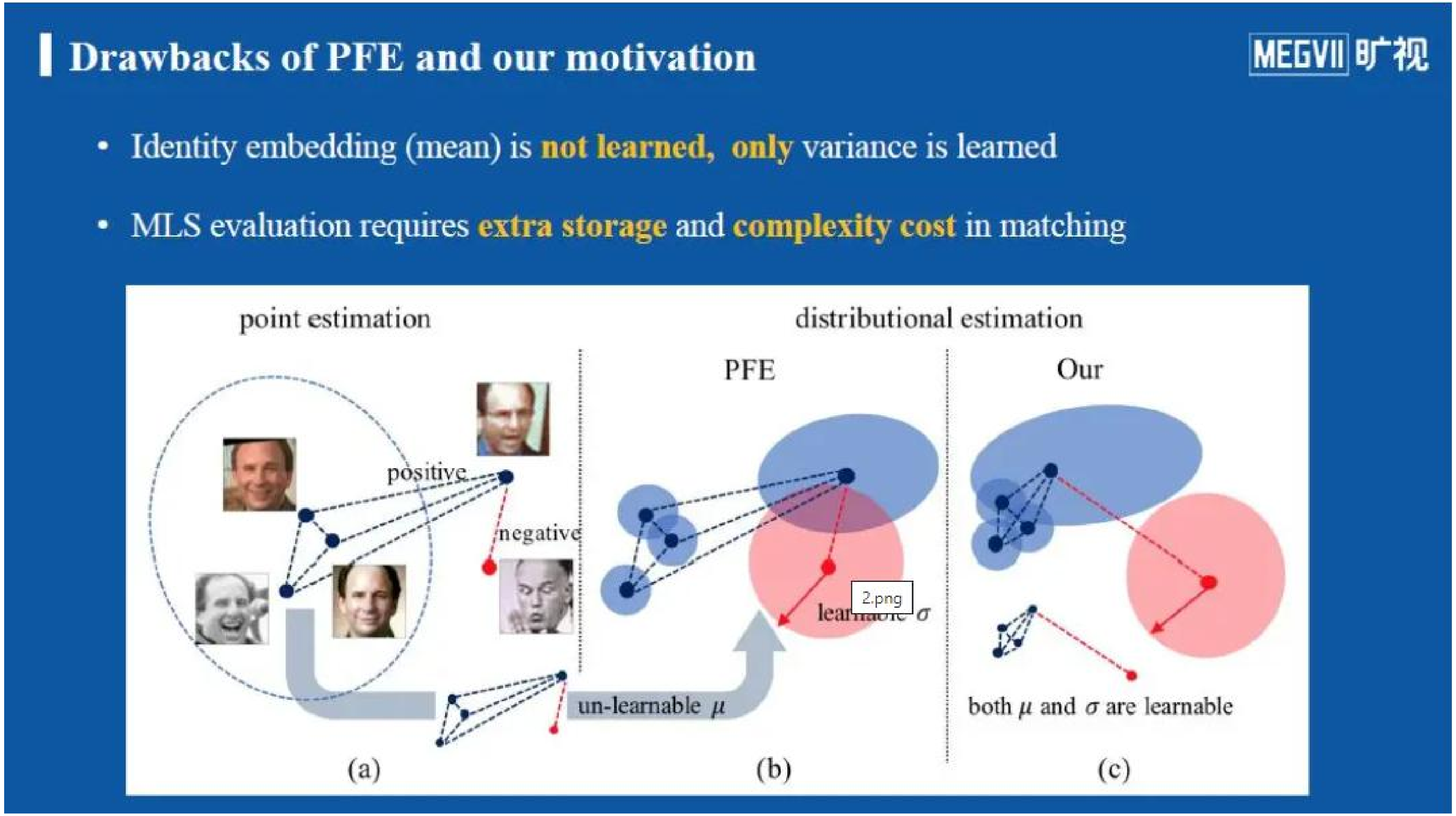
(a) Gaussian Blur (b) Occlusion (c) Random Gaussian Noise

Probabilistic Face Embeddings



(a) Gaussian Blur (b) Occlusion (c) Random Noise

Probabilistic Face Embeddings



Thanks for attention!