端到端声纹识别

End-to-end speaker recognition

张晓雷

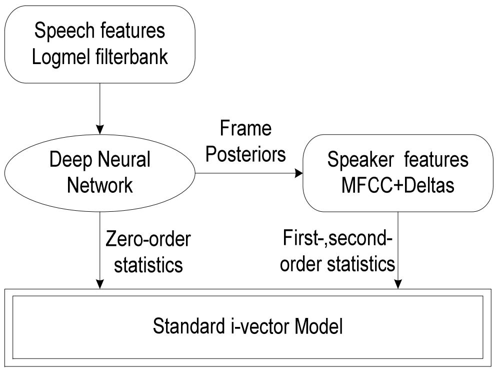
西北工业大学
智能声学与临境通信研究中心

目 录

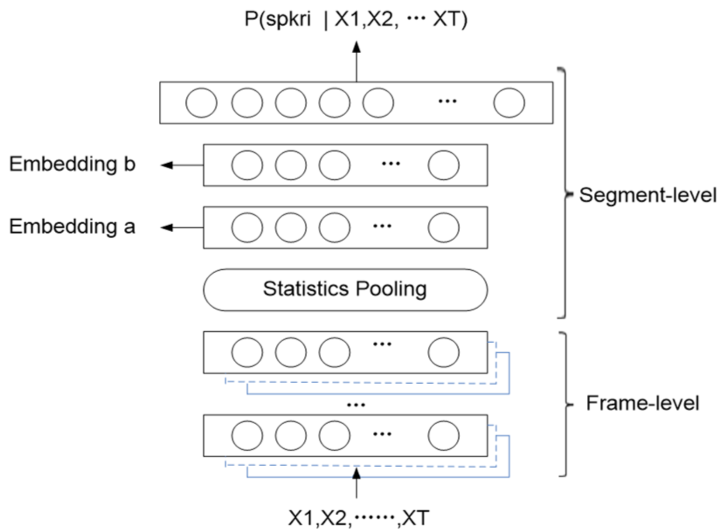
一、研究背景及问题

基于深度学习的声纹识别的三个分支

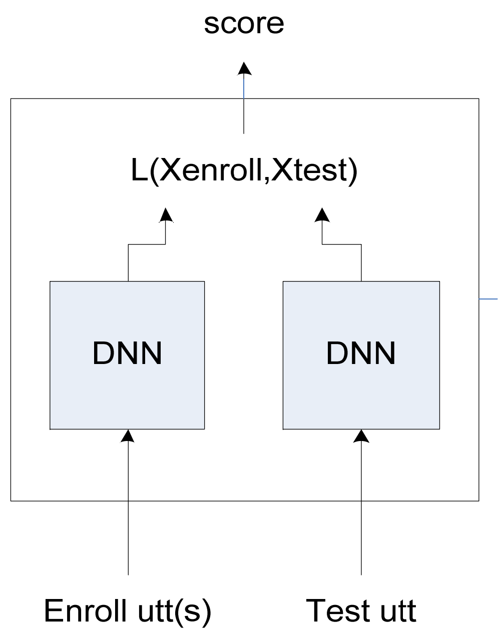
DNN/i-vector



Embedding

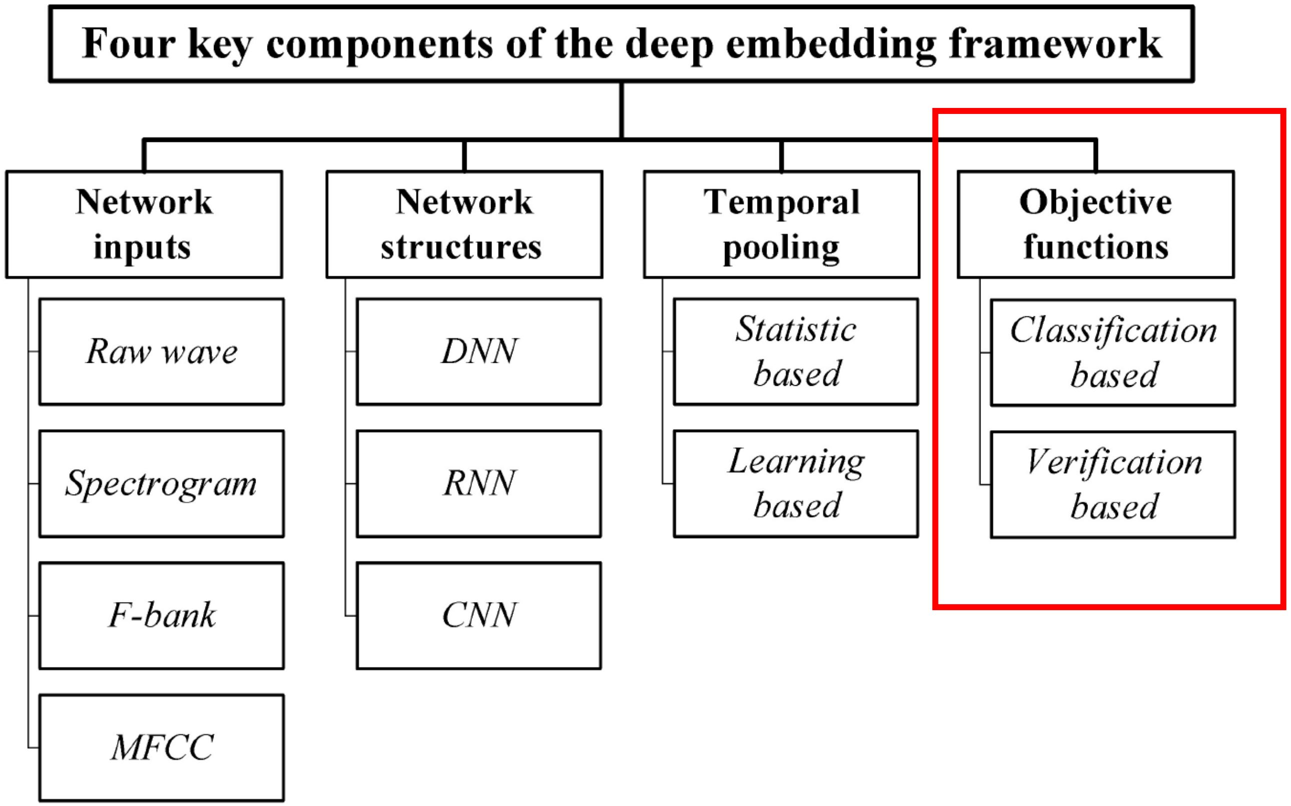


End-to-End



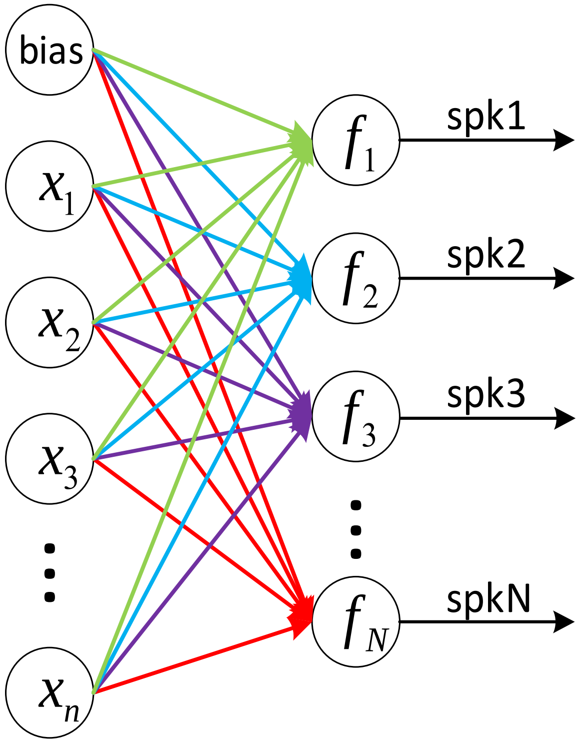
一、研究背景及问题

声纹识别的研究重点



二、非端到端分类损失

分类损失1：Softmax with cross-entropy loss



$$L = \frac{1}{N}\sum_i L_i = \frac{1}{N}\sum_i -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$$

$$f_{y_i} = \boldsymbol{W}_{y_i}^T \boldsymbol{x}_i + b_{y_i} \qquad f_j = \boldsymbol{W}_j^T \boldsymbol{x}_i + b_j$$

$$\boldsymbol{W}_j^T \boldsymbol{x}_i = \|\boldsymbol{W}_j\|\,\|\boldsymbol{x}_i\|\cos(\theta_{j,i})$$

$$\begin{aligned} L_i &= -\log\left(\frac{e^{\boldsymbol{W}_{y_i}^T \boldsymbol{x}_i + b_{y_i}}}{\sum_j e^{\boldsymbol{W}_j^T \boldsymbol{x}_i + b_j}}\right) \\ &= -\log\left(\frac{e^{\|\boldsymbol{W}_{y_i}\|\|\boldsymbol{x}_i\|\cos(\theta_{y_i,i})+b_{y_i}}}{\sum_j e^{\|\boldsymbol{W}_j\|\|\boldsymbol{x}_i\|\cos(\theta_{j,i})+b_j}}\right)\end{aligned}$$

$$\|\boldsymbol{W}_j\| = 1$$

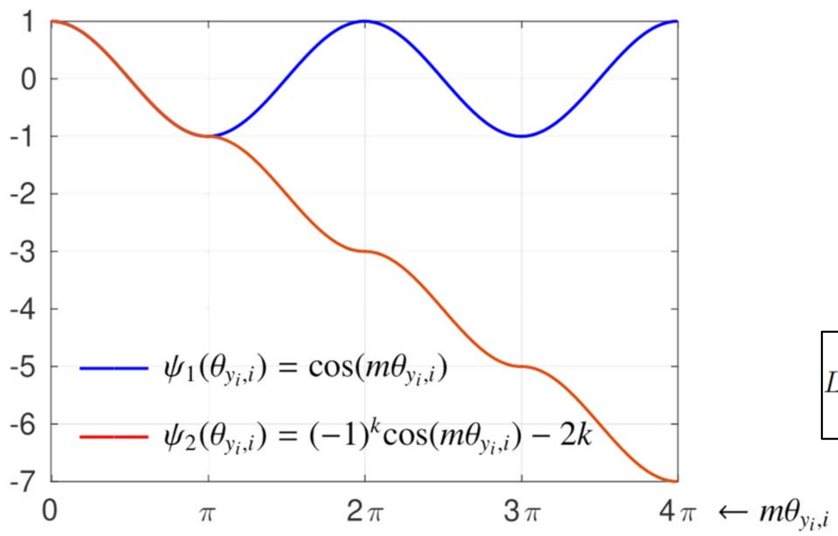
$$L_{\text{modified}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\cos(\theta_{y_i,i})}}{\sum_j e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)$$

1）将开集问题当闭集问题处理
2）只最大化类间距离，没有最小化类内方差

参考文献

二、非端到端分类损失

分类损失2： Angular softmax



$m\theta_{y_i,i}$ ***with*** $m = 4$ ***and*** $\theta_{y_i,i} \in [0, \pi]$

$$\boxed{L_{\text{modified}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\cos(\theta_{y_i,i})}}{\sum_j e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)}$$

$$\boxed{L_{\text{ang}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\cos(m\theta_{y_i,i})}}{e^{\|\boldsymbol{x}_i\|\cos(m\theta_{y_i,i})} + \sum_{j\neq y_i} e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)}$$

$$\psi(\theta_{y_i,i}) = (-1)^k\cos(m\theta_{y_i,i}) - 2k$$

$$\theta_{y_i,i} \in \left[\frac{k\pi}{m}, \frac{(k+1)\pi}{m}\right] \text{ and } k \in [0, m-1]$$

$$\boxed{L_{\text{ang}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})}}{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})} + \sum_{j\neq y_i} e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)}$$

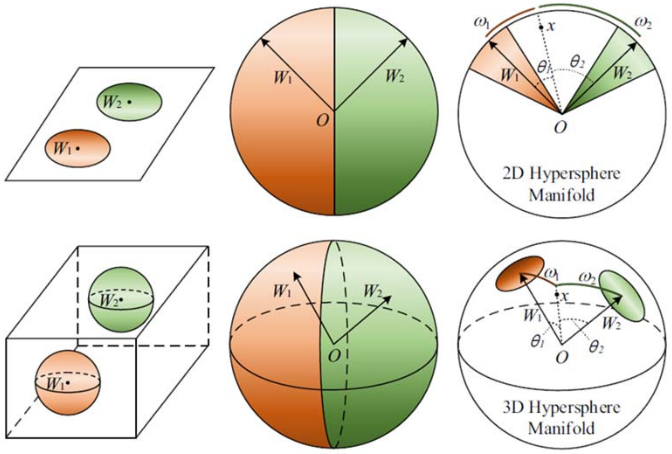
优点：
1）最小化类内方差（通过增加类间的角度margin）
2）与cosine similarity scoring匹配

参考文献

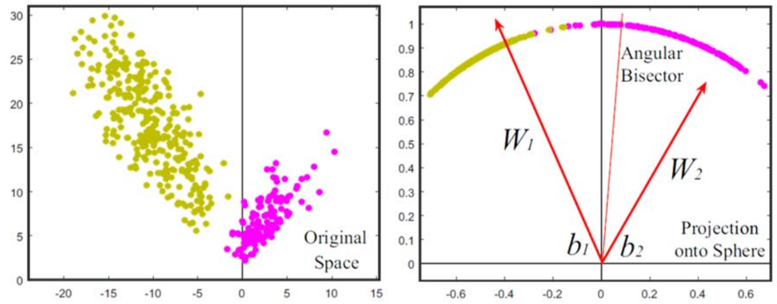
Huang et al., Angular softmax for short-duration text-independent speaker verification. in Interspeech, 2018
Cai et al., Exploring the encoding layer and loss function. in end-to-end speaker and language recognition system, in: Proc. Odyssey, 2018

二、非端到端分类损失

分类损失2： Angular softmax

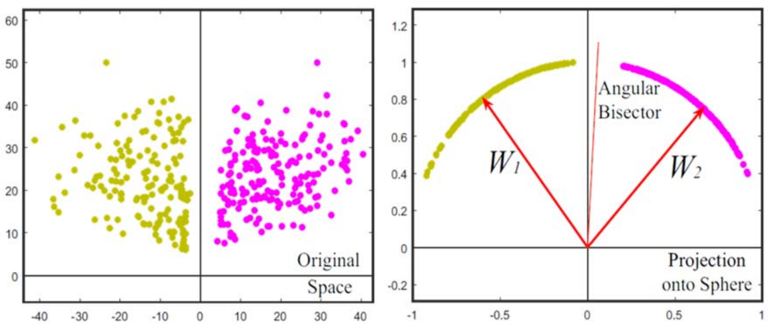


Euclidean Margin Loss　　Modified Softmax Loss　　A-Softmax Loss (m≥2)

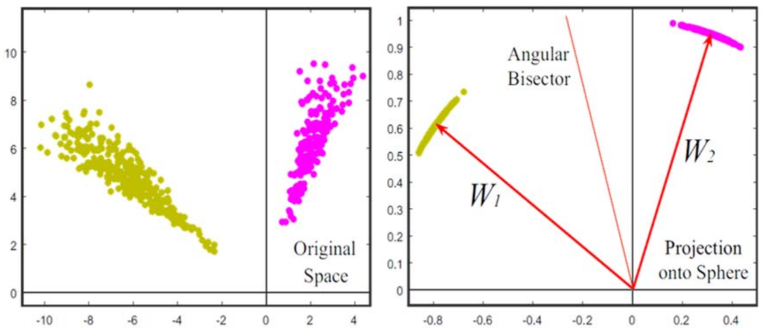


(a) Original Softmax Loss　　(b) Original Softmax Loss

$$L = \frac{1}{N}\sum_i L_i = \frac{1}{N}\sum_i -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right) \qquad f_j = \boldsymbol{W}_j^T \boldsymbol{x}_i + b_j$$



(c) Modified Softmax Loss　　(d) Modified Softmax Loss



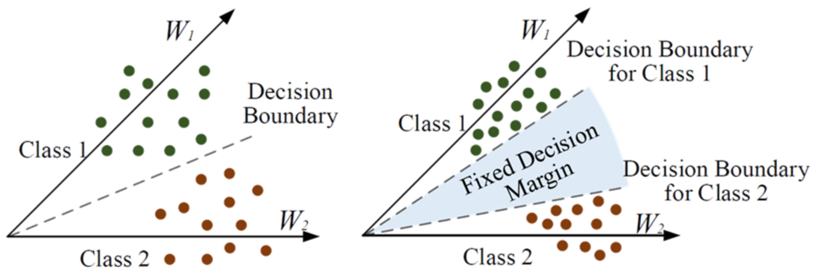
(e) A-Softmax Loss　　(f) A-Softmax Loss

$$L_{\text{modified}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\cos(\theta_{y_i,i})}}{\sum_j e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right) \qquad \|\boldsymbol{W}_j\|=1$$

$$L_{\text{ang}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})}}{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})} + \sum_{j\neq y_i} e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)$$

二、非端到端分类损失

分类损失3： Additive margin softmax (AMS)，Additive angular margin softmax(AAMS)

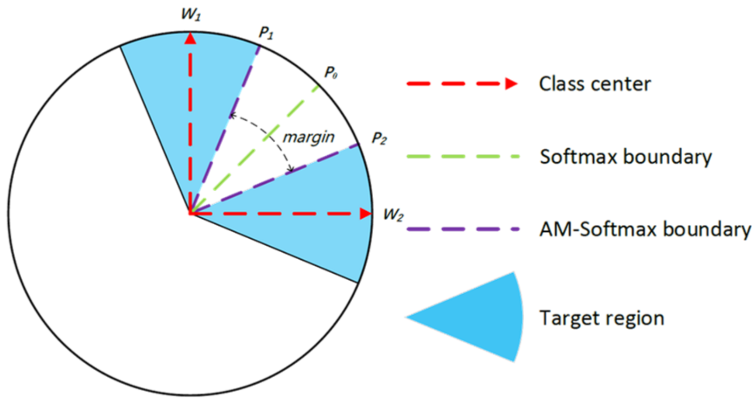


Original Softmax **Additive Margin Softmax**

$$L_{\text{ang}} = \frac{1}{N}\sum_i -\log\left(\frac{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})}}{e^{\|\boldsymbol{x}_i\|\psi(\theta_{y_i,i})} + \sum_{j\neq y_i} e^{\|\boldsymbol{x}_i\|\cos(\theta_{j,i})}}\right)$$

$$\|\boldsymbol{x}_i\| = 1$$

$$\psi(\theta_{y_i,i}) \rightarrow \begin{cases} \cos(\theta_{y_i,i}) - m \\ \cos(\theta_{y_i,i} + m) \end{cases}$$



$$\mathcal{L}_{\text{AMS}} = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i,i})-m)}}{e^{s(\cos(\theta_{y_i,i})-m)} + \sum_{j\neq y_i} e^{s(\cos(\theta_{j,i}))}}$$

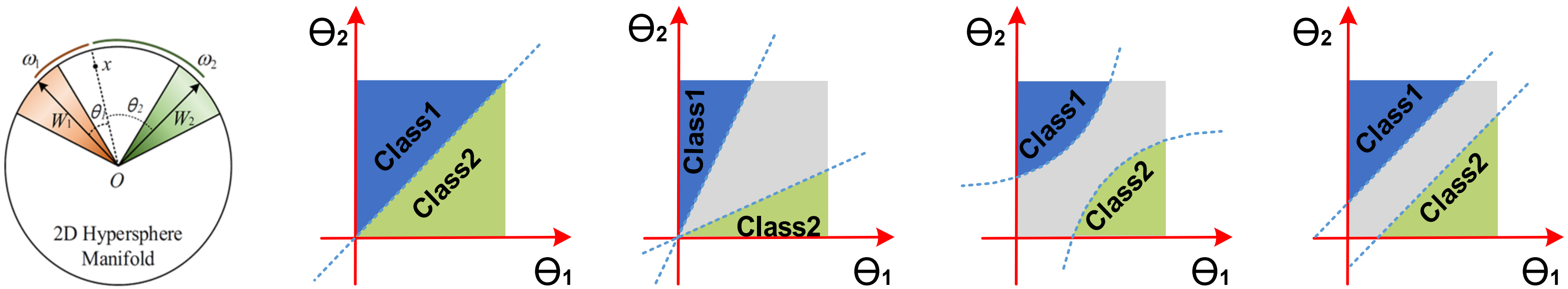
$$\mathcal{L}_{\text{AAMS}} = -\frac{1}{N}\sum_{n=1}^{N}\log\frac{e^{s(\cos(\theta_{y_i,i}+m))}}{e^{s(\cos(\theta_{y_i,i}+m))} + \sum_{j\neq y_i} e^{s(\cos(\theta_{j,i}))}}$$

参考文献

Xie et al., Utterance-level aggregation for speaker recognition in the wild, ICASSP 2019.
Liu et al., Large margin softmax loss for speaker verification, Interspeech 2019.

二、非端到端分类损失

分类损失总结：物理含义



Modified Softmax | ASoftmax | AMSoftmax | AAMSoftmax

$$\|\boldsymbol{x}\|(\cos\theta_1 - \cos\theta_2) = 0$$

$$\|\boldsymbol{x}\|(\cos m\theta_1 - \cos\theta_2) = 0$$
$$\|\boldsymbol{x}\|(\cos\theta_1 - \cos m\theta_2) = 0$$

$$\cos(\theta_1) - \cos(\theta_2) = m$$
$$\cos(\theta_2) - \cos(\theta_1) = m$$

$$\cos(\theta_2 + m) = \cos(\theta_1)$$
$$\cos(\theta_1 + m) = \cos(\theta_2)$$

二、非端到端分类损失

分类损失的正则项方法

正则项框架：

$$\mathcal{L} = \mathcal{L}_{\mathrm{S}} + \lambda \mathcal{L}_{\mathrm{Regular}}$$

类中心正则项（Class-center loss）：

$$\mathcal{L}_{\mathrm{C}} = \frac{1}{2} \sum_{n=1}^{N} \|\mathbf{x}_n - \mathbf{c}_{l_n}\|^2$$

环损失正则项（Ring loss）：

$$\mathcal{L} = \mathcal{L}_{\mathrm{AMS}} + \lambda \times \frac{1}{N} \sum_{n=1}^{N} (\|\mathbf{x}_n\|_2 - R)^2$$

Gaussian prior：

$$\mathcal{L} = \mathcal{L}_{\mathrm{S}} + \lambda \sum_{j} \sum_{\mathbf{e}_n \in \varepsilon(j)} \|\mathbf{e}_n - \mathbf{w}_j\|$$

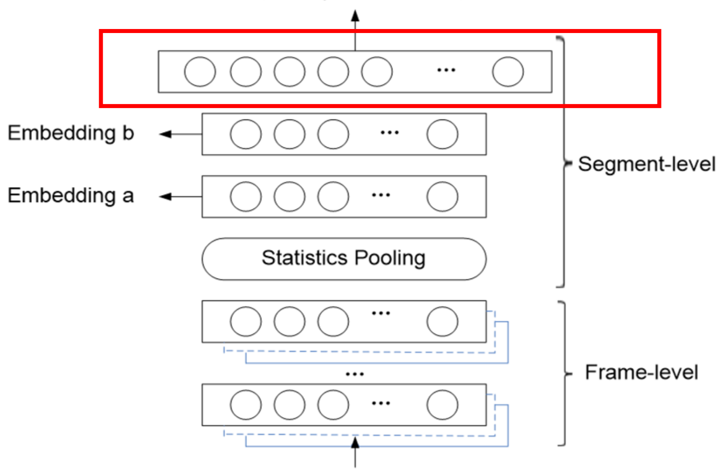
参考文献

Cai et al., Exploring the encoding layer and loss function in end-to-end speaker and language recognition system, Odyssy 2019,
Liu, et al., Large margin softmax loss for speaker verification,Proc. Interspeech 2019
Li et al., Gaussian-constrained training for speaker verification, in: ICASSP 2019

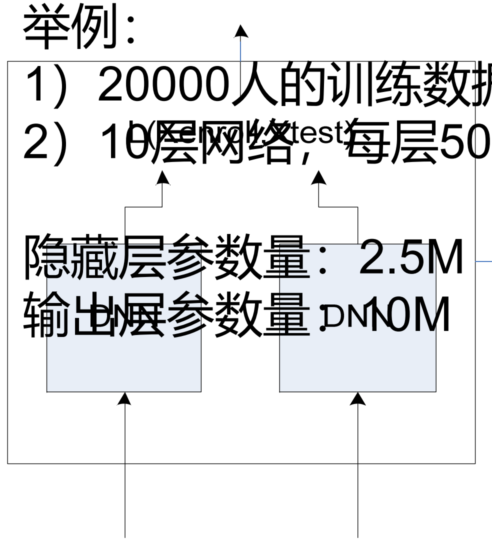
二、非端到端分类损失

优点	缺点
• 有效 • 模型训练稳定	• 标签需要精确到每句话对应的说话人身份 • 优化替代损失—softmax，可能并非最优 • 输出层随说话人数量增加而变大

训练阶段



测试阶段



举例：

1）20000人的训练数据集

2）10层网络，每层500个神经元

隐藏层参数量：2.5M

输出层参数量：10M

三、端到端确认损失

End-to-end SV

- Design a training loss
  - ✓ Pairwise loss
  - ✓ Triplet loss
  - ✓ Quadruplet loss
  - ✓ Prototypical network loss
  - ✓ . . .
- Define a similarity metric
  - ✓ PLDA-based similarity metric
  - ✓ Cosine similarity metirc
  - ✓ Squared Euclidean distance
  - ✓ . . .
- Select and construct training trials
  - ✓ hard negative
  - ✓ semi-hard
  - ✓ Partial AUC
  - ✓ . . .

三、端到端确认损失

确认损失1： Pairwise loss

Binary cross-entropy loss

$$\mathcal{L}_{\mathrm{BCE}} = -\sum_{n=1}^{N}\left[l_n \ln\left(p(\mathbf{x}_n^e, \mathbf{x}_n^t)\right) - \eta(1-l_n)\ln\left(1 - p(\mathbf{x}_n^e, \mathbf{x}_n^t)\right)\right]$$

- $$p(\mathbf{x}_n^e, \mathbf{x}_n^t) = \frac{1}{1+\exp(-S(\mathbf{x}_n^e, \mathbf{x}_n^t))}$$ $$S(\mathbf{x}_n^e, \mathbf{x}_n^t) = (\mathbf{x}_n^e)^T\mathbf{x}_n^t - (\mathbf{x}_n^e)^T\mathbf{S}\mathbf{x}_n^e - (\mathbf{x}_n^t)^T\mathbf{S}\mathbf{x}_n^t + b$$ PLDA

- $$p(\mathbf{x}_n^e, \mathbf{x}_n^t) = \frac{1}{1+\exp(-wS(\mathbf{x}_n^e, \mathbf{x}_n^t) - b)}$$ $$S(\mathbf{x}_n^e, \mathbf{x}_n^t) = \frac{\mathbf{x}_n^{e^T}\mathbf{x}_n^t}{\|\mathbf{x}_n^e\|\,\|\mathbf{x}_n^t\|}$$ Cosine

- $$p(\mathbf{x}_n^e, \mathbf{x}_n^t) = \frac{1}{1+\exp(-s_n^{e,t})}$$ $$s_n^{e,t} = S(\mathbf{x}_n^{e,t})$$ Attention based

Contrastive loss

$$\mathcal{L}_{\mathrm{C}} = \frac{1}{2N}\sum_{n=1}^{N}\left(l_n \cdot d_n^2 + (1-l_n)\max(\rho - d_n, 0)^2\right)$$

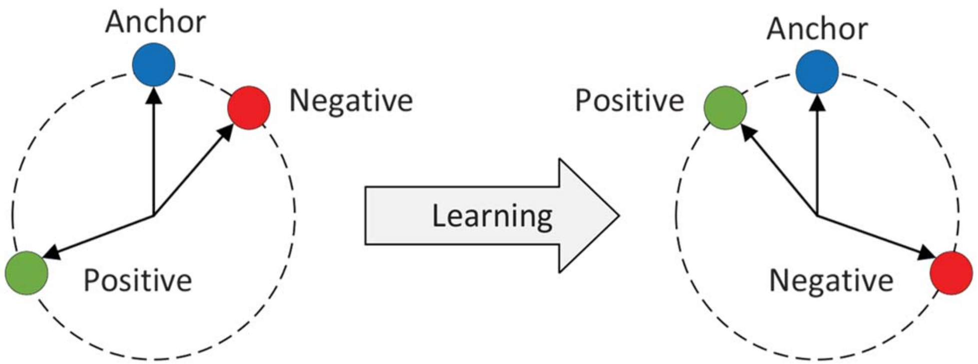
margin

参考文献

Heigold et al., End-to-end text-dependent speaker verification, in: ICASSP 2016

三、端到端确认损失

确认损失2： Triplet loss



$$\mathcal{X}_{\text{trip}} = \{(\mathbf{x}_n^a, \mathbf{x}_n^p, \mathbf{x}_n^n) | n = 1, 2, \cdots, N\}$$

$$s_n^{an} - s_n^{ap} + \zeta \leq 0$$

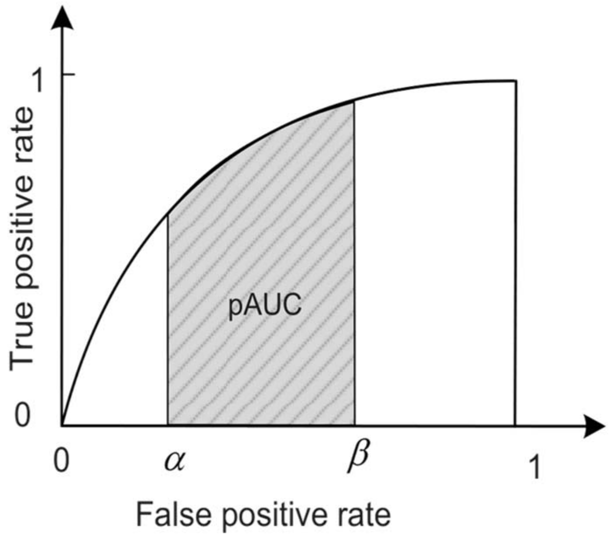
$$\mathcal{L}_{\text{trip}} = \sum_{n=1}^{N} \max(0, s_n^{an} - s_n^{ap} + \zeta)$$

参考文献

- Li et al., Deep speaker: an end-to-end neural speaker embedding system, arXiv preprint arXiv:1705.02304.
- Zhang et al., Text-independent speaker verification based on triplet convolutional neural network embeddings, IEEE/ACM TASLP 2018

三、端到端确认损失

确认损失3： Quadruplet loss



$$\text{pAUC} = 1 - \frac{1}{IK} \sum_{\forall i: s_i \in \mathcal{P}} \sum_{\forall k: s_k \in \mathcal{N}_0} \left[\mathbb{I}(s_i < s_k) + \frac{1}{2}\mathbb{I}(s_i = s_k) \right]$$

$$\mathcal{P} = \{(s_i, l_i = 1) | i = 1, 2, \cdots, I\}$$

$$\mathcal{N}_0 = \{(s_k, l_k = 0) | k = 1, 2, \cdots, K\}$$

$$s_n = f(\mathbf{x}_n, \mathbf{y}_n) = \frac{\mathbf{x}_n^T \mathbf{y}_n}{||\mathbf{x}_n|| \, ||\mathbf{y}_n||}$$

$$\ell'_{\text{hinge}}(z) = \max(0, \delta - z)^2$$

$$\min \frac{1}{IK} \sum_{\forall i: s_i \in \mathcal{P}} \sum_{\forall k: s_k \in \mathcal{N}_0} \max\left(0, \delta - (s_i - s_k)\right)^2$$

参考文献

Bai et al., Partial AUC optimization based deep speaker embeddings with class-center learning for text-independent speaker verification, in: ICASSP 2020

三、端到端确认损失

确认损失3： Quadruplet loss 的类中心学习算法

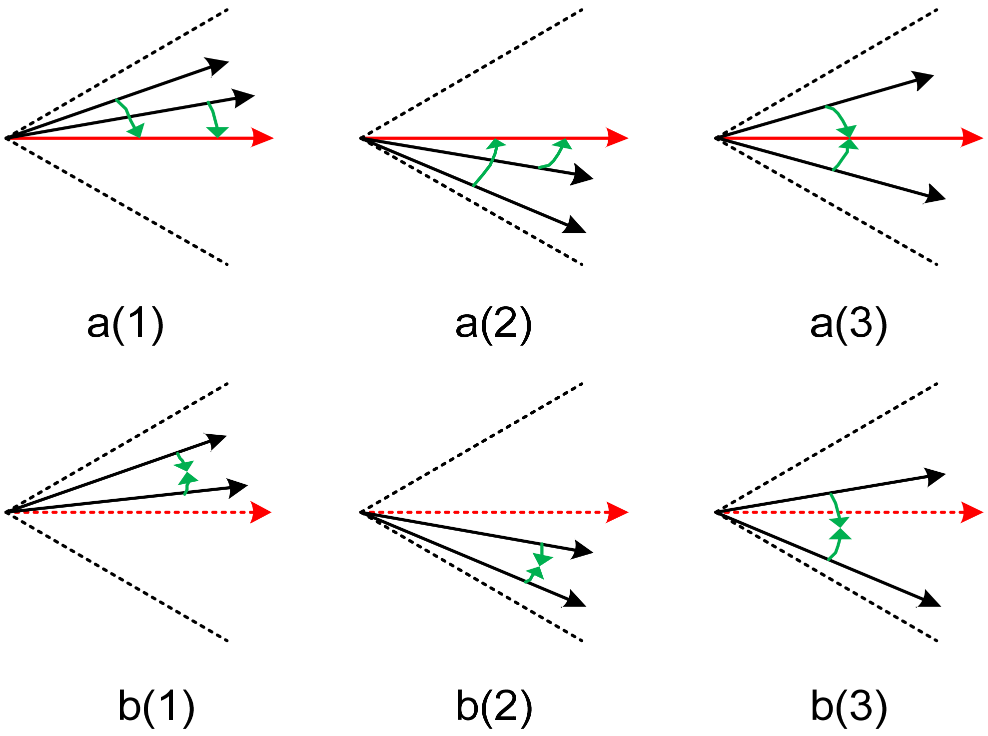
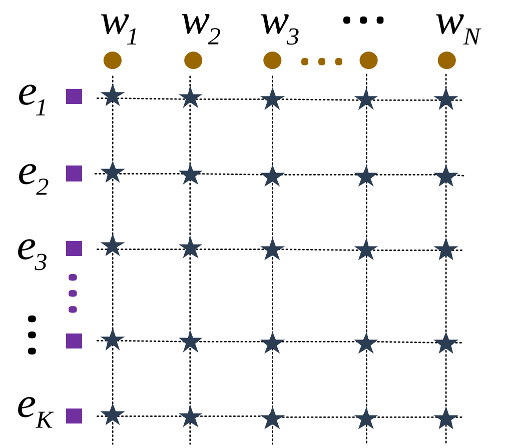


Fig2: a,Class-center learning; b，Random sampling.



$Batchsize = K, Speakers = N; (e_k, w_n)_{\forall k=1:K; \forall n=1:N}$

参考文献

Bai et al., Partial AUC optimization based deep speaker embeddings with class-center learning for text-independent speaker verification, in: ICASSP 2020

三、端到端确认损失

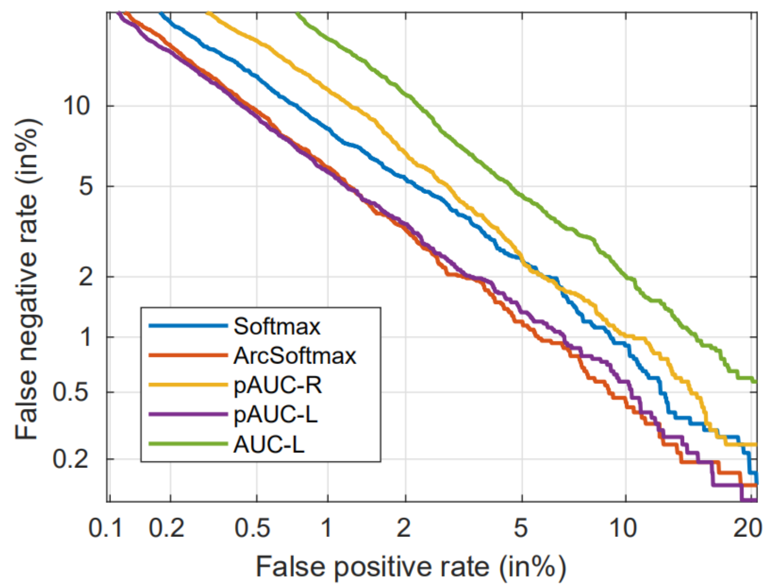
确认损失3： Quadruplet loss 的实验性能

Table 1. Results on SITW.

Name	Loss	EER(%)	$DCF10^{-2}$	$DCF10^{-3}$
Dev.Core	Softmax (kaldi)	3.0	-	-
	Softmax	3.04	0.2764	**0.4349**
	ArcSoftmax	**2.16**	**0.2565**	0.4501
	pAUC-R	3.20	0.3412	0.5399
	pAUC-L	**2.23**	**0.2523**	**0.4320**
	AUC-L	4.27	0.4474	0.6653
Eval.Core	Softmax (kaldi)	3.5	-	-
	Softmax	3.45	0.3339	**0.4898**
	ArcSoftmax	**2.54**	**0.3025**	0.5142
	pAUC-R	3.74	0.3880	0.5797
	pAUC-L	**2.56**	**0.2949**	0.5011
	AUC-L	4.76	0.5005	0.7155

Table 2. Results on the Cantonese language of NIST SRE 2016.

Back-end	Loss	EER(%)	$DCF10^{-2}$	$DCF10^{-3}$
No-adaptation	Softmax (kaldi)	7.52	-	-
	Softmax	6.76	0.5195	0.7096
	ArcSoftmax	**5.59**	**0.4640**	**0.6660**
	pAUC-R	15.25	0.8397	0.9542
	pAUC-L	**6.01**	**0.5026**	**0.7020**
	AUC-L	7.92	0.5990	0.8072
Adaptation	Softmax (kaldi)	4.89	-	-
	Softmax	4.94	0.4029	0.5949
	ArcSoftmax	**4.13**	**0.3564**	**0.5401**
	pAUC-R	8.65	0.6653	0.8715
	pAUC-L	**4.25**	**0.3704**	**0.5471**
	AUC-L	5.36	0.4439	0.6480

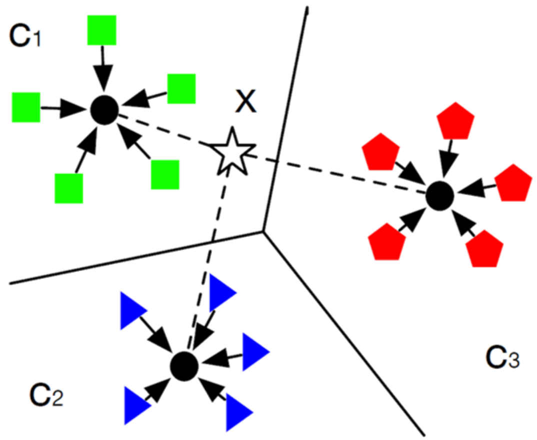


参考文献

Bai et al., Partial AUC optimization based deep speaker embeddings with class-center learning for text-independent speaker verification, in: ICASSP 2020

三、端到端确认损失

确认损失4： Prototypical network loss



一个mini-batch

$$\mathcal{S} = \{(\mathbf{x}_n, l_n) | n = 1, 2, \cdots, N\}$$

一个Query set

$$\mathcal{Q} = \{(\mathbf{x}_q, l_q) | q = 1, 2, \cdots, Q\}$$

在所有样本上计算类中心

$$\mathbf{c}_j = \frac{1}{|\mathcal{S}_j|} \sum_{(\mathbf{x}_n, l_n) \in \mathcal{S}_j} \mathbf{x}_n, \quad j = 1, 2, \cdots, J$$

$$\mathcal{L}_{\text{PNL}} = -\sum_{(\mathbf{x}_q, l_q) \in \mathcal{Q}} \log \frac{\exp\big(-d(\mathbf{x}_q, \mathbf{c}_{l_q})\big)}{\sum_{j'=1}^{J} \exp\big(-d(\mathbf{x}_q, \mathbf{c}_{j'})\big)}$$

参考文献 Chung et al., In defence of metric learning for speaker recognition, in: Interspeech 2020.

三、端到端确认损失

确认损失4： Prototypical network loss

Table1: Equal Error Rates (EER, %) on the VoxCeleb1 test set, where CHNM denotes curriculum hard negative mining

Objective	Hyperparameters	VGG-M-40	Thin ResNet-34	Fast ResNet-34
Softmax	——	10.14±0:20	5.82±0.47	6.46±0.06
AM-Softmax	m = 0.1, s = 30	4.76±0.10	2.59±0:09	2.41±0.01
AAM-Softmax	**m = 0.2, s = 30**	**4.64±0.04**	**2.36±0:04**	**2.38±0.01**
Triplet	**m = 0.2, CHNM**	**4.67± 0.06**	**2.60±0:02**	**2.71±0.06**
GE2E	M = 3	4.40±0.08	2.52±0:07	2.37±0.10
Prototypical	**M = 2**	**4.59±0.02**	**2.34±0:08**	**2.32±0.02**
Angular Prototypical	M = 2	4.29±0.07	2.21±0.03	2.22±0.05

参考文献 Chung et al., In defence of metric learning for speaker recognition, in: Interspeech 2020.

四、总结

已有成果总结：

- 非端到端的分类损失需要引入减小类内方差的margin
- 端到端确认损失引入类中心学习可以增加训练稳定性、提高性能

可能的发展趋势：

- 新型的端到端确认损失
- 端到端确认损失与非端到端分类损失形成优势互补与融合
- 真正的端到端并不需要独立的back-end scoring

谢谢！

张晓雷

西北工业大学
智能声学与临境通信研究中心