On the influence of metric learning loss functions for robust self-supervised speaker verification to label noise

Abderrahim Fathan, Xiaolin Zhu*, Jahangir Alam
Computer Research Institute of Montreal (CRIM)
Montreal, Canada
abderrahim.fathan@crim.ca, alice.zhuxl@gmail.com, jahangir.alam@crim.ca

***Abstract*—While clustering-driven Pseudo-Labels (PLs) are commonly employed to optimize Speaker Embedding (SE) networks and facilitate training of self-supervised Speaker Verification (SV) systems, the efficacy of PL-based self-supervised training hinges on the accuracy of these generated labels. In this paper, we perform a large-scale comparative study of a wide range of recent metric learning loss functions for better generalization of self-supervised SV systems. In particular, we investigate the effect of these losses on the robustness of the self-supervised SV task against label noise using various real-life clustering-based PLs. We present an extensive comparative evaluation of the performance of these loss functions using different numbers of clusters and show that our proposed selection of loss functions is effective against label noise and leads to considerable improvements in SV performance. Moreover, using our selected losses combined with the adopted CAMSAT clustering algorithm-based PLs to train our SE system allows us to achieve state-of-the-art self-supervised SV performance. Code of our experiments will be made publicly available.**

***Index Terms*—Speaker verification, clustering, pseudo-labels, label noise, loss functions**

I. Introduction

Automatic speaker verification (ASV) consists of using the voiceprint of a speaker to verify their identity. ASV is one of the most convenient means of biometric recognition [1]. Based on a speaker's known utterances, the speaker verification (SV) task consists of confirming that the identity of a speaker is who they purport to be.

Fixed-dimensional embeddings are typically extracted at the utterance level from both enrollment and test speech samples. These embeddings are then fed into a scoring algorithm, such as cosine distance, to measure their similarity and determine the likelihood that they originate from the same speaker. Classically, the i-vector paradigm has been one of the most prominent approaches for speaker embedding [2], [3], owing to its capacity to capture the distributive patterns of the speech in an unsupervised fashion, even with a relatively small amount of training data. This framework generates fixed-sized compact vectors (i-vectors) that encapsulate the speaker's identity in a speech utterance, regardless of its duration. Moreover, in recent years, a plethora of deep learning-based architectures and techniques have been proposed to extract embeddings [4]–[6]. These approaches have demonstrated remarkable performance when a large amount of training data from a sufficient number of speakers is available [7]. A widely adopted architecture for this purpose is ECAPA-TDNN [8], renowned for achieving state-of-the-art (SOTA) performance in text-independent speaker recognition. The ECAPA-TDNN incorporates squeeze-and-excitation (SE), utilizes channel- and context-dependent statistics pooling, multi-layer aggregation, and employs self-attention pooling to obtain an utterance-level embedding.

Indeed, the majority of deep embedding models are trained under full supervision, necessitating large speaker-labeled datasets for effective training. However, creating well-annotated datasets can be a costly and time-intensive endeavor, prompting the research community to explore more affordable self-supervised learning (SSL) techniques utilizing extensive unlabeled datasets. A typical approach to address this issue for SV systems is to employ clustering schemes [5], [6], [9] to generate Pseudo-Labels (PLs) or other self-supervised objectives such as SimCLR or MoCo [10] to produce PLs used later in discriminative training [11], [12]. Although these PL-based Self-Supervised SV schemes exhibit striking performance, the efficacy of clustering continues to impede all aforementioned approaches [12], [13], primarily because downstream performance heavily relies on precise PLs. However, these PLs are generally noisy and inaccurate due to the mismatch between the clustering objective and the final SV task. Additionally, despite the advantages of iterative clustering-classification frameworks, the persistence of erroneous information from incorrect PLs degrades the final downstream task performance [12], [14]. Indeed, recent studies have demonstrated that label noise can significantly affect downstream performance [6]. Thus, the need for better-performing SV approaches that can withstand label noise to mitigate its negative effect on generalization. In this paper, we investigate a variety of metric learning loss functions, including maximum margin-based softmax losses, symmetric losses, normalized losses, mining-based softmax variants (e.g. CurricularFace, Focal loss), sample-to-sample based losses, and noise-robust losses for the task of SV under label noise. To this aim, we explore various recent clustering-

* Independent Researcher

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00186

based clustering algorithms (classical and deep models) to study the generalization and behavior of self-supervised SV systems under various types of real-world label noise. We propose a curated selection of loss objectives (see Table II) that we experimentally found to be effective against label noise and enhance the generalization of self-supervised SV systems to out-of-set samples, beyond discrepancies in the PLs. The contributions of this paper are as follows:

- We propose the first large-scale investigative and comparative study of various recent state-of-the-art loss objectives for the task of speaker verification, using various clustering algorithms. Several of these losses we apply for the first time in the domain of speaker verification.
- We show that maximum-margin -based softmax losses are beneficial to mitigate the memorization effects of label noise during training.
- We demonstrate that several recent maximum-margin softmax variants provide a great advantage in terms of generalization and noise-robustness over some widely-used losses in the domain of SV, such as the angular additive margin softmax (AAMSoftmax) [15] loss.
- Using CAMSAT-based PLs [16], our proposed selection of loss objectives allowed us to achieve SOTA SV performance, outperforming various benchmarks.

II. Background and Related Work

A. Noise-robust loss functions

We can broadly categorize methods for learning from noisy data into two groups: one focuses on developing noise-robust algorithms to learn directly from noisy labels [17]–[22], while the other aims at label-cleansing, which involves removing or correcting mislabeled data [23]–[25]. In recent years, various robust loss-based methods were proposed to learn with noisy labels. [26] proved theoretically that symmetric loss functions, such as Mean Absolute Error (MAE), are robust to label noise, while other losses like commonly used Cross Entropy (CE) are not. Besides, [27] introduced Generalized Cross Entropy (GCE), a generalized mixture of CE and MAE. [28] proposed Symmetric Cross Entropy (SCE) which is a combination of CE and scaled MAE. Reverse Cross Entropy (RCE) was also suggested to learn more distinguished feature representations for detecting adversarial examples. Additionally, [29] suggested a state-of-the-art Active Passive Loss (APL) to create fully robust loss functions. It showed that any loss function can be made robust to noisy labels by a simple normalization operation that makes loss functions symmetric. On the other hand, recently [30] found that APL still struggles with MAE and suffers from a problem of underfitting. For this reason, they suggested a new class of passive loss functions that are different from MAE, called Negative Loss Functions (NLFs), and proposed a new class of theoretically robust passive loss functions, called Normalized Negative Loss Functions (NNLFs). By replacing the MAE in APL with NNLF, they proposed an additional Active Negative Loss (ANL), a robust loss function framework with stronger fitting ability. In this paper, we investigate several robust loss functions created by the APL framework and NLFs, including the proposed normalization operation.

Moreover, in the domain of SV, [6] found that regularization through Mixup is effective against label noise memorization [31], and induces better generalization of self-supervised speaker verification systems since Mixup can dilute the label noise and create synthetic samples around the borders that lead to smoothing the data manifold and better class separation. In the same line of work, [32] also proposed an effective noise-robust self-supervised Multi-task learning framework based on various mixup variants to leverage the diverse complementary information that can be obtained by integrating various tasks, thereby enhancing the performance and robustness of speaker verification systems.

B. Maximum margin-based softmax loss objectives

The goal of Metric Learning is to learn representation functions that map objects into an embedded space. The aim is to simplify the comparison function of speaker utterances all the way down to the most simple distance function (e.g. cosine distance) by delegating the hard task of generating speaker representations to the trained embedding network which should ensure intra-class compactness and inter-class separability.

To improve performance on previously unseen data and generalize to out-of-domain speech samples, various maximum margin-based softmax variants based on different objectives have been proposed. Indeed, softmax suffers from several drawbacks such as that (1) its computation of inter-class margin is intractable [33] and (2) the learned projections are not guaranteed equi-spaced. Indeed, the projection vectors for majority classes occupy more angular space compared to minority classes [34]. To solve these problems, several alternatives to softmax have been proposed [15], [35]–[38]. For instance, AMSoftmax [35] loss applies an additive margin constraint in the angular space to the softmax loss for maximizing inter-class variance and minimizing intra-class variance. To provide a clear geometric interpretation of data samples and enhance the discriminative power of deep models, AAMSoftmax (angular additive margin softmax) [15] objective introduces an additive angular margin to the target angle (between the given features and the target center). Due to the exact correspondence between the angle and arc in the normalized hypersphere, AAMSoftmax can directly optimize the geodesic distance margin, thus its other name ArcFace.

Additionally, CosFace (large margin cosine loss) [38] reformulates the softmax loss as a cosine loss by L2 normalizing both features and weight vectors to remove radial variations, based on which a cosine margin term is introduced to further maximize the decision margin in the angular space. On the other hand, OCSoftmax [36] uses one-class learning instead of multi-class classification and does not assume the same distribution for all classes/speakers. More recently, AdaFace [37] loss has been proposed, emphasizing misclassified samples according to the quality of speaker embeddings (via feature norms). As an improvement, SMAFace was also introduced for low-quality face recognition images by incorporating sample mining into

TABLE I: A study of a wide variety of metric learning loss functions. Results are presented in terms of the EER (%) downstream SV evaluation performance. We used the CAMSAT algorithm to generate PLs using different predefined numbers of clustering.

Loss function	**No. of clusters**			**Loss function**	**No. of clusters**		
	5,000	**5,994**	**10,000**		**5,000**	**5,994**	**10,000**
OCSoftmax	**2.964**	3.134	2.969	Focal loss	13.001	13.340	12.561
Subcenter-ArcFace	2.969	3.059	2.943	Agent Center loss	13.34	13.393	12.508
SMAFace	3.028	3.192	3.033	Generalized Cross Entropy	13.351	13.277	13.966
AMSoftmax	3.054	3.224	2.959	Reverse Cross Entropy	14.252	14.687	14.555
AdaFace	3.059	3.112	3.059	Softmax	14.486	14.507	15.085
AAMSoftmax	3.065	3.309	3.134	AExp loss	14.565	14.973	14.756
ArcFace-VPL	2.996	3.059	2.996	AGCE loss	14.464	14.390	14.608
CosFace-VPL	3.075	**3.022**	2.948	AUE loss	14.666	14.947	14.772
CosFace	3.096	3.043	**2.863**	Normalized Cross Entropy	18.664	19.692	20.594
Unified Cross Entropy (UniFace) loss	3.15	3.208	3.16	Normalized Focal loss	18.754	19.565	20.700
Normalized BCE loss	3.213	3.181	3.192	Normalized Cross Entropy	18.664	19.692	20.594
Normalized Softmax loss	3.134	3.118	3.028	Hard Gumbel-Softmax	23.096	47.397	22.778
CurricularFace	3.229	3.256	3.192	Normalized Negative Cross Entropy	23.261	26.156	27.45
Cross Entropy	5.477	5.827	5.546	Normalized Negative Focal loss	22.969	24.146	25.779
AS-Softmax	5.748	6.272	6.607	Soft Gumbel-Softmax	25.774	43.871	22.683
MagFace	8.499	8.409	3.139	Center loss	27.126	29.173	27.625
DropMax	7.137	7.264	8.006	Unified Threshold Integrated Sample-to-Sample (UniTSFace) loss	36.49	36.437	36.972
Symmetric Cross Entropy	12.773	13.266	13.091	Sparsemax	42.179	42.54	46.124

conventional margin-based methods. At its core, SMAFace focuses on prioritizing information-dense samples, namely hard samples or easy samples, which present more distinctive features. To this aim, it employs a probability-driven mining strategy, enabling the model to adeptly navigate hard samples, thereby bolstering its robustness and adaptability. Besides, as softmax has no unified threshold to separate positive sample-to-class pairs from negative sample-to-class pairs, a Unified Cross Entropy (UniFace) [39] loss for face recognition model training was designed on the vital constraint that all the positive sample-to-class similarities shall be larger than the negative ones. Additionally, as sample-to-class loss-based models cannot explore the full relationships between samples across large datasets, UniTSFace [40] proposed a unified threshold integrated sample-to-sample based loss (USS), which introduces an explicit unified threshold for distinguishing positive pairs from negatives. Furthermore, to incorporate additional sample-to-sample comparisons during training, [41] proposed Variational Prototype Learning (VPL), which represents every class as a distribution instead of a point in the latent space. Identifying the slow feature drift phenomenon, authors directly injected memorized features into prototypes to approximate variational prototype sampling. Finally, as above methods are susceptible to label noise, Subcenter-ArcFace [42] relaxes the intra-class constraint of ArcFace by designing K sub-centers for each class to improve the robustness to label noise. In this case, the training sample only needs to be close to any of the K positive sub-centers instead of the only one positive center.

III. Experimental setup

For all our clustering algorithms, we use 400-dimensional i-vectors as condensed input. They serve as unsupervised representations of speakers and enable more efficient clustering by mitigating the high dimensionality associated with MFCC acoustic features.

We assess the performance of our examined loss functions and the resulting pseudo-labels (PLs) for self-supervised speaker verification through a series of experiments conducted on the VoxCeleb2 dataset [43]. We train the embedding networks on the development subset of VoxCeleb2, comprising 1.092 million utterances from 5,994 distinct speakers. Evaluation follows the VoxCeleb1 trials list [44], encompassing 37,720 trials with 4,874 utterances from 40 speakers. For our speaker verification (SV) system, we employ 40-dimensional Mel-frequency cepstral coefficients (MFCCs) as input features to our ECAPA-TDNN model. MFCCs are computed every 10 ms with a 25 ms Hamming window, using the Kaldi toolkit [45].

Moreover, to follow other SV works in training the ECAPA-TDNN-based systems, we have applied data augmentation at the waveform level, such as additive noise and room impulse response (RIR) simulation, as described in [7]. Furthermore, we extended augmentation to the extracted MFCCs features, following a similar approach to the specaugment scheme [46]. All speaker verification experiments were run over 7 days on a single RTX2080Ti GPU, utilizing a batch size of 200 MFCC samples. All margin-based losses are run with a scale factor (s) set to 30 and the angular margin (m) to 0.2. Cosine similarity serves as the backend for scoring verification between embeddings of enrollment and test speech samples.

A. *Our clustering-based self-supervised speaker embedding framework*

Figure 1 depicts a schematic diagram of our general clustering-based self-supervised speaker verification process that we follow throughout the paper. During our work, we explore various loss functions and clustering algorithms and conduct different analyses on their impact on the robustness of speaker verification performance. We employ ECAPA-TDNN as our speaker embedding network and use our adopted loss

TABLE II: A study of various margin-based softmax losses for better generalization of our ECAPA-TDNN -based SV system, using different pseudo-labels. Results are presented in terms of the EER (%) downstream SV evaluation performance.

Pseudo-labels	No. of clusters	AAMSoftmax [15]	AMSoftmax [35]	OCSoftmax [36]	AdaFace [37]	CosFace [38]	Subcenter-ArcFace [42]	UniFace [39]	SMAFace	Cross Entropy
True labels	5,994	1.437	1.522	1.416	**1.326**	1.463	1.400	1.421	1.373	3.489
GMM	5,000	5.429	4.851	4.682	5.095	4.862	**4.639**	5.016	4.857	8.425
AHC	5,000	3.621	3.664	3.584	3.526	3.6	3.669	3.590	**3.499**	6.479
IMSAT	5,000	4.507	4.141	3.881	**3.807**	4.083	3.886	4.290	4.120	7.206
	5,994	4.146	3.961	3.892	4.008	**3.696**	3.828	4.099	3.971	7.333
	10,000	4.438	4.024	**4.003**	4.046	4.072	4.035	4.077	4.019	7.370
$L_{IMSAT} + L_{SupCon}$	5,000	4.623	4.401	4.396	4.576	4.48	**4.226**	4.687	4.337	7.179
	5,994	4.475	4.502	4.491	4.443	4.427	**4.369**	4.687	4.518	7.179
	10,000	4.348	4.221	4.189	4.343	4.173	**3.993**	4.305	4.173	7.174
$L_{IMSAT} + L_{Mixup} + L_{SupCon}$	5,000	4.231	4.046	3.924	4.056	4.332	**3.860**	4.433	4.067	7.391
	5,994	4.321	4.146	4.024	4.125	4.199	**3.971**	4.067	4.024	7.28
	10,000	4.252	4.03	4.146	4.21	4.093	3.993	4.051	**3.945**	7.259
$L_{IMSAT} + L_{Mixup} + L_{SupCon} + L_{aug}$	5,994	3.298	2.974	3.049	2.985	3.155	**2.932**	3.478	3.139	6.272
	10,000	3.377	3.287	3.293	3.399	3.298	3.309	3.383	**3.118**	5.695
$L_{IMSAT} + L_{Mixup} + L_{SupCon} + L_{aug} + L_{InfoNCE}$	5,994	3.245	2.985	2.943	3.017	3.176	**2.816**	3.176	3.112	6.098
	10,000	3.362	**3.001**	3.006	3.043	3.181	3.086	3.150	3.038	5.801
CAMSAT	5,000	3.065	3.054	**2.964**	3.059	3.096	2.969	3.150	3.028	5.477
	5,994	3.309	3.224	3.134	3.112	**3.043**	3.059	3.208	3.192	5.827
	10,000	3.134	2.959	2.969	3.059	**2.863**	2.943	3.160	3.033	5.546

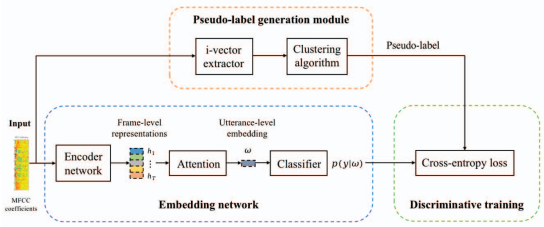


Fig. 1: General training scheme of our clustering generated pseudo-label-based self-supervised speaker embedding networks.

objectives to train this system using pseudo-labels generated by the different clustering algorithms.

B. Clustering-based pseudo-label generation

We have utilized the Kaldi toolkit [45] to extract i-vector embeddings [2], [3] for clustering purposes. These i-vectors represent statistical unsupervised fixed-dimensional representations extracted from each training utterance. Subsequently, clustering was performed on these embeddings. Once the clustering algorithms were trained, we assigned each utterance to the corresponding cluster and used the cluster-id as a PL. These clustering-based PLs enabled us to train the speaker embedding network using our metric learning loss objectives, mimicking supervised learning.

For a thorough comparison, we have set the number of clusters to be in {5000, 5994, 10000} to study the influence of the predefined number of clusters on our studied losses and on the downstream speaker verification performance (5994 is the ground truth number of speakers).

C. Clustering performance of our pseudo-labels

Table III shows the clustering performance of our employed clustering-based pseudo-labels using different clustering algorithms or self-supervised learning-based objectives to generate these pseudo-labels. From the relatively low accuracy and mutual information scores, we can see that our obtained cluster assignments are often noisy and impure, leading to discrepancies between the PLs and the true speaker identities. As a result, in several cases, our SV performance was degraded from overfitting this label noise.

TABLE III: The clustering performance of our employed clustering-based pseudo-labels using different clustering algorithms or combining self-supervised learning-based objectives.

Pseudo-labels	No. of clusters	ACC	NMI
GMM	5,000	0.45	0.747
AHC	5,000	0.602	0.838
IMSAT	5,000	0.578	0.822
	5,994	0.6	0.833
	10,000	0.621	0.844
$L_{IMSAT} + L_{SupCon}$	5,000	0.497	0.784
	5,994	0.514	0.793
	10,000	0.548	0.813
$L_{IMSAT} + L_{Mixup} + L_{SupCon}$	5,000	0.602	0.836
	5,994	0.619	0.846
	10,000	0.639	0.86
$L_{IMSAT} + L_{Mixup} + L_{SupCon} + L_{aug}$	5,994	0.69	0.884
	10,000	0.714	0.894
$L_{IMSAT} + L_{Mixup} + L_{SupCon} + L_{aug} + L_{InfoNCE}$	5,994	0.702	0.889
	10,000	**0.725**	**0.9**
CAMSAT	5,000	0.655	0.874
	5,994	0.669	0.878
	10,000	0.709	0.889

D. Clustering Evaluation Metrics

Following the commonly used evaluation metrics for clustering, we evaluate our studied clustering models by assessing the quality of their generated pseudo-labels using the following two supervised clustering metrics:

- **Unsupervised Clustering Accuracy (ACC)**: Based on the Hungarian algorithm [47] to efficiently find the optimal mapping between labels and the generated PLs, ACC evaluates the agreement between the true labels and the PLs. $ACC = \underset{m}{max} \frac{\sum_{i=1}^{N} \mathbb{1}\{y_i = m(c_i)\}}{N}$ where y_i is the true label, c_i is the generated PL assignment, and m is a mapping function that ranges over all possible one-to-one mappings between true labels and assignments.

- **Normalized Mutual Information (NMI)** [48]: $NMI(Y,C) = \frac{I(Y,C)}{\frac{1}{2}[H(Y)+H(C)]}$ where Y and C denote the ground-truth labels and the clustering assignments, respectively. H is the entropy function and I denotes the MI metric.

E. Details of our adopted clustering algorithms

For clustering, we adopt the same CAMSAT clustering approach used in [16] to generate pseudo-labels. CAMSAT is founded on mixing augmentations and self-augmented training. The goal is to impose invariance to data augmentation on the output predictions of deep models in an end-to-end manner. Simultaneously, the approach aims to maximize the information-theoretic dependency between samples and their predicted cluster assignments (discrete representations). It provided both state-of-the-art speaker clustering and speaker verification performance. In this paper, we try to investigate several metric learning loss functions to enhance the generalization performance of self-supervised speaker embedding systems and to mitigate the negative effect of heavy noise in the generated pseudo-labels (PLs) used to train these systems. For comparison, we also include 2 classical clustering models, namely Gaussian mixture model (GMM) and Agglomerative Hierarchical Clustering (AHC) [49] which have also demonstrated great performance. Please refer to [16] for further details about the clustering model architecture and training details. The following list provides a brief description of the self-supervised learning-based objectives used for clustering in our experiments:

- **Mixup Loss (Virtual Mixup Training [50])**: $L_{mixup} = \frac{1}{N}\sum_{i=1}^{N} KL(\alpha_i p_i + (1-\alpha_i)p_{r_i} || f(\alpha_i x_i + (1-\alpha_i)x_{r_i})$. $r_i \in \{1,..,N\}$ is a random index, and $\alpha_i \in [0,1]$ is the interpolation coefficient. $KL(.||.)$ operator denotes the Kullback-Leibler divergence and N is the mini-batch size. $p_i = f(x_i) \in \mathbb{R}^{1xC}$, $p_{r_i} = f(x_{r_i})$ correspond to the predictions of original clean data samples x_i and x_{r_i}. C is the predefined number of clusters.
- **Information Maximizing Self-Augmented Training (IMSAT) Loss**: IMSAT loss L_{IMSAT} [51] maximizes mutual information (MI) in an end-to-end fashion between data and their clustering assignments by encouraging the prediction of the neural network to remain invariant under data perturbation/augmentation, while maximizing the information-theoretic dependency between data and their predicted discrete representations. It minimizes the following objective:
  $L_{IMSAT} = R_{SAT}(\theta, T_{VAT}) + \lambda(H(Y|X) - \mu H(Y))$
  R_{SAT} is a loss term that encourages the representations of augmented samples to approach those of the original samples. Additionally, it helps to regulate the complexity of the network against local perturbations through Virtual Adversarial Training (VAT) [52]. $T_{VAT}(x) = x + r$ is the augmentation function using local perturbations to enforce invariance where $r = \arg\max_{r'}\{R_{SAT}(\hat{\theta}; x, x + r'); \|r'\|_2 \leq \epsilon\}$ is an adversarial direction. $H(.)$ and $H(.|.)$ denote the marginal and conditional entropy, respectively. Their difference represents the MI between input X and label Y, which we aim to maximize. $\hat{\theta}$ are the current parameters of the model's network. Hyper-parameters $\lambda, \mu \in \mathbb{R}$ control the trade-offs between the complexity regularization of the model (via R_{SAT}) and the MI maximization, and between the two entropy terms, respectively.
- **Data augmentation loss** L_{aug}: L_{aug} constrains the predicted representations of augmented samples to closely resemble those of the original data points by minimizing the KL-divergence between both predictions, as follows:

  $$L_{aug} = \frac{1}{N}\sum_{i=1}^{N} KL(p_i^{aug_{r_i}} || p_i) \qquad (1)$$

  with $J = \{aug_1, ..., aug_{|J|}\}$ is the ensemble of available augmentations and $r_i \in \{1,..,|J|\}$ denotes a random augmentation from J. $KL(.||.)$ denotes the Kullback-Leibler divergence operator, and N is the mini-batch size. $p_i = f(x_i) \in \mathbb{R}^{1xC}$, and $p_i^{aug_j} = f(x_i^{aug_j})$ correspond to the predictions of data sample x_i and its augmented version $x_i^{aug_j}$, respectively.
- **Contrastive Self-Supervised Learning (InfoNCE)**: InfoNCE [53], where NCE stands for Noise-Contrastive Estimation, is a type of contrastive loss function used for self-supervised learning in SimCLR [54], also known as the NT-Xent loss (Normalized Temperature-scaled Cross Entropy). The goal is to maximize the similarity between the representations of two augmented versions of the same input, i.e., Z_i and Z_j while minimizing it to all other examples in the batch.
  In short, the InfoNCE loss compares the similarity of Z_i and Z_j to the similarity of Z_i to any other representation in the batch by performing a softmax over the similarity values. The InfoNCE loss $l_{i,j}$ for pair (i,j) can be written as follows:
  $l_{i,j} = -log\frac{\exp sim(Z_i,Z_j)/\tau}{\sum_{k=1}^{2N} \mathbb{1}_{k\neq i} \exp sim(Z_i,Z_k)/\tau}$.
  $\mathbb{1}_{k\neq i} \in \{0,1\}$ is an indicator function evaluating to 1 iff $k \neq i$, and $\tau = 1$ denotes the temperature parameter. The final $L_{InfoNCE}$ loss is computed across all positive pairs, both (i, j) and (j, i), in a mini-batch (a sample and its augmented version). Cosine similarity is used as a similarity metric, defined as: $sim(Z_i, Z_j) = \frac{{Z_i}^T.Z_j}{\|Z_i\|\|Z_j\|}$.
- **Supervised Contrastive Loss (SupCon)**:
  L_{SupCon} from [55] extends the self-supervised batch contrastive approach of the NT-Xent loss (Normalized Temperature-scaled Cross Entropy) [54] to the fully-supervised setting, allowing us to effectively leverage label information. For that, clusters of points belonging to the same class are pulled together in normalized embedding space, while simultaneously pushing apart clusters of samples from different classes. The SupCon extension allows for multiple positives per anchor instead of a single sample in addition to many negatives, and draws from samples of the same class as the anchor, rather than being data augmentations of the anchor, as done in previous works. It showed benefits for robustness to

natural corruptions and is more stable to hyperparameter settings such as optimizers and data augmentations.

Since the SupCon loss requires labels, we use online generated labels as input labels to the SupCon loss function, which allows us to use it in a completely unsupervised fashion without the need for ground-truth labels. We use the implementation from https://github.com/wangz10/contrastive_loss/blob/master/losses.py with temperature=1 and base_temperature=1.

IV. Results and Discussion

In Table I, we performed a large-scale study of 37 metric learning loss functions including all the above-mentioned families of loss objectives and other widely used losses using CAMSAT-based pseudo-labels.

Besides, to further validate the noise robustness and generalization of these losses, in Table II we summarize our results using various other clustering-based pseudo-labels (with different predefined numbers of clusters) employed to train our SV model using the selection of our best-performing loss functions in Table I.

Throughout our experiments, we can observe that incorporating a margin can easily enhance the performance of our metric learning loss functions. Results show clearly that our selection of maximum-margin softmax variants in Table II are very effective in improving the generalization of our speaker verification systems across all types of label noise contained in the PLs. In particular, unlike the widely used AAMSoftmax loss in speaker verification, to our knowledge, our results indicate for the first time that variants such as OCSoftmax using one-class learning instead of multi-class classification and not assuming the same distribution for all speakers (which is more realistic in our case), or the recent AdaFace and SMAFace losses, perform consistently better across all pseudo-labels and the ground truth labels. Indeed, AAMSoftmax is susceptible to massive label noise [15]. This is because if a training sample is noisy (misclassified), it does not belong to the intended positive class. In AAMSoftmax, such a noisy sample produces a significant erroneous loss value, which negatively impedes the model training. This partially explains the underperformance of AAMSoftmax compared to other variants when using PLs for training. Interestingly, thanks to its design to be robust to label noise, we can also observe the good performance of Subcenter-ArcFace, which often outperforms all other losses across our various studied PLs.

Besides, in our experiments on the VoxCeleb1-O test set, sample-to-sample loss functions and other losses such as MagFace, DropMax, Center loss, Softmax, Gumbel-Softmax and Sparsemax performed poorly and seem to suffer from serious problems of convergence, numerical instability, or sensitivity to hyperparameters. On the other hand, we can observe that the normalization operation to make our losses symmetric helped us to improve performance in the case of Softmax and Binary CE (BCE). Finally, we found, as shown in Table I, that recently proposed NLFs and NNLFs losses both performed poorly in our case compared to our suggested maximum-margin softmax-based variants.

Moreover, using different predefined numbers of clusters including the ground truth number of clusters, we can see that the final downstream SV evaluation performance depends more on the quality of the PLs, and that the consideration of the predefined number of clusters is less important.

TABLE IV: A comparison of several SOTA Self-Supervised SV approaches to our simple SV system trained with CAMSAT-based PLs and Subcenter-ArcFace loss. All approaches employ the same ECAPA-TDNN underlying model. Results are presented on the original VoxCeleb1 test set (Voxceleb1_O).

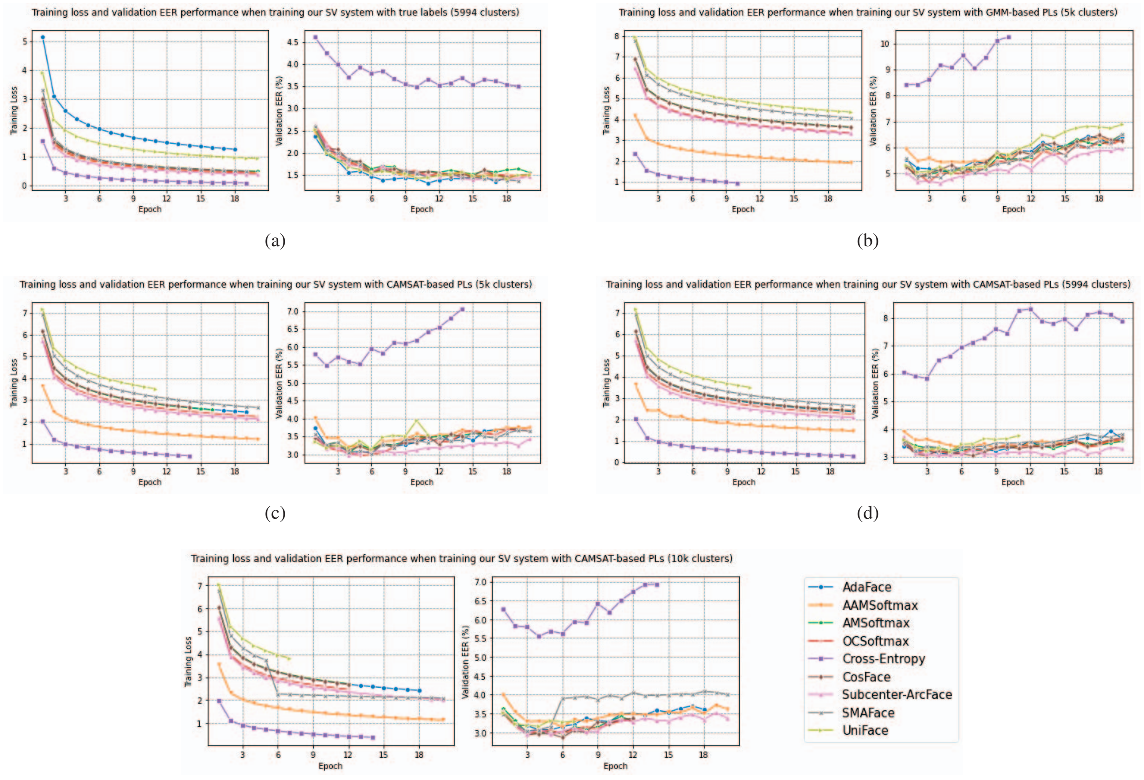
SSL Objective	EER (%)
MoBY [10]	8.2
InfoNCE [12]	7.36
MoCo [56]	7.3
ProtoNCE [10]	7.21
PCL [10]	7.11
CA-DINO [57]	3.585
i-mix [58]	3.478
l-mix [58]	3.377
Iterative clustering [12]	3.09
CAMSAT [16]	3.065
Our approach (using Subcenter-ArcFace)	**2.816**

Finally, Table IV shows a comparison of our approach for Self-Supervised SV training using CAMSAT-based PLs and our best-performing Subcenter-ArcFace loss, compared to recent SOTA self-supervised SV approaches employing diverse SSL objectives employing the same underlying ECAPA-TDNN model encoder. The results demonstrate clearly that our approach provides very competitive performance close to the supervised baseline while being simple and fast. Besides, our approach outperforms all the baselines, which suggests that the consideration of loss functions is still crucial and that simply refining the loss objectives of existing self-supervised speaker recognition systems can still provide further enhancements.

A. Behaviour of metric learning losses over epochs

In Figure 2, we study the evolution of the downstream evaluation EER (%) performance and the training loss of our system trained with our selection of maximum-margin-based loss functions. In particular, we perform the same experiments using the original ground-truth labels to suppress the effect of label noise and study its impact on the generalization and training of SV systems. First of all, despite the good generalization of our SV systems, we can observe that these losses still suffer from overfitting and from the phenomenon of label noise memorization [31] when training with noisy pseudo-labels (training with noisier GMM-based PLs performs worse than with the more accurate CAMSAT-based PLs and leads to more degradation of the validation performance).

Indeed, due to memorization effects [31], deep networks, especially overparameterized models, initially learn simple (clean) patterns in the PLs. Over time, they progressively overfit more challenging and complex (noisy) patterns. This induces the model to overfit the noise/corruption present in the training



(a)

(b)

(c)

(d)

(e)

Fig. 2: Training loss and validation performance over time of our speaker verification (SV) systems trained under various loss functions using (a) ground-truth labels (b) GMM-based pseudo-labels (c) CAMSAT-based pseudo-labels (5k clusters) (d) CAMSAT-based pseudo-labels (5994 clusters) and (e) CAMSAT-based pseudo-labels (10k clusters).

PLs, which ultimately causes a gradual decline in the validation curve. This highlights the importance of having highly accurate PLs for good generalization of self-supervised SV systems. Very interestingly, on the contrary to other losses where validation performance starts to degrade after only the first few epochs, we can find experimentally that Subcenter-ArcFace is more robust to label noise and does suffer the least from overfitting compared to other losses. It is worth mentioning, however, that Subcenter-ArcFace remains much slower than other losses due to its use of a much bigger matrix of subcenters.

Finally, our visualizations in Figure 2 with different metric learning loss functions and our large-scale study also demonstrate that producing compact cluster assignments (compact probabilities) with more discriminative ability does not really help to mitigate memorization of label-noise. Despite inducing better generalization to out-of-set samples, maximum-margin softmax losses do not seem to reduce sufficiently the model's ability to accommodate random noise during training.

V. Conclusion

In this work, we performed a large-scale comparative study of a wide range of recent metric learning loss functions for better generalization of Self-Supervised Speaker Verification (SSSV) systems. In particular, we investigated the effect of these losses on the robustness of the SSSV task against label noise using various real-life clustering-based pseudo-labels, and proposed a selection of loss functions against label noise that lead to considerable improvements in self-supervised SV performance.

VI. Acknowledgment

The authors wish to acknowledge the funding from the Government of Canada's New Frontiers in Research Fund through grant NFRFR-2021-00338 and Natural Sciences and Engineering Research Council of Canada through grant RGPIN-2019-05381.

References

[1] John H.L. Hansen and Taufiq Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal Processing Magazine*, vol. 32, 2015.

[2] Najim Dehak et al., "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, 2011.

[3] Patrick Kenny, "A Small Footprint I-vector Extractor," in *Odyssey*, 2012, pp. 1–6.

[4] Z. Bai and X. L. Zhang, “Speaker recognition based on deep learning: An overview,” *Neural Networks*, 2021.

[5] Woo Hyun Kang, Jahangir Alam, and Abderrahim Fathan, “l-mix: a latent-level instance mixup regularization for robust self-supervised speaker representation learning,” *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[6] Abderrahim Fathan, Jahangir Alam, and Woohyun Kang, “On the impact of the quality of pseudo-labels on the self-supervised speaker verification task,” in *NeurIPS 2022 Second ENLSP Workshop*, 2022.

[7] David Snyder et al., “X-vectors: Robust dnn embeddings for speaker recognition,” in *Proc. of IEEE ICASSP*, 2018, pp. 5329–5333.

[8] Brecht Desplanques et al., “ECAPA-TDNN: emphasized channel attention, propagation and aggregation in TDNN based speaker verification,” in *Interspeech 2020*. ISCA.

[9] Woo Hyun Kang, Jahangir Alam, and Abderrahim Fathan, “An analytic study on clustering-based pseudo-labels for self-supervised deep speaker verification,” in *SPECOM*, 2022.

[10] Wei Xia et al., “Self-supervised text-independent speaker verification using prototypical momentum contrastive learning,” in *ICASSP*. IEEE, 2021.

[11] Junyi Peng et al., “Progressive Contrastive Learning for Self-Supervised Text-Independent Speaker Verification,” in *Proc. of Odyssey Workshop*, 2022.

[12] Ruijie Tao et al., “Self-supervised speaker recognition with loss-gated learning,” in *ICASSP*. IEEE, 2022.

[13] B. Han, Z. Chen, and Y. Qian, “Self-supervised speaker verification using dynamic loss-gate and label correction,” *arXiv preprint arXiv:2208.01928*, 2022.

[14] Yunfan Li et al., “Contrastive clustering,” in *AAAI*, 2021.

[15] Jiankang Deng et al., “Arcface: Additive angular margin loss for deep face recognition,” *IEEE TPAMI*, 2021.

[16] Abderrahim Fathan and Jahangir Alam, “Camsat: Augmentation mix and self-augmented training clustering for self-supervised speaker recognition,” in *IEEE Automatic Speech Recognition and Understanding (ASRU) Workshop*, 2023.

[17] Eyal Beigman and Beata Beigman Klebanov, “Learning with annotation noise,” in *Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, 2009, pp. 280–287.

[18] Melody Guan et al., “Who said what: Modeling individual labelers improves classification,” in *Proc. of the AAAI conference on artificial intelligence*, 2018, vol. 32.

[19] David Rolnick, Andreas Veit, et al., “Deep learning is robust to massive label noise,” *ICLR*, 2018.

[20] Armand Joulin, Laurens van der Maaten, Allan Jabri, and Nicolas Vasilache, “Learning visual features from large weakly supervised data,” in *European Conference on Computer Vision*. Springer, 2016, pp. 67–84.

[21] Ishan Misra et al., “Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels,” in *Proc. of the IEEE-CVPR conference*, 2016, pp. 2930–2939.

[22] Davood Karimi et al., “Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis,” *Medical Image Analysis*, vol. 65, 2020.

[23] Carla E Brodley and Mark A Friedl, “Identifying mislabeled training data,” *Journal of artificial intelligence research*, 1999.

[24] Sainbayar Sukhbaatar et al., “Training convolutional networks with noisy labels,” *arXiv preprint arXiv:1406.2080*, 2014.

[25] Andreas Veit et al., “Learning from noisy large-scale datasets with minimal supervision,” in *Proc. of the IEEE conference on CVPR*, 2017.

[26] Aritra Ghosh et al., “Robust loss functions under label noise for deep neural networks,” in *Proceedings of the AAAI conference on artificial intelligence*, 2017, vol. 31.

[27] Z. Zhang and M. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” *Advances in neural information processing systems*, vol. 31, 2018.

[28] Yisen Wang et al., “Symmetric cross entropy for robust learning with noisy labels,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 322–330.

[29] X. Ma, H. Huang, Y. Wang, et al., “Normalized loss functions for deep learning with noisy labels,” in *International conference on machine learning*. PMLR, 2020, pp. 6543–6553.

[30] Xichen Ye et al., “Active negative loss functions for learning with noisy labels,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[31] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, et al., “A closer look at memorization in deep networks,” in *International conference on machine learning*. PMLR, 2017, pp. 233–242.

[32] Abderrahim Fathan, Jahangir Alam, and Xiaolin Zhu, “Multi-task learning over mixup variants for the speaker verification task,” in *International Conference on Speech and Computer*. Springer, 2023, pp. 446–460.

[33] G. F. Elsayed et al., “Large margin deep networks for classification,” 2018.

[34] W. Liu, Y. Wen, et al., “Large-margin softmax loss for convolutional neural networks.,” in *ICML*, 2016, vol. 2.

[35] F. Wang et al., “Additive margin softmax for face verification,” *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[36] You Zhang et al., “One-class learning towards synthetic voice spoofing detection,” *IEEE Signal Processing Letters*, 2021.

[37] Minchul Kim, Anil K Jain, and Xiaoming Liu, “Adaface: Quality adaptive margin for face recognition,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 18750–18759.

[38] Hao Wang et al., “Cosface: Large margin cosine loss for deep face recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[39] J. Zhou, X. Jia, Q. Li, L. Shen, and J. Duan, “Uniface: Unified cross-entropy loss for deep face recognition,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 20730–20739.

[40] Qiufu Li, Xi Jia, Jiancan Zhou, et al., “Unitsface: Unified threshold integrated sample-to-sample loss for face recognition,” *arXiv preprint arXiv:2311.02523*, 2023.

[41] Jiankang Deng, Jia Guo, Jing Yang, Alexandros Lattas, and Stefanos Zafeiriou, “Variational prototype learning for deep face recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11906–11915.

[42] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, “Sub-center arcface: Boosting face recognition by large-scale noisy web faces,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, Proceedings, Part XI 16*. Springer, 2020, pp. 741–757.

[43] J. S. Chung, A. Nagrani, and A. Zisserman, “Voxceleb2: Deep speaker recognition,” in *INTERSPEECH*, 2018.

[44] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: a large-scale speaker identification dataset,” in *INTERSPEECH*, 2017.

[45] Daniel Povey et al., “The kaldi speech recognition toolkit,” in *In IEEE 2011 workshop*, 2011.

[46] Daniel S. Park et al., “Specaugment: A simple data augmentation method for automatic speech recognition.,” in *Interspeech*, 2019, pp. 2613–2617.

[47] Harold W Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics (NRL)*, vol. 52, no. 1, pp. 7–21, 2005.

[48] Pablo A Estévez et al., “Normalized mutual information feature selection,” *IEEE Transactions on neural networks*, 2009.

[49] William H. E. Day et al., “Efficient algorithms for agglomerative hierarchical clustering methods,” *Journal of Classification*, vol. 1, pp. 7–24, 1984.

[50] Xudong Mao, Yun Ma, Zhenguo Yang, Yangbin Chen, and Qing Li, “Virtual mixup training for unsupervised domain adaptation,” *arXiv preprint arXiv:1905.04215*, 2019.

[51] Weihua Hu et al., “Learning discrete representations via information maximizing self-augmented training,” in *International conference on machine learning*. PMLR, 2017, pp. 1558–1567.

[52] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii, “Virtual adversarial training: a regularization method for supervised and semi-supervised learning,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1979–1993, 2018.

[53] Aaron van den Oord, Yazhe Li, et al., “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.

[54] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[55] Prannay Khosla, Piotr Teterwak, et al., “Supervised contrastive learning,” *NeurIPS*, 2020.

[56] Jejin Cho et al., “The jhu submission to voxsrc-21: Track 3,” *arXiv preprint arXiv:2109.13425*, 2021.

[57] Bing Han et al., “Self-supervised learning with cluster-aware-dino for high-performance robust speaker verification,” *arXiv preprint arXiv:2304.05754*, 2023.

[58] Abderrahim Fathan and Jahangir Alam, “On the influence of the quality of pseudo-labels on the self-supervised speaker verification task: a thorough analysis,” in *IWBF*. IEEE, 2023.