The DKU-MSXF Speaker Verification System for the VoxCeleb Speaker Recognition Challenge 2023

——Track 3

Ze Li[1*], *Yuke Lin*[1*], *Xiaoyi Qin*[1*], *Ning Jiang*[2], *Guoqing Zhao*[2], *Ming Li*[1]

[1]Data Science Research Center, Duke Kunshan University, Kunshan, China
[2]Mashang Consumer Finance Co., Ltd, China

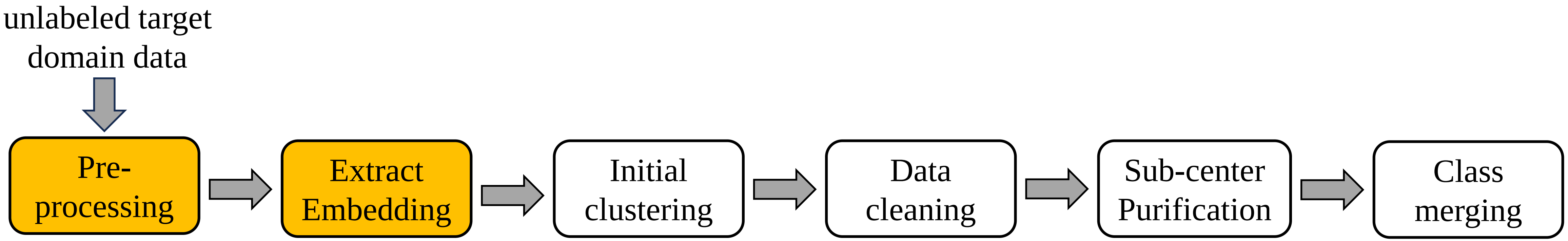
Overview

- Pre-training
  - Source domain data
- Pseudo-labeling
  - A novel method based on triple thresholds and sub-center purification
- Fine-tuning
  - Pseudo-labeled and ground-truth target domain data
- Score Calibration and Normalization
  - AS-Norm & QMF

Pre-training

- Data usage： Voxceleb2 dev （1092009 utts | 5994 spks）
- Speaker embedding model：
  - SimAM-Resnet100-ASP
  - ResNet100-TSP
  - SimAM-ResNet100-ASP
  - ResNet152-ASP
  - ResNet152-Stat
- Data augmentation：
  - Online 3-fold speed perturbation (Spk Aug)
  - On-the-fly data augmentation (Noise/RIR/Tempo/Vol)
- Loss function： ArcFace (m=0.2 , s=32)

Pseudo-labeling



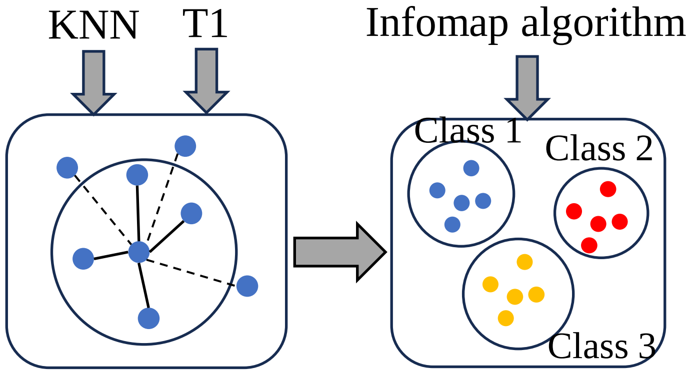
- Removing audios whose duration is less than 1 second.
  - Too short duration audio may not contain text information.
- Extracting speaker embeddings using the pre-trained speaker model.

Pseudo-labeling

unlabeled target domain data

Pre-processing → Extract Embedding → **Initial clustering** → Data cleaning → Sub-center Purification → Class merging

1. Generating the graph using the K-Nearest Neighbors algorithm.
2. Removing the edges with weights less than threshold T1.
3. Infomap algorithm is employed for initial clustering.



Initial clustering

Threshold T1

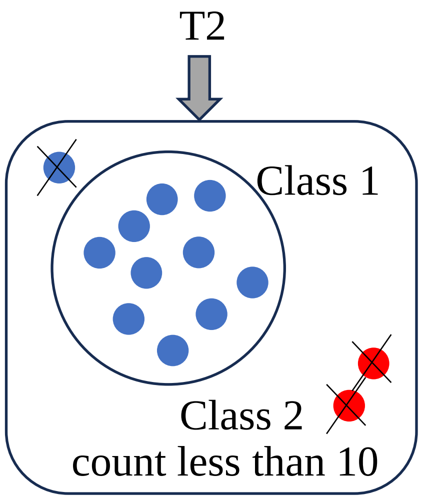
1. Computing the cosine similarity between each embedding and all other embeddings from labeled target domain data.
2. Recording the cosine similarity value between each embedding and the first embedding with a different label, and selecting the maximum one as threshold T1.

Pseudo-labeling

unlabeled target domain data

Pre-processing → Extract Embedding → Initial clustering → **Data cleaning** → Sub-center Purification → Class merging

1. Removing outliers data within each class based on threshold T2.
2. Eliminating classes with a data count of less than 10.

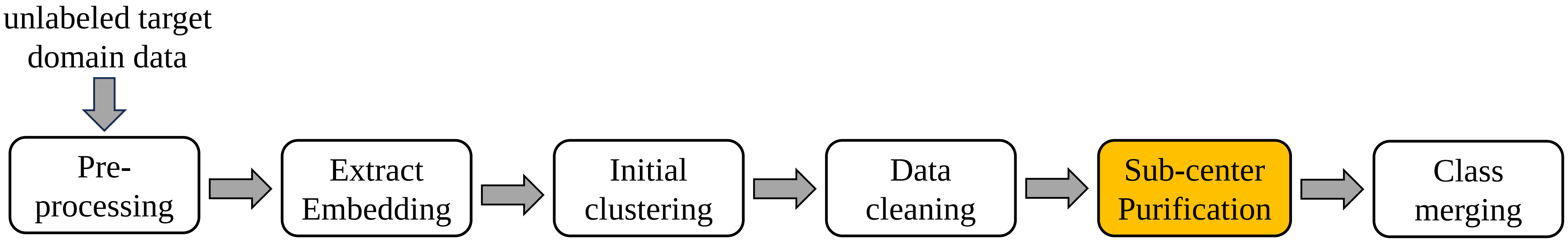


Data cleaning

Threshold T2

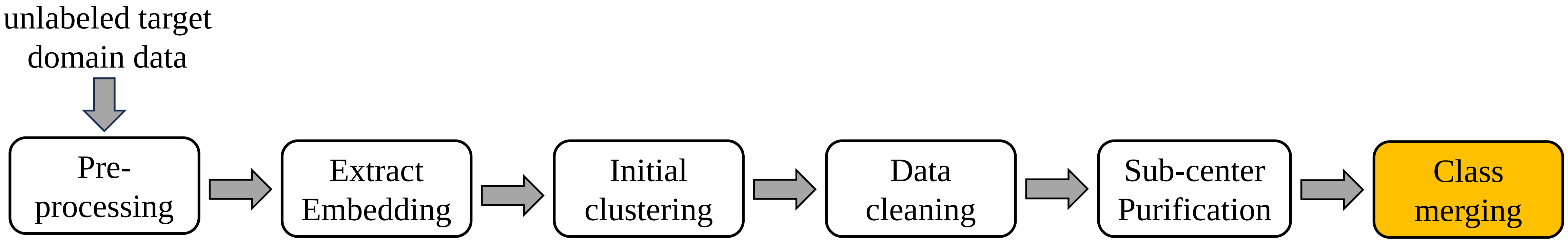
1. Calculating the cosine similarity between the embedding of each class and its respective centroid vector from labeled target domain data.
2. Then selecting the maximum value from the minimum cosine similarity values of each class as T2.

Pseudo-labeling

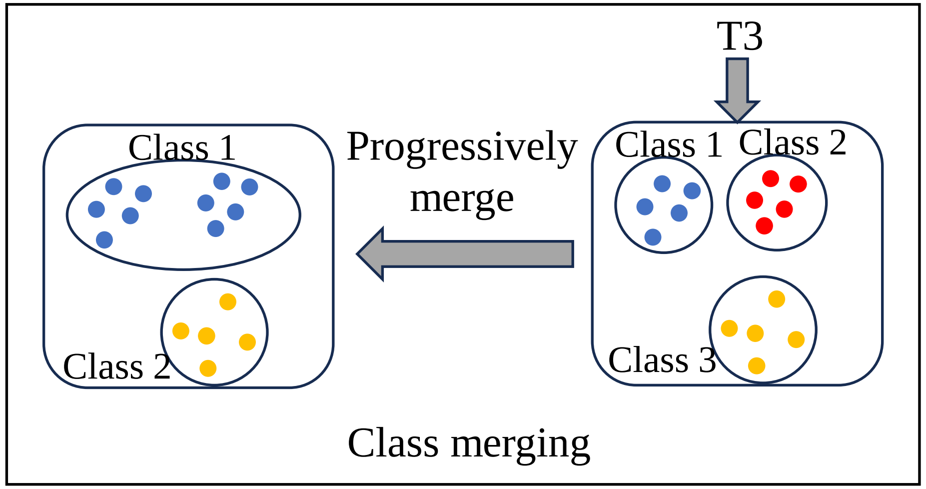


1. Assigning pseudo-labels to the unlabeled target domain data after data cleaning.
2. Utilizing these pseudo-labeled data as input to train a Sub-Center ArcFace classifier.
3. Passing all the data through the classifier and computing the selection probability for each class’s sub-center.
4. Removing the class which exhibit multiple sub-centers.

Pseudo-labeling



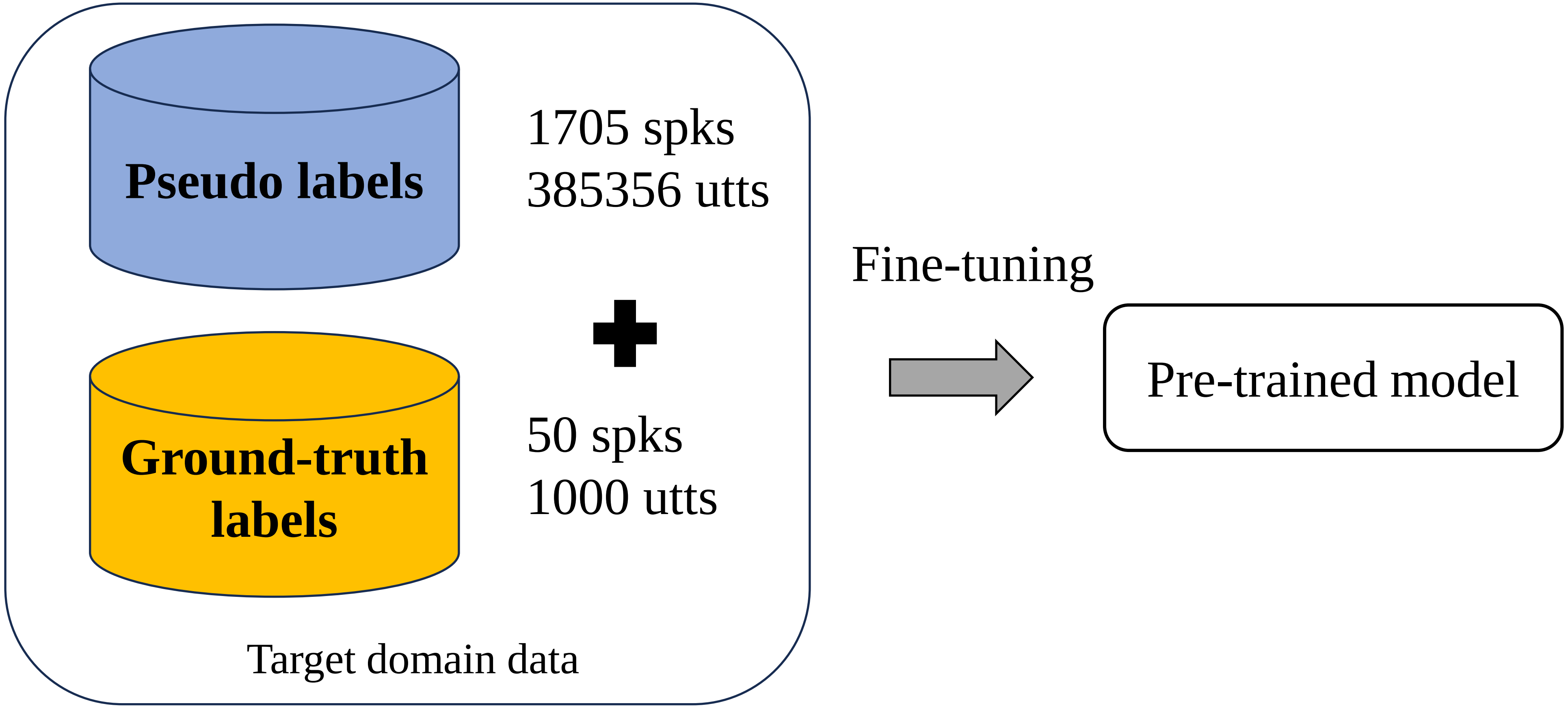
1. Progressively merge the classes based on threshold T3.



Threshold T3

1. Computing the cosine similarity between the centroid vectors of each class from labeled target domain data.
2. Selecting the maximum value as T3.

Fine-tuing



Loss function: Sub-Center ArcFace (k=3, m=0.3, s=32)

Score Calibration and Normalization

- Scoring
  - Cosine similarity
- AS-Norm
  - randomly select 20,000 utterances (duration over 4s) from unlabeled data as cohort set.
- QMF
  - 1) logarithm the enrollment utterance's duration,
  - 2) logarithm the test utterance's duration,
  - 3) magnitude of the enrollment embedding,
  - 4) magnitude of the test embedding,
  - 5) SNR of the enrollment utterance,
  - 6) SNR of the test utterance,

Results

Table 4: *The performance of various systems in the track3.*

ID & Model	VoxSRC23 val		VoxSRC23 test		VoxSRC22 test	
	EER[%]	mDCF$_{0.05}$	EER[%]	mDCF$_{0.05}$	EER[%]	mDCF$_{0.05}$
1 SimAM-ResNet100-ASP	7.490	0.342	5.287	0.3037	6.927	0.409
2 ResNet100-TSP(v2)	7.350	0.360	-	-	-	-
3 SimAM-ResNet100-ASP(v2)	7.525	0.335	-	-	-	-
4 ResNet152-ASP	7.240	0.347	-	-	-	-
5 ResNet152-Stat	7.535	0.358	-	-	-	-
Fusion(1+2)	7.115	0.324	5.095	0.2869	-	-
Fusion(1+2+3+4+5)	6.725	0.311	4.952	0.2777	6.584	0.374

Thanks~