Short-duration Speaker Verification (SdSV) Challenge 2020: the Challenge Evaluation Plan

Hossein Zeinali, Kong Aik Lee, Jahangir Alam, Lukaš Burget

Version 1.1, March 10, 2020

1   Introduction

This document describes the Short-duration Speaker Verification (SdSV) Challenge 2020. The main goal of the challenge is to evaluate new technologies for text-dependent (TD) and text-independent (TI) speaker verification (SV) in a short duration scenario.

The proposed challenge evaluates SdSV with varying degree of phonetic overlap between the enrollment and test utterances (cross-lingual). It is the first challenge with a broad focus on systematic benchmark and analysis on varying degrees of phonetic variability on short-duration speaker recognition. We expect that modern methods (deep neural networks in particular) will play a key role.

The evaluation dataset used for the challenge is drawn from the recently released multi-purpose DeepMine dataset [1, 2]. The dataset has three parts and among them, Part 1 is used for TD-SV while Part 3 is for TI-SV.

2   Tasks Description

2.1   Task 1 - Text-Dependent Speaker Verification

Task 1 of the SdSV Challenge 2020 is defined as speaker verification in text-dependent mode: given a test segment of speech and the target speaker's enrollment data, automatically determine whether a specific phrase and the test segment was spoken by the target speaker. In contrast to text-independent speaker verification, the lexical content of the utterance is also taken into consideration. As such, Task 1 is a twofold verification task in which both the speaker and phrase are verified.

In Task 1, each trial consists of a test segment along with a model identifier which indicates three enrollment utterances and a phrase ID that uttered in the utterances. The system is required to process each trial independently and produce a log-likelihood ratio (LLR) score which combines both speaker and phrase verification scores.

The enrollment and test phrases are drawn from a fixed set of ten phrases consisting of five Persian and five English phrases, respectively. The in-domain training data contains utterances from **963 speakers**, some of which have only Persian phrases. Model enrollment is done in a phrase and language-dependent way using three utterances for each model. Given

the same set of target speakers, Task 1 provides a good basis for analyzing the language factor in text-dependent speaker recognition.

2.1.1 Trial types

Given the ground truth, there are four types of trial in a TD-SV task [3]. The first is Target-Correct (TC) where the target speaker utters the correct pass-phrase. The second is Target-Wrong (TW) where the target speaker utters a wrong pass-phrase. In the same manner, the Imposter-Correct (IC) and Imposter-Wrong (IW) refer to the case where the imposter utters the correct or a wrong pass-phrase. The system should accept the TC trials as the target trial and reject the other three types as non-target (imposture) trials. Notice that the main difference between text-dependent and text-independent speaker verification is considering TW trials as imposter trial while both TC and TW are considered as target trials in the text-independent mode. There are no cross-language and cross-gender trials in Task 1 of the challenge.

2.1.2 Training condition

The training condition is defined as the amount of data/resources used to build a Speaker Recognition (SR) system. Unlike SRE19, we adopted a fixed training condition where the system should only be trained using a designated set. The fixed training set consists of the following:

- VoxCeleb1

- VoxCeleb2

- LibriSpeech

- DeepMine (Task 1 Train Partition)

The use of other public or private speech data for training is forbidden, while the use of non-speech data for data augmentation purposes is allowed. The in-domain DeepMine training data can be used for any purpose, such as neural network training, LDA or PLDA model training, and score normalization. Part of the data could also be used as a development set since there is no separate development data provided for the challenge. Note that, however, usage of Task 2 in-domain data for this task is not allowed.

2.1.3 Enrollment Condition

The enrollment is accomplished using three utterances of a specific phrase for each model. We decided to use three utterances for the model enrollment since it is commonly adopted in practice. Note that using enrollment utterances of other models is forbidden, for example, for calculating score normalization parameters (i.e., trials are to be processed independently).

2.1.4 Test Condition

Each trial in the evaluation contains a test utterance and a target model. As described above, there are four types of trials in the evaluation and only TC is considered as target and the rest will be considered as imposture. Similar to the SRE 2019 CTS challenge, the whole set of trials will be divided into two subsets: a progress subset (30%), and an evaluation subset (70%). The progress subset is used to monitor progress on the leaderboard, while the evaluation subset is used to generate the official results at the end of the challenge.

2.2 Task 2 - Text-Independent Speaker Verification

Task 2 of the SdSV Challenge is speaker verification in text-independent mode: given a test segment of speech and the target speaker enrollment data, automatically determine whether the test segment was spoken by the target speaker.

Each trial in this task contains a test segment of speech along with a model identifier which indicates one to several enrollment utterances. The net enrollment speech for each model is uniformly distributed between 3 to 120 seconds (after applying an energy-based VAD). The system is required to process each trial independently and produce a log-likelihood ratio (LLR) for each of them.

The in-domain training data in this task contains text-independent Persian utterances from **588 speakers**. This data can be used for any purpose such as LDA/PLDA, score normalization, training data for neural network, reducing the effect of language for cross-lingual trials, etc.

2.2.1 Trials

There are two partitions in this task. The first partition consists of typical text-independent trials where the enrollment and test utterances are from the same language (Persian). The second partition consists of text-independent cross-language trials where the enrollment utterances are in Persian and test utterances are in English. For this partition, the system should reduce the language effects in order to verify the test utterances in a different language. Similar to Task 1, there are no cross-gender trials in Task 2. Note that any further information about test language will not be provided but participants are allowed to train any language identification system to do it if they needed.

2.2.2 Training condition

Similar to Task 1, we adopted a fixed training condition where the system should only be trained using a designated set. The available training data is as follow:

- VoxCeleb1

- VoxCeleb2

- LibriSpeech

- DeepMine (Task 2 Train Partition)

The use of other public or private speech data for training is forbidden, while the use of non-speech data for data augmentation purposes is allowed. The in-domain DeepMine training data can be used for any purpose, such as neural network training, LDA or PLDA model training, and score normalization. Part of the data could also be used as a development set since there is no separate development data provided for the challenge. Note that, however, usage of Task 1 in-domain data for this task is not allowed.

2.2.3 Enrollment Condition

The enrollment data in Task 2 consists of one to several variable-length utterances. The net speech duration for each model is roughly 3 to 120 seconds. Since each enrollment utterance is a complete recording without trimming to a specific duration, the overall duration might not be exactly uniform. Note that using the enrollment utterances from the other models is forbidden, for example, for calculating score normalization parameters.

2.2.4 Test Condition

Each trial in the evaluation contains a test utterance and a target model. The duration of the test utterances varies between 1 to 8 seconds. Similar to Task 1, the whole set of trials is divided into two subsets: a progress subset (30%), and an evaluation subset (70%). The progress subset is used to monitor progress on the leaderboard. The evaluation subset is used to generate the official results at the end of the challenge.

2.3 Performance Measurement for Both Tasks

The main metric for the challenge is normalized minimum Detection Cost Function (DCF) as defined is SRE08. This detection cost function is defined as a weighted sum of miss and false alarm error probabilities:

$$C_{Det} = C_{Miss} \times P_{Miss\,|\,Target} \times P_{Target} + C_{FalseAlarm} \times P_{FalseAlarm\,|\,NonTarget} \times (1 - P_{Target})\,,$$

where $C_{Miss} = 10$, $C_{FalseAlarm} = 1$ and $P_{Target} = 0.01$. Based on the parameters, the normalized DCF (DCF_{norm}) will be DCF divide by 0.1 as the best cost that could be obtained without processing the input data. In addition to $minDCF_{norm}^{0.01}$, the Equal Error Rate (EER) will be reported.

2.4 Data Description

The main data for the challenge is the DeepMine dataset which was collected using crowd-sourcing [4]. Participants in the data collection project installed an Android application and record phrases in the application. The full description of the project and the dataset can be found in [2, 1]. Bibtex sources for citing the database are provided below easy reference:

```
@inproceedings{deepmine2018odyssey,
title={{DeepMine} Speech Processing Database: Text-Dependent and
    Independent Speaker Verification and Speech Recognition in
    {Persian and English}.},
```

Table 1: Phrases in Task1 of the challenge.

Id	Phrase
01	sedaye man neshandahandeye hoviyyate man ast.
02	sedaye har kas monhaser be fard ast.
03	hoviyyate man ra ba sedaye man tayid kon.
04	sedaye man ramze obure man ast.
05	baniadam azaye yekdigarand.
06	My voice is my password.
07	OK Google.
08	Artificial intelligence is for real.
09	Actions speak louder than words.
10	There is no such thing as a free lunch.

```
    author={Zeinali, Hossein and Sameti, Hossein and Stafylakis, Themos},
    year=2018,
    booktitle={Proc. Odyssey 2018 The Speaker and Language Recognition
    Workshop},
    pages={386--392},
}


@inproceedings{deepmine2019asru,
    title={A Multi Purpose and Large Scale Speech Corpus in {Persian and
    English} for Speaker and Speech Recognition: the {DeepMine} Database},
    author={Zeinali, Hossein and Burget, Lukas and Cernocky, Jan},
    year=2019,
    booktitle={Proc. ASRU 2019 The 2019 IEEE Automatic Speech Recognition
    and Understanding Workshop},
}
```

The database was recorded in realistic environments, which took place in Iran, and the collection protocol was designed to incorporate various kinds of noises in the collection. The main language is Farsi (Persian) while most of the participants also participated in the English partition. Part 1 of the dataset contains five Persian phrases as well as five English phrases that are used in Task 1 of the challenge. The English phrases and transliteration of the Persian phrases are shown in Table 1. The phoneme transcription of the phrases will be provided with the data and participants can use them in any way they want. Part 3 of the dataset contains text-independent phrases in Persian and is using in Task 2 of the challenge as the enrollment data.

2.5   Data Organization

Data will be provided in three separate zip (tar) files for each task. The first zip file contains the in-domain DeepMine training data. The second zip file contains enrollment data, model definition file, and trial file. The last zip file contains test utterances. If all three files will be extracted in a directory, the directory structure is as follow:

```
<base directory>/
  README.txt
  docs/
    model_enrollment.txt
    train_labels.txt
    trials.txt
  wav/
    enrollment/
      enr_000000.wav
      enr_000001.wav
      ...
    evaluation/
      evl_000000.wav
      evl_000001.wav
      ...
    train/
      trn_000000.wav
      trn_000001.wav
      ...
```

Note that the zip files for Task 1 and Task 2 should be extracted into separate directories since their contents are partially overlapping.

2.6   Format of Model Enrollment File

2.6.1   Task 1 Enrollment File

The enrollment file of Task 1 is a space-separated five-column text file `model_enrollment.txt` located in the `docs` directory. There is a header line at the beginning of the file. The first record in each line indicates a model-ID, the second record shows the phrase ID which indicates the phrase uttered in the corresponding utterances. The remaining three columns show the enrollment file IDs. There is only one space between two records in each line. The format of the enrollment file is as follow:

```
model-id<SPACE>phrase-id<SPACE>enroll-file-id1<SPACE>enroll-file-id2
<SPACE>enroll-file-id3<NEWLINE>
```

where `model-id` is the model identifier, `phrase-id` is the phrase identifier and `enroll-file-ids` are the enrollment utterance identifiers.

For example:

```
model-id phrase-id enroll-file-id1 enroll-file-id2 enroll-file-id3
model_00000 07 enr_007492 enr_023277 enr_012882
model_00001 02 enr_035341 enr_027674 enr_032835
model_00002 09 enr_020095 enr_015193 enr_024742
model_00003 06 enr_032246 enr_014610 enr_014698
model_00004 09 enr_033841 enr_037127 enr_033859
```

2.6.2   Task 2 Enrollment File

The enrollment file of Task 2 is a space-separated text file `model_enrollment.txt` located in the `docs` directory. There is a header line at the beginning of the file. There are at least two records in each line. The first record indicates a model ID and the second record shows the first enrollment file. If the model has more than one enrollment files, the rest will follow in the same line. The number of columns in each line may differ depending on the number of enrollment files. There is only one space between two records in each line. The format of the enrollment file is as follow:

```
model-id<SPACE>enroll-file-id1[<SPACE>enroll-file-id2
[<SPACE>enroll-file-id3...]]<NEWLINE>
```

where `model-id` is the model identifier and `enroll-file-ids` are the enrollment utterance identifiers.
     For example:

```
model-id enroll-file-ids ...
model_15002 enr_110254 enr_264593
model_15005 enr_188426 enr_303480 enr_200614 enr_117624
model_15006 enr_072239
model_15007 enr_248083 enr_316783 enr_088834
model_15008 enr_177720 enr_334136 enr_226306 enr_057733 enr_190105
model_15009 enr_059968 enr_234582
model_15011 enr_310490 enr_264156 enr_055518 enr_091529
```

2.7   Format of Trial File

The trial file for both tasks is a space-separated two-column text file `trials.txt` located in the `docs` directory. There is a header line at the beginning of the file. The first record in each line indicates a model ID and the second record indicates an evaluation file ID. There is only one space between two records in each line. The format of the trial file is as follow:

```
model-id<SPACE>evaluation-file-id<NEWLINE>
```

where model-id is the model identifier and evaluation-file-id is the test utterance identifier.
     For example:

```
model-id evaluation-file-id
model_00000 evl_000018
model_00000 evl_000021
model_00000 evl_000035
model_00000 evl_000109
model_00000 evl_000117
model_00000 evl_000165
```

3 In-domain Training Set

3.1 Task 1 In-domain Data

As described earlier, the in-domain data for Task 1 consists of utterances from 963 speakers. All training utterances are stored in the `wav/train` directory. The `train_labels.txt` file in the `docs` directory is a space-separated text file that contains the information for each utterance. Each line in this file contains three columns, where the first column shows `train-file-id`, the second one indicates the `speaker-id` and the last one shows `phrase-id`. There is a header line at the beginning of the file. The format of train label file is as follow:

`train-file-id<SPACE>speaker-id<SPACE>phrase-id<NEWLINE>`

where `rain-file-id` is the train utterance identifier, `speaker-id` is the speaker label and finally, the `phrase-id` is the identifier of phrase of each utterance.

For example:

```
trn_000001 spk_000001 09
trn_000002 spk_000001 09
trn_000003 spk_000001 09
trn_000004 spk_000001 09
trn_000005 spk_000001 09
trn_000006 spk_000001 09
trn_000007 spk_000001 09
```

3.2 Task 2 In-domain Data

The in-domain data for Task 2 consists of utterances from 588 speakers. All training utterances are stored in the `wav/train` directory. The `train_labels.txt` file in docs directory is a space-separated text file that contains the information for each utterance. Each line in this file contains two columns, where the first column shows `train-file-id` and the second one indicates the `speaker-id`. There is a header line at the beginning of the file. The format of train label file is as follow:

`train-file-id<SPACE>speaker-id<NEWLINE>`

where `train-file-id` is the train utterance identifier and `speaker-id` is the speaker label.

For example:

```
trn_101064 spk_001000
trn_101065 spk_001000
trn_101066 spk_001000
trn_101067 spk_001000
trn_101068 spk_001000
trn_101069 spk_001000
trn_101070 spk_001000
```

4 Evaluation Rules and Requirements

The overall rules are pretty the same as NIST SREs. First of all, participants must follow the data restriction where there is only a fixed training condition. Participants are also required to process the test data according to the following rules and upload results to the challenge website for evaluation. These rules are:

- Participants agree to make at least one valid submission for one of the tasks.

- Participants agree to process each trial independently. That is, each decision for a trial is to be based only upon the specified test segment and target speaker enrollment data. The use of information about other test segments and/or other target speaker data is not allowed.

- Participants agree not to probe the enrollment or test segments via manual/human means such as listening to the data or producing the manual transcript of the speech.

- Participants are allowed to use any automatically derived information for training, development, enrollment, or test segments.

- Participants may make multiple challenge submissions (only one per day). Based on the leaderboard results participant should select up to two systems and submit them with system description to the provided links.

5 Baselines

There is a common x-vector [5] baseline for both text-dependent and independent tasks. Also, we will prepare the second baseline for Task 1 (text-dependent) during the challenge period. We provide below a brief description of the baselines. The source code for the x-vector baseline can be found on the challenge website.

5.1 X-Vector Baseline

The first baseline is a well-known x-vector system for text-independent speaker verification [5]. Here E-TDNN topology [6] is used and trained using VoxCeleb1 and VoxCeleb2. After training the network, an LDA with 150 dimensions is applied to the extracted x-vectors and after that, a PLDA with both VoxCeleb datasets is trained. Finally, trials are scored using the PLDA without any score normalization.

5.2 I-Vector/HMM Baseline

Because the x-vector baseline does not consider any phrase information and is not optimized for the text-dependent task, the i-vector/HMM method is selected as the second baseline for Task1. The method was proposed in [7, 8] and have achieved very good results on both RSR2015 [3] and RedDots [4] datasets.

In this method, i-vector is used as a fixed dimension representation for each utterance. In contrast to the conventional i-vector/GMM method which uses GMM for aligning the frames to Gaussian components, here monophone HMMs are used as frame alignment strategy. So, the first step is training monophone HMMs using the training data. These models then will be used to extract sufficient statistics and by using them an i-vector extractor is trained. Scoring is done using LDA-Cosine and scores are normalized using the s-norm method. The full description of this method can be found in [8].

6 Prizes

There will be three cash prizes for each task. The winners will be selected based on the results of the primary systems on the evaluation subset. In addition to the cash prize, each winner will receive a certificate for their achievement. The cash prizes are as follow:

- Rank 1: **500** EUR

- Rank 2: **300** EUR

- Rank 3: **100** EUR

7 Evaluation Protocol

7.1 Challenge Website

The challenge has a Github page (`https://sdsvc.github.io/`). All information and news about the challenge will be posted on the website. System description submitted by the participants will be made available on the website as well. Participants are allowed to make the description anonymous (in a blind format without any personal and affiliation information).

7.2 Leaderboard platform

As described above, there is an online leaderboard for each task and participants can submit one system per day during the period of evaluation. The leaderboard shows the performance of the systems on the progress subset. At the end of the challenge, participants should submit the outputs of their selected systems via the provided links. The challenge leaderboard platforms are available at

- Task 1: `https://competitions.codalab.org/competitions/22393`

- Task 2: `https://competitions.codalab.org/competitions/22472`

7.3 Required submissions

In addition to submitting score files to the online leaderboard, participants should submit the selected systems and corresponding system description using a separate link which will be provided later. The selected systems are as follow:

- **Primary system**: this system is the primary system for participants and is mandatory to submit this system. This system will be used for ranking the participants.

- **Single system**: this submission shows the single system. If the primary system was a fusion of several systems, participants should submit a single system as well. If the primary system is created based on a single system, this submission is not necessary. The single system definition for the challenge will be provided in the following paragraph.

The final submission file is a zip (tar) file for each task and should contain the following:

- primary.sco

- single.sco [optional in the case of using same system in the primary]

7.3.1 Single system definition

There is no single definition for what constitutes a single-system submission. For this challenge, we adopted the following definition.

Let's divide a verification system to front-end and back-end modules. Front-end means any molding mechanism like i-vector or x-vector for extracting embedding from utterances, while the back-end module is used for scoring, for example, LDA-Cosine or LDA-PLDA. Both subparts should be trained and used sequentially. In other words, the output of a system should be used in the consequent subparts. So we have the following rules:

- Feature fusion by concatenating is allowed.

- The network can be in any format and size but it should be trained only in one training pass (for example training two separate models is not allowed while a big network from combination of the two networks is fine) except for phrase or language-specific modeling which separate models can be trained for each phrase or each language. Note that in a phrase or language-dependent modeling only one of the networks will be used for scoring each trial.

- The network can be used for scoring in an end-to-end fashion or for extracting embeddings.

- Any phrase or language-dependent scoring can be used but for each trial, only one of them should be used. For example, in phrase-dependent mode, only the corresponding model to phrase should be used.

- The only allowed score fusion for the single system is a fusion between DNN posteriors and backend outputs.

7.4 System Output Format

The system output should be a one-column text file. Each line of the file indicates a LLR score (a float number) of the corresponding trial. The order of scores must be the same as the trials file and all of the trials must be scored. Any inconsistency will cause an error in the evaluation of the system.

For example:

```
-6.1284
-97.8528
-16.8025
-44.3276
4.4121
-61.0123
-42.9890
```

7.5 Data License Agreement

The evaluation data for this challenge is a subpart of the DeepMine dataset. To use this dataset, participants should sign a data license agreement specific to SdSV 2020 Challenge. This license allows participants to use the data for the challenge as well as the subsequent paper publication. Any other usage of the data is not allowed. The license agreement file can be found on the challenge website.

7.6 System Description

Each participant is required to submit a full system description. The system description will be made available online. Participants can use their team ID as the team name and write the description in an anonymous format. In addition to the system description, we strongly recommend participants to submit papers to the special session related to the challenge in InterSpeech2020. The papers will be reviewed as a normal paper, so they should be in a proper format and has sufficient novelty for acceptance.

The system description should have at least 2 pages and must include the following information about the submitted systems:

- a complete description of the system components, including front-end and back-end modules along with their configurations.

- a complete description of the data partitions used to train the various models.

- performance of the submitted systems on the progress and evaluation subsets reported on the leaderboard website.

Bibtex source for citing the evaluation plan is provided below for easy reference.

```
@techreport{sdsvc2020plan,
    title={Short-duration Speaker Verification (SdSV) Challenge 2020:
    the Challenge Evaluation Plan.},
```

```
    author={Zeinali, Hossein nad Lee, Kong Aik and Alam, Jahangir and
    Burget, Luka\v{s}},
    institution={arXiv preprint arXiv:1912.06311},
    year=2020,
}
```

8   Planned Evaluation Schedule

Release of evaluation plan:	January 10, 2020
Evaluation platform open:	January 15, 2020
Release of Train, Dev, and Eval sets:	January 10, 2020
Challenge deadline:	April 17, 2020
Release of results:	April 27, 2020
Post-challenge evaluation:	October 25, 2020
INTERSPEECH Paper submission:	May 8, 2020
SdSV Challenge 2020 special session:	October 25-29, 2020

References

[1] H. Zeinali, H. Sameti, T. Stafylakis, DeepMine speech processing database: Text-dependent and independent speaker verification and speech recognition in Persian and English., in: Proc. Odyssey 2018 The Speaker and Language Recognition Workshop, 2018, pp. 386–392.

[2] H. Zeinali, L. Burget, J. Cernocky, A multi purpose and large scale speech corpus in Persian and English for speaker and speech recognition: the DeepMine database, in: Proc. ASRU 2019 The 2019 IEEE Automatic Speech Recognition and Understanding Workshop, 2019.

[3] A. Larcher, K. A. Lee, B. Ma, H. Li, Text-dependent speaker verification: Classifiers, databases and rsr2015, Speech Communication 60 (2014) 56–77.

[4] K. A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brümmer, D. v. Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma, et al., The reddots data collection for speaker recognition, in: Sixteenth Annual Conference of the International Speech Communication Association, 2015.

[5] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, S. Khudanpur, X-vectors: Robust dnn embeddings for speaker recognition, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2018, pp. 5329–5333.

[6] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, S. Khudanpur, Speaker recognition for multi-speaker conversations using x-vectors, in: ICASSP 2019-2019 IEEE

International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 5796–5800.

[7] H. Zeinali, L. Burget, H. Sameti, O. Glembek, O. Plchot, Deep neural networks and hidden Markov models in i-vector-based text-dependent speaker verification., 2016.

[8] H. Zeinali, H. Sameti, L. Burget, HMM-based phrase-independent i-vector extractor for text-dependent speaker verification, IEEE/ACM Transactions on Audio, Speech, and Language Processing 25 (7) (2017) 1421–1435.