**The VoiceMOS Challenge 2022  
System Description and Participant Questionnaire**

(Please edit the space between questions to fill in your answers)

**About your team:**

1. Team name:

DKU

1. Affiliation and country / academia or industry

Duke Kunshan University, China (academia)

1. How many people were involved in your team?

3

1. Does your team have background in text-to-speech synthesis / voice conversion?

No

1. Have you participated in similar challenges in the past? (e.g., other challenges related to speech quality assessment)

No

**About your system and components:**

1. If any, please specify the name of your system   
   SLDNet
2. How long your system and/or the related research has been under development/study (e.g., 10 years)?

Less than 1 year

1. Is your system available commercially or as open source? If so, please specify it and/or provide a link with related online information.

https://github.com/tjysdsg/VoiceMOS

1. If any, please specify components from common system(s) or toolkit(s) used in your system.

The LDNet baseline

**About the use of the data**

1. Did you use data other than those provided for the challenge (i.e. the training sets from the main track and the OOD track) to train your systems? If so, please specify the nature of the data (database) and how it was used in your system.

No

1. Please specify any pre-processing that was applied to the audio before training models. What kind of features does your model use as input?

Speaker embeddings were extracted by a pretrained speaker verification model provided by the original authors of *The DKU-DukeECE Systems for*

*VoxCeleb Speaker Recognition Challenge 2020*. This model is the one using ResNet34 without BAM, as described in Section 1.

The paper can be found at https://arxiv.org/pdf/2010.12731.pdf

1. During training, did you use averaged scores per utterance during training, or individual per-listener scores?

Yes. Averaged scores are used to train mean listener interface, the same as LDNet.

1. Did you use the listener info and demographics in any way during training? If so, how?

Listener IDs are used, the same as LDNet.

1. If you participated in the OOD track, please specify how you used the OOD data, both the labeled set and the unlabeled set.

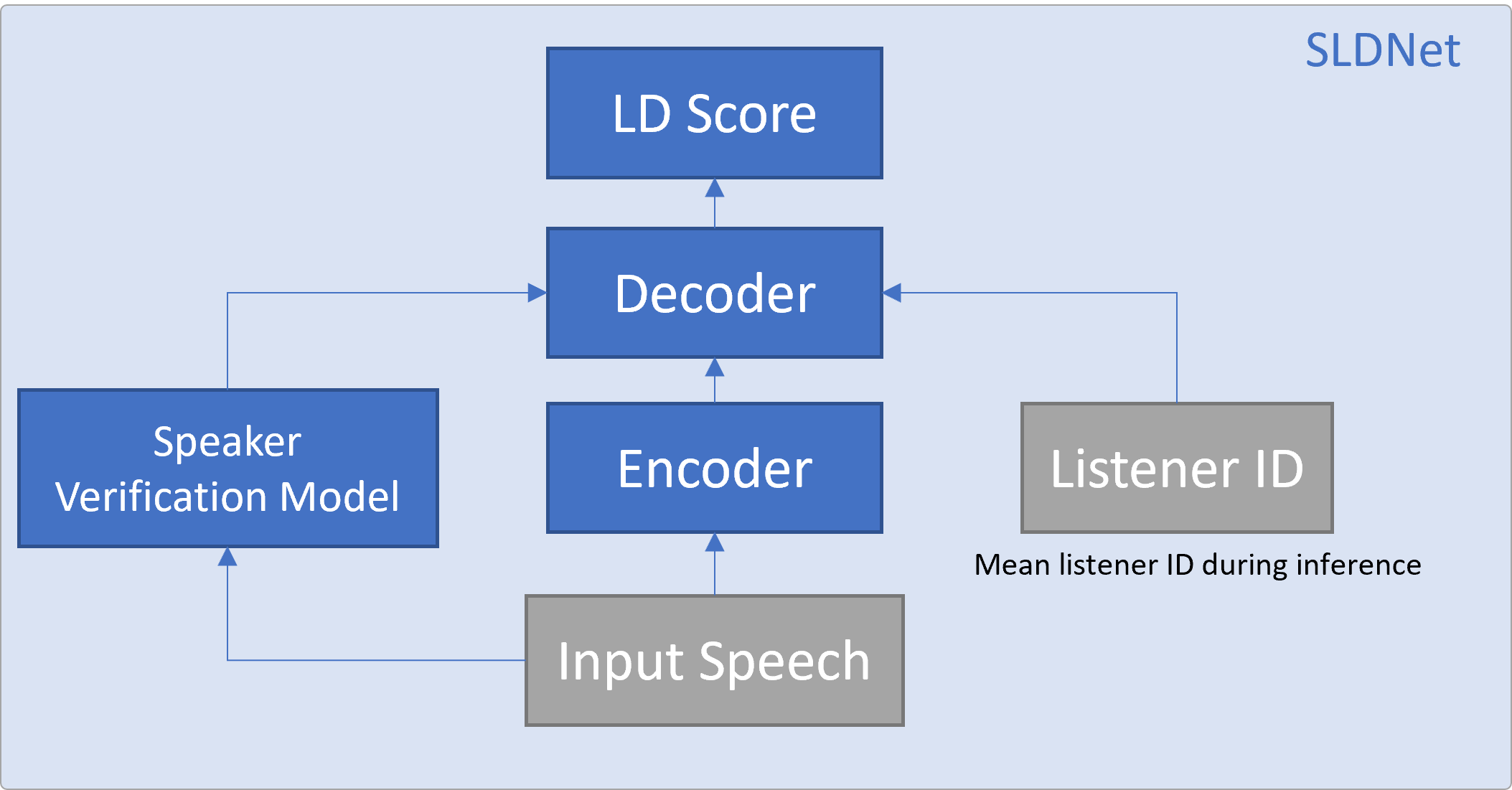
We did not participate in the OOD track.

**About MOS prediction model architecture and training process**

1. Please describe in detail the system architecture of your MOS prediction model. Please also include details about training losses, model selection criteria, learning rate, etc. If your OOD model architecture differs from your main-track one, please also specify those differences.  
   Compared to LDNet, The SLDNet model has one additional input: speaker embeddings with a size of 128 are concatenated to the decoder input. The vectors are repeated along the time dimension (and judge dimension if using the mean listener mode) before concatenating. We used the same optimizer and learning rate scheduler configurations as LDNet. The file at dku/run.sh contains specific configuration used and the exact command to train the model.
2. Please clarify any differences between the techniques/models used for the training and those used during inference (e.g. use of listener info, etc.). (For example, using individual listener scores during training and ‘all listeners’ / ‘mean listener’ mode during inference, as in the LDNet baseline.)

During inference, we used mean listener mode as did in LDNet.

1. Please provide a diagram of your system. Standard network architectures and toolkits can be represented as a box for simplicity and just specify the part you modified. If the model you used for the OOD track is substantially different from the main one, then please include two diagrams.



**About the computational cost**

1. Please provide the number of trainable parameters of your system.

Speaker verification model: 6.1 million parameters

SLDNet: 1.6 million parameters

1. How many CPU and GPU hours did it take to build / train your model?  
   Main track: ~30 GPU hours   
   OOD track: ~NA  
   Wav2vec2 was pretrained and its training time is not included.
2. How many GPUs did you use to build / train your model?

5 GPU

1. What is the memory footprint of the system at runtime?

Around 650 MB of VRAM during inference

**Paper writing**

1. Are you planning to submit a paper to Interspeech about your system?  
 No

2. If not, what are the reasons that you will not submit a paper? (E.g., novelty is not enough, results are not good, company policy, etc.)  
Novelty is not enough

3. If you will not submit an Interspeech paper, will you submit a preprint that summarizes your system to arxiv.org?

No