Google

Closing the gap between single-user and multi-user VoiceFilter-Lite

Rajeev V. Rikhye, Quan Wang, Qiao Liang, Yanzhang He, Ian McGraw

ODYSSEY
June 28th-July 1st 2022, Beijing, China

Abstract

Problem:

- Most *speaker conditioned speech models (eg.* VoiceFilter-Lite) only allows a single enrolled speaker
- Our previous multi-user VoiceFilter-Lite model suffered from worse performance compared to the best single-user version when a single speaker is enrolled

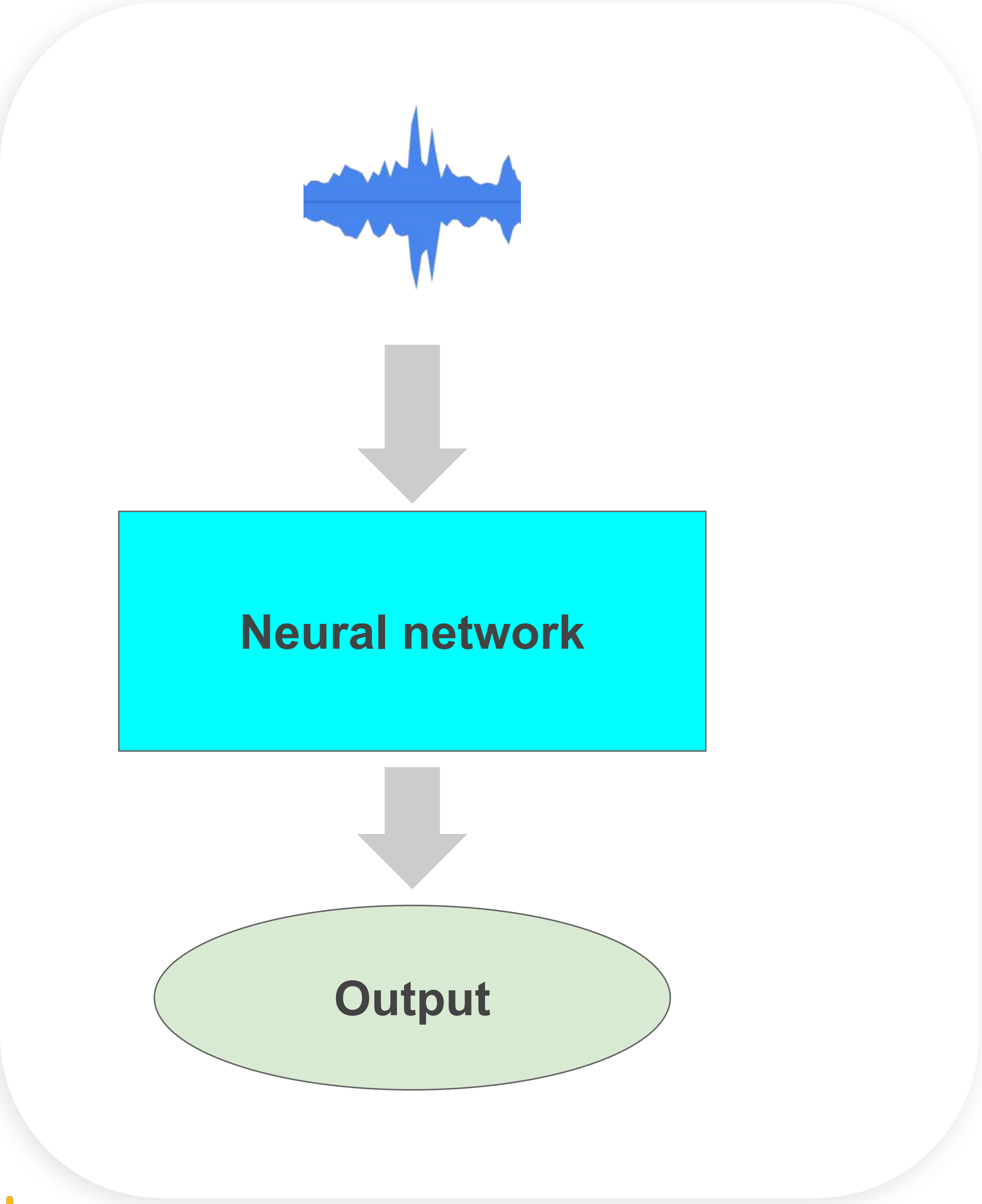
Our solution:

- A novel **attention mechanism** to identify which of the *N* enrolled users is speaking in a particular frame
- We used **feature-wise linear modulation** (FiLM) to condition the *VoiceFilterNet* with the attended speaker embedding
- We developed a dual learning rate schedule to train the *AttentionNet* at a lower rate than the *VoiceFilterNet*

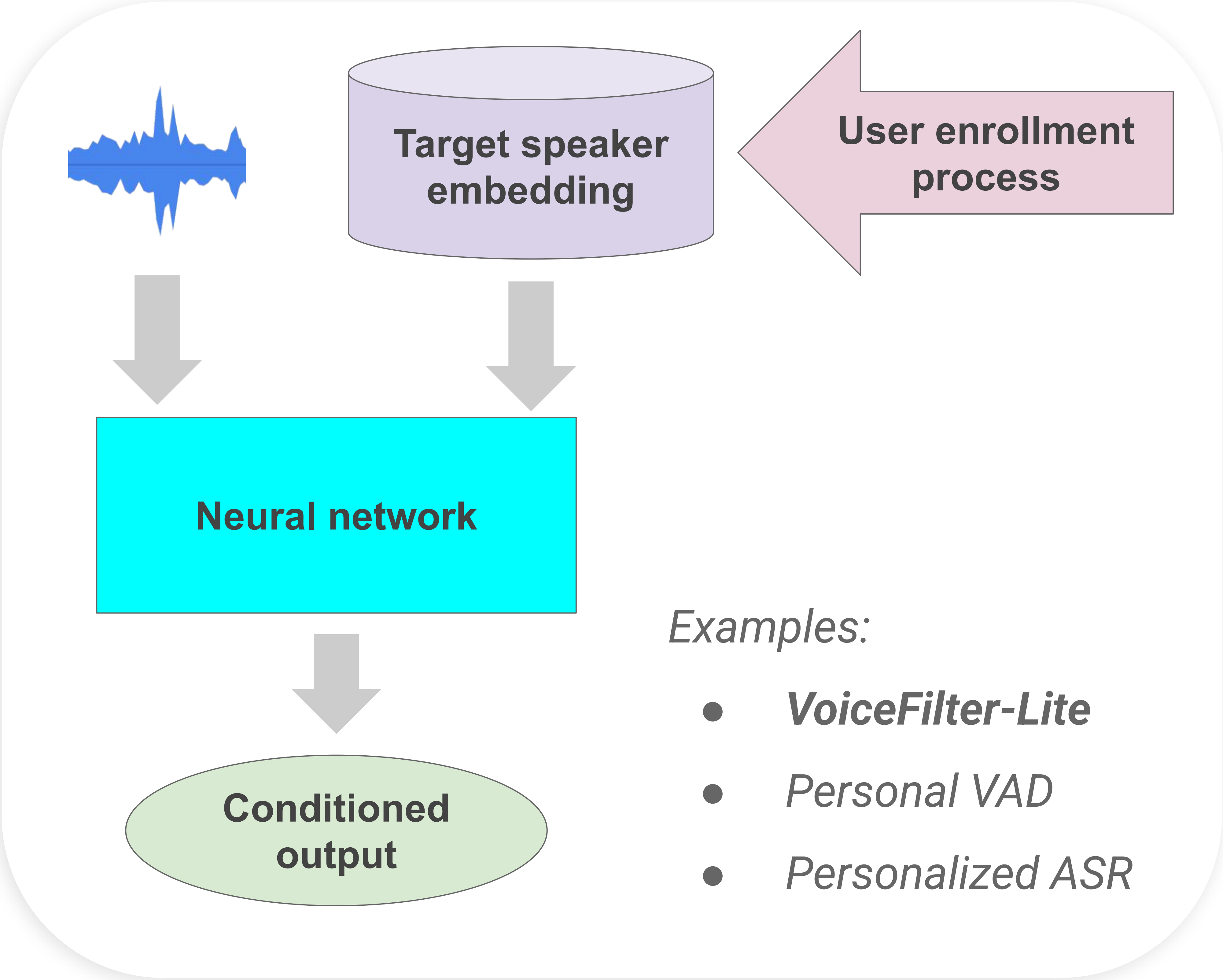
Outcome:

- We significantly improved the performance of the multi-user VoiceFilter-Lite model
    - Single enrolled user case: EER is on-par with the best single-user VoiceFilter-Lite model
    - Two enrolled users case: slight degradation compared with single-user; still significant improvement compared with no-VoiceFilter-Lite

Speaker conditioned speech models



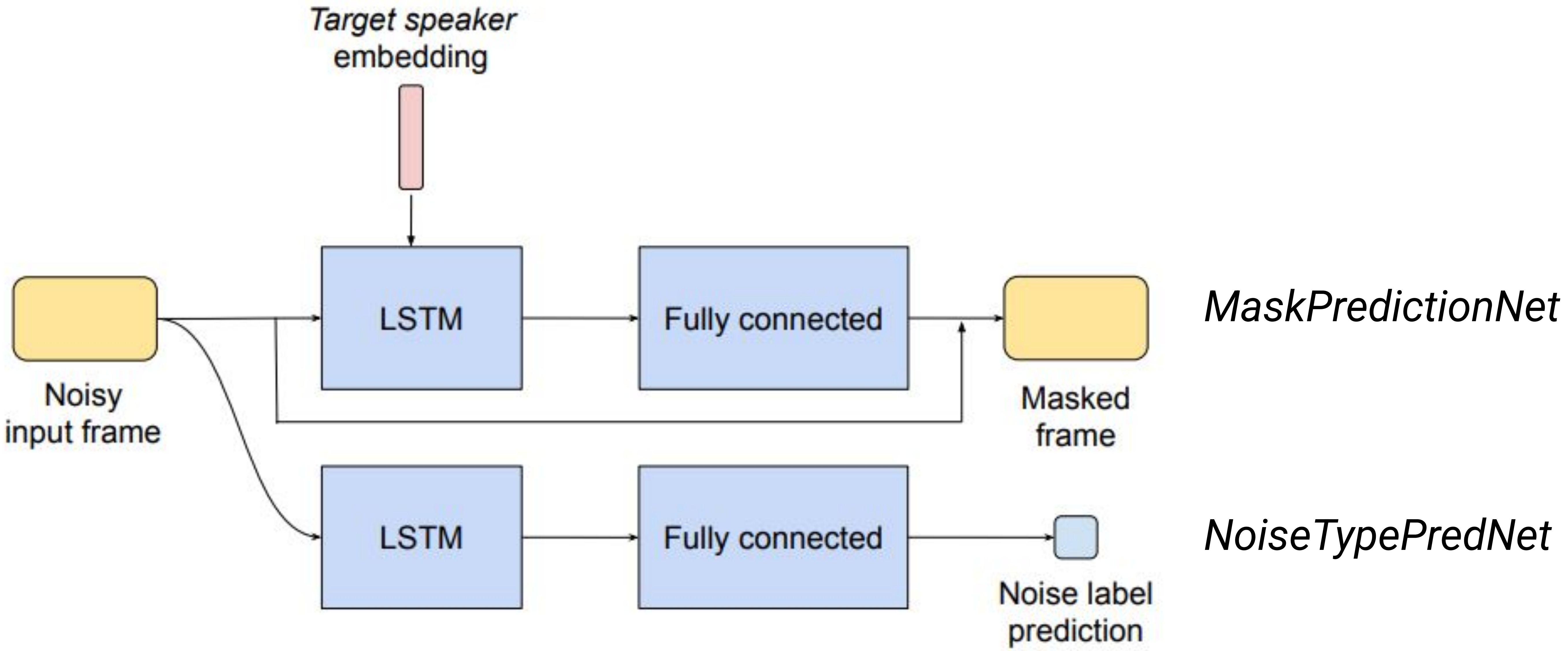
Generic speech model



Speaker conditioned speech model

VoiceFilter-Lite enhances **target user** speech in multitalker environments

Model size: 2.62 MB



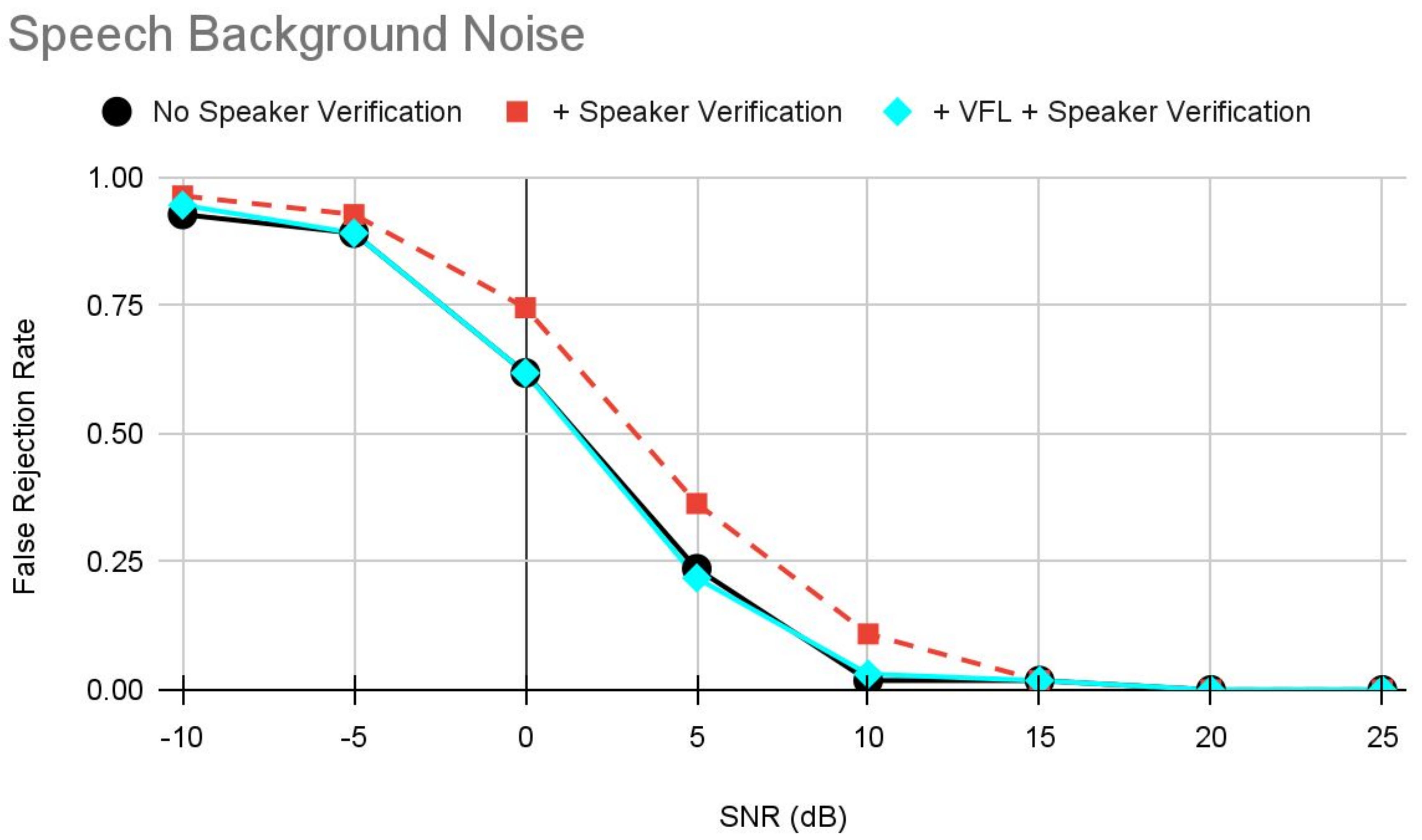
- The VoiceFilter-Lite (SUVF) [1] takes as input the target speaker embedding and a stacked log Mel filterbank energies (LFBE) and returns an "enhanced" LFBE and a noise label prediction.
- SUVF **suppresses** overlapping speech from non-enrolled users.
- Noise label is used to disable the SUVF when the frame does not contain overlapping speech.

[1] Q. Wang, et al., "VoiceFilter-Lite: Streaming targeted voice separation for on-device speech recognition," in Proc. Interspeech, 2020, pp. 2677–2681

VoiceFilter-Lite improves speaker verification robustness to overlapping background speech

Vendor-collected dataset (303 speakers, 92K queries, 97 hours)

Noise source	Room	SNR (dB)	EER (%) No VFL	EER (%) With VFL
Speech	Additive	-5	12.83	**4.24**
		0	8.34	**2.35**
		5	4.99	**1.47**
	Reverb	-5	17.76	**7.03**
		0	11.04	**3.63**
		5	6.41	**2.09**



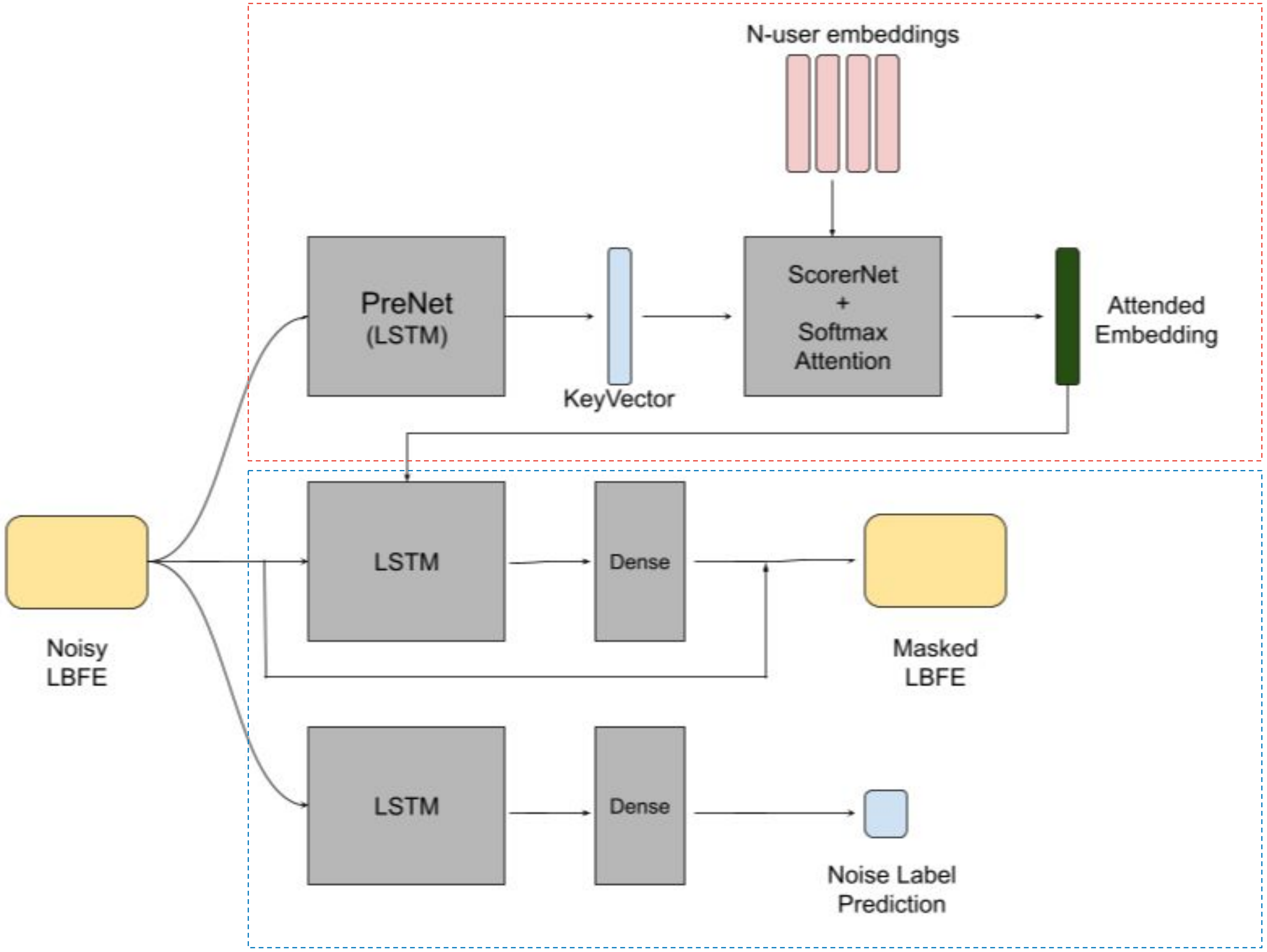
- Identifying the target speaker when there is overlapping speech is a known challenge. [2]
- Adding VoiceFilter-Lite in the speaker verification frontend helps to improve target speaker verification, reducing the number false rejects in the keyphrase detection.
- However, the VoiceFilter-Lite model only supports a single-enrolled user, which is undesirable since most smart speakers have multiple users.

[2] R. Rikhye, Q. Wang, et al., "Personalized Keyphrase Detection using Speaker and Environment Information" in Proc. Interspeech, 2021

Extending VoiceFilter-Lite to support an arbitrary number of enrolled users

Multi-user VoiceFilter-Lite (MUVF) model Architecture

Model size: 3.3 MB



AttentionNet

VoiceFilterNet

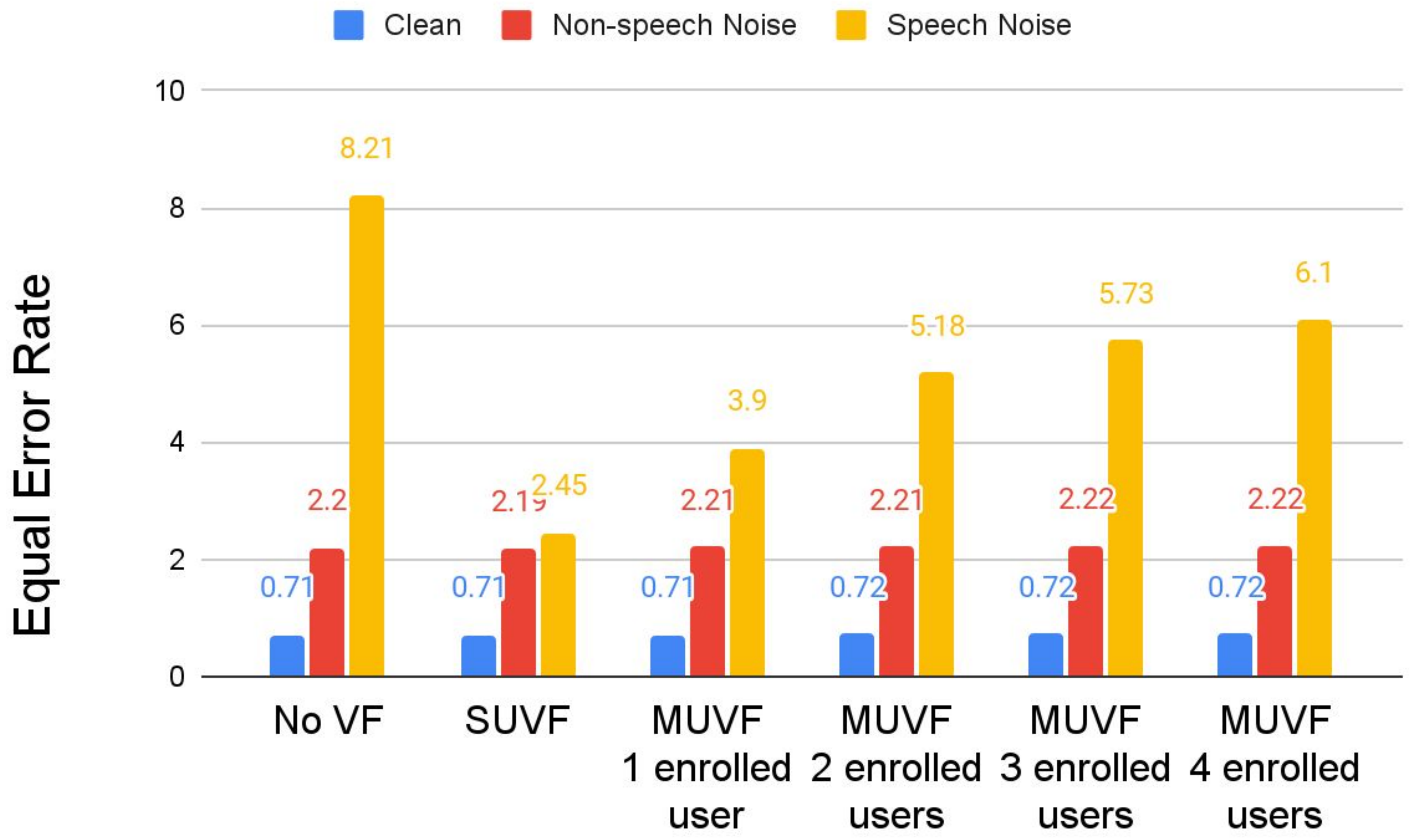
(identical to single-user VoiceFilter-Lite)

MUVF uses attention to compute the ***most likely* target speaker embedding** from the input conditioned on a set of known speaker profiles

[3] Rikhye et al., "Multi-user VoiceFilter-Lite via Attentive Speaker Embedding," in ASRU 2021

Multi-user VoiceFilter-Lite (MUVF) has poor single user performance

Speaker Verification task under various noise conditions.



Vendor-collected dataset (958 speakers, 220K utterances)

Note: Only SNR 0dB, additive noise condition is shown

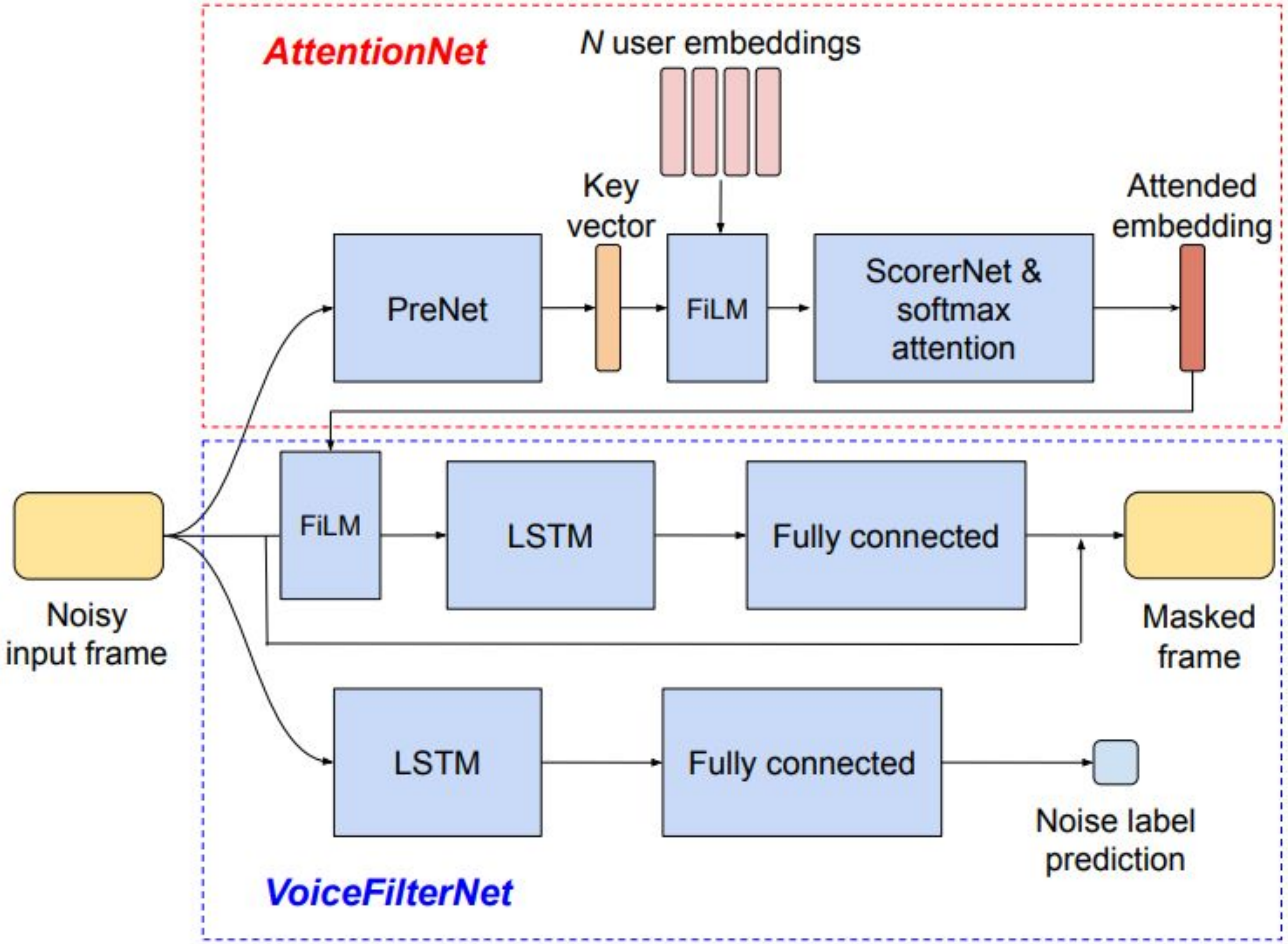
Our previously published results showed that although having an MUVF reduces the overall equal error rate (EER), performance with just 1 enrolled user is **significantly worse** than the current SUVF.

[3] Rikhye et al., "Multi-user VoiceFilter-Lite via Attentive Speaker Embedding," in ASRU 2021

Improving the multi-user VoiceFilter-Lite model to match single-user performance

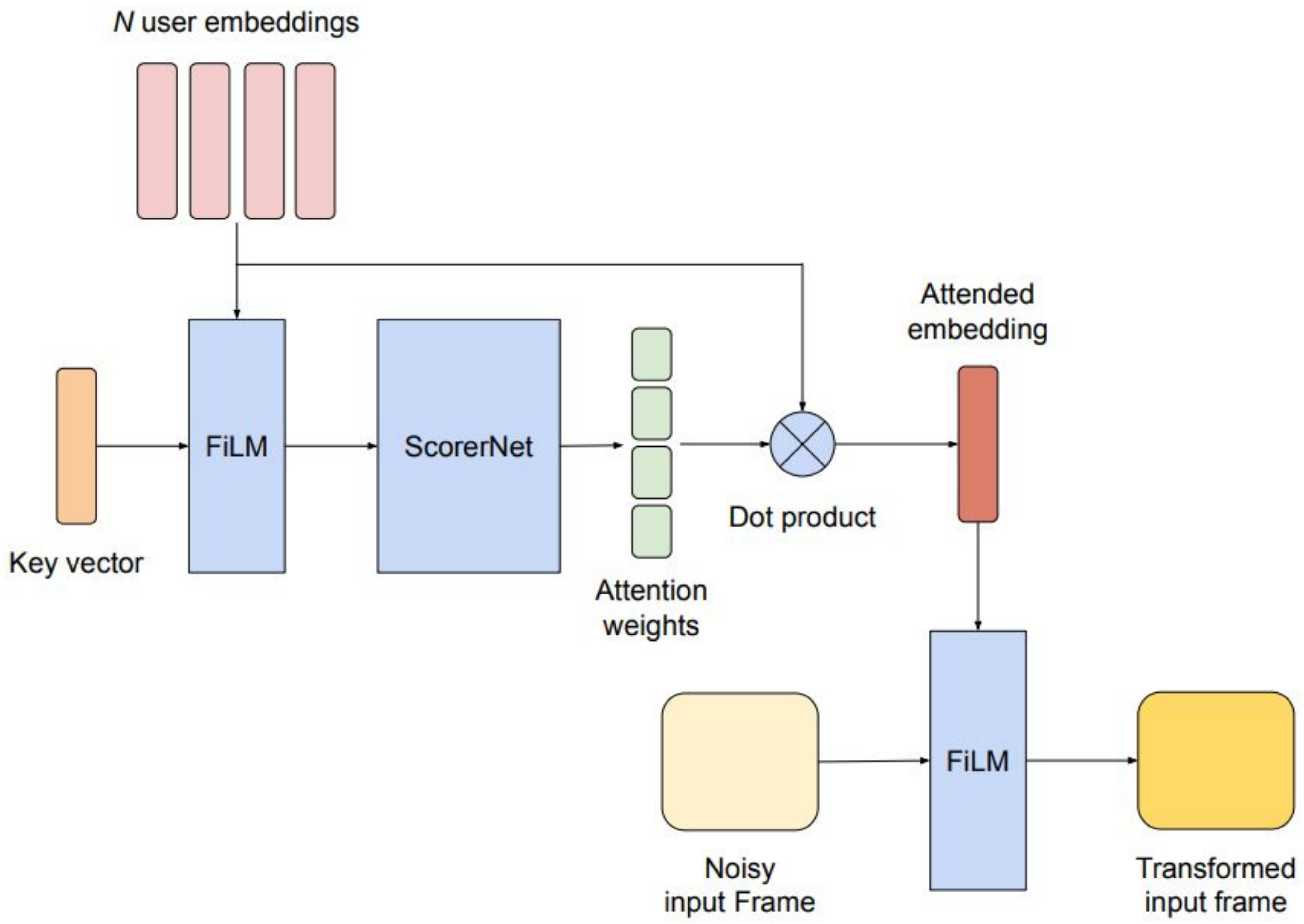
Updated multi-user VoiceFilter-Lite (MUVF) model Architecture

Model size: 3.47 MB



- **AttentionNet** to compute the most likely speaker (attended embedding) from a given frame.
- **Feature-wise Linear modulation (FiLM)** to condition the input to the VoiceFilterNet with the attended embedding.
- **Dual learning rate scheduler**, which trains the AttentionNet with a slower learning rate.

AttentionNet Architecture



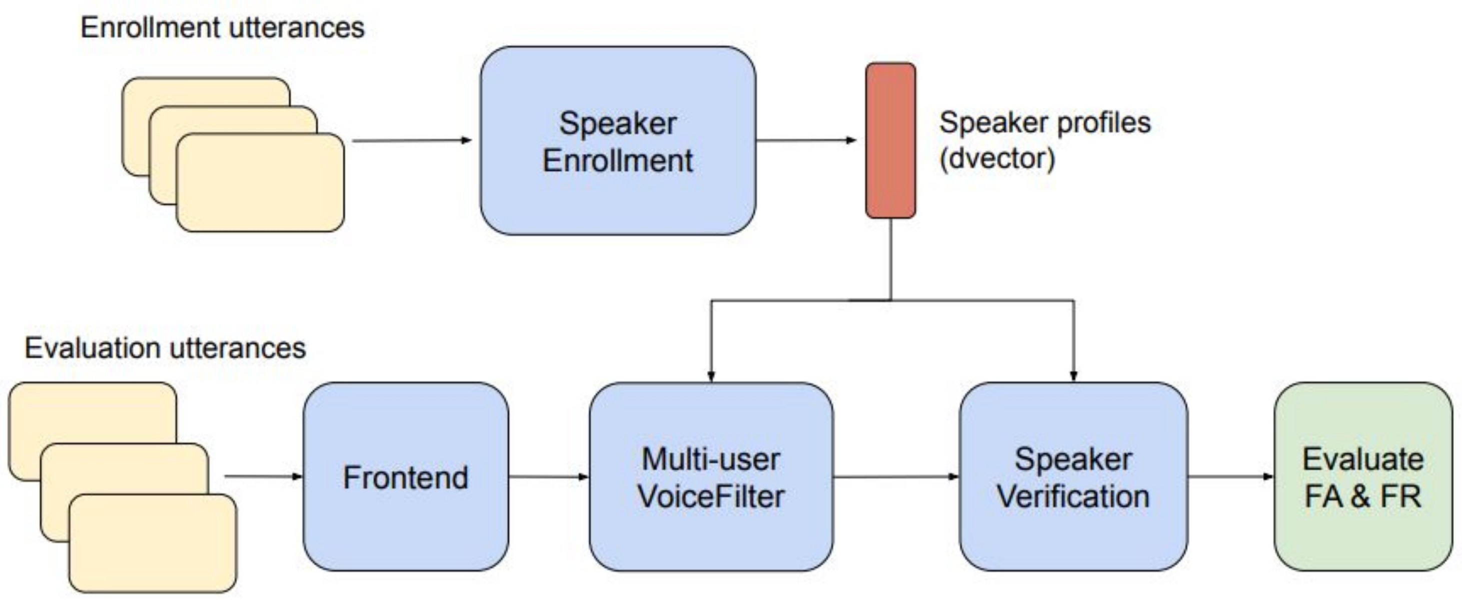
- The ***ScorerNet*** computes a similarity score between the KeyVector and each of the speaker embeddings and outputs a set of N attention weights
- The **Attended Embedding** is the dot product of the weights and the embedding inputs

Experimental setup

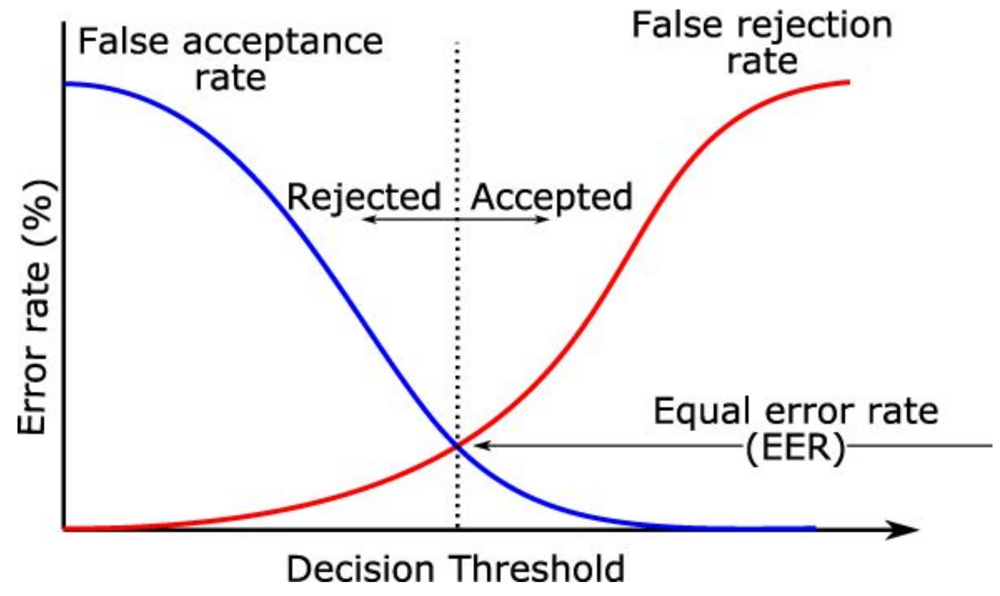
Data

- Webhound dataset
  - 958 speakers,
  - 220K utterances
- Noisified with MTR:
  - overlapping speech (other webhound speakers)
  - Non-speech noise
  - No noise (clean)
- Using 3 different SNR levels -5 dB, 0 dB and 5 dB

Evaluation Pipeline



Evaluation Metric



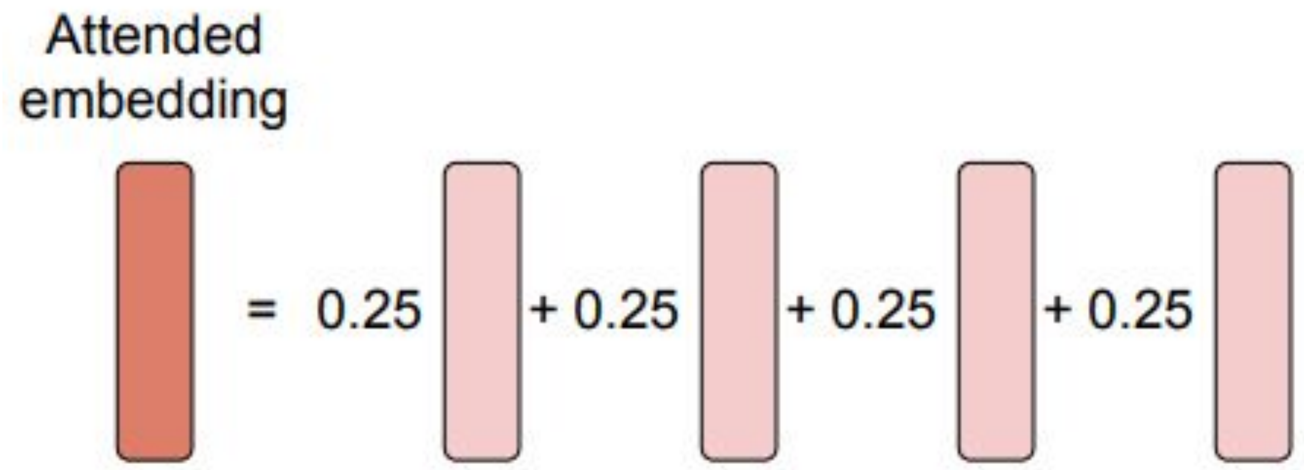
- We measure the impact that MUVF has on speaker verification accuracy via the **Equal Error Rate (EER)** metric.

We are optimizing for matched single-enrolled user performance (compared to current SUVF) and improved two-user performance (compared to previous MUVF).

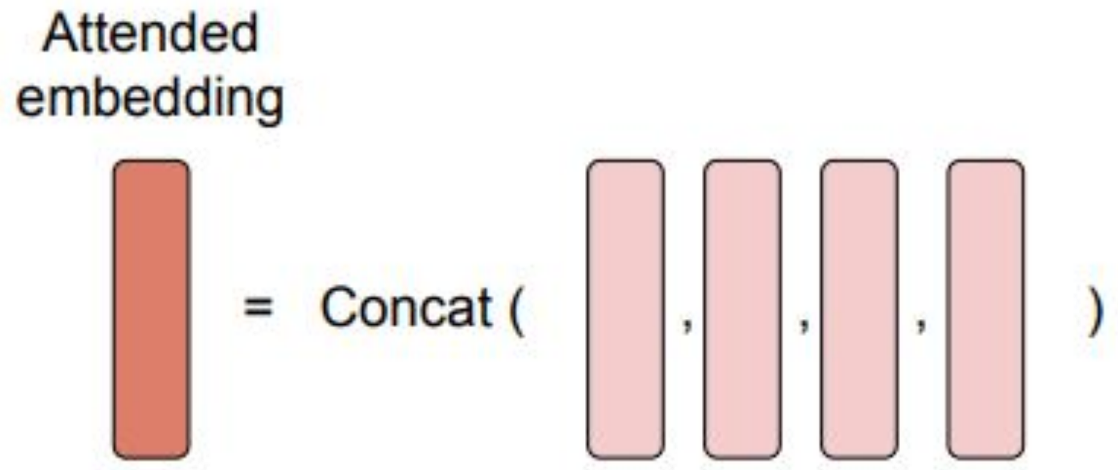
Experiment 1 : Is Attention necessary?

Models with no attention

Model 1 : Averaging Model

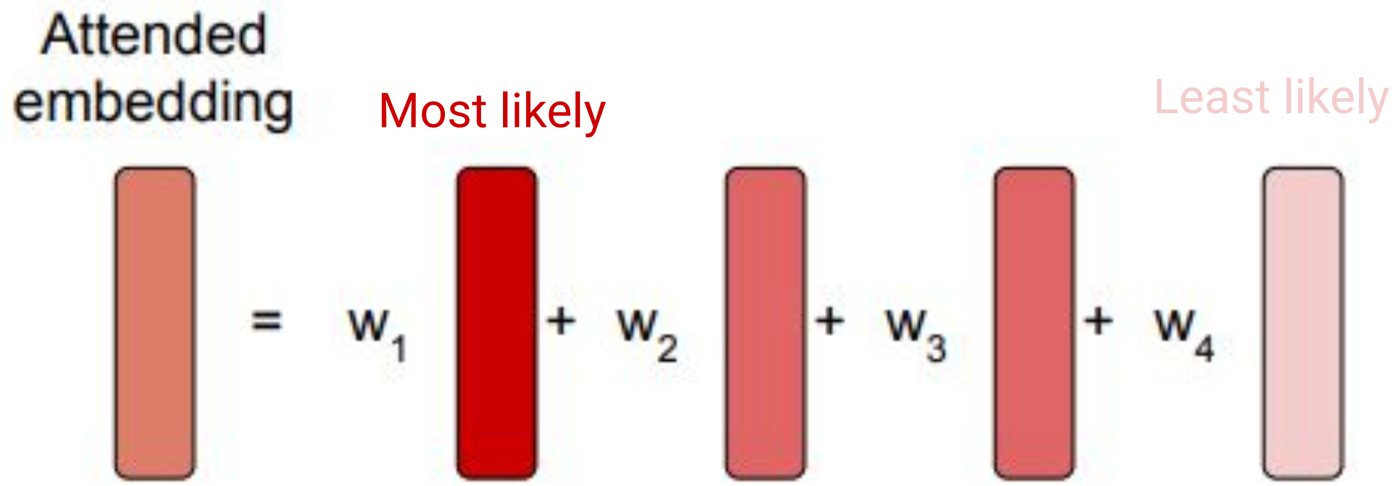


Model 2 : Concat Model

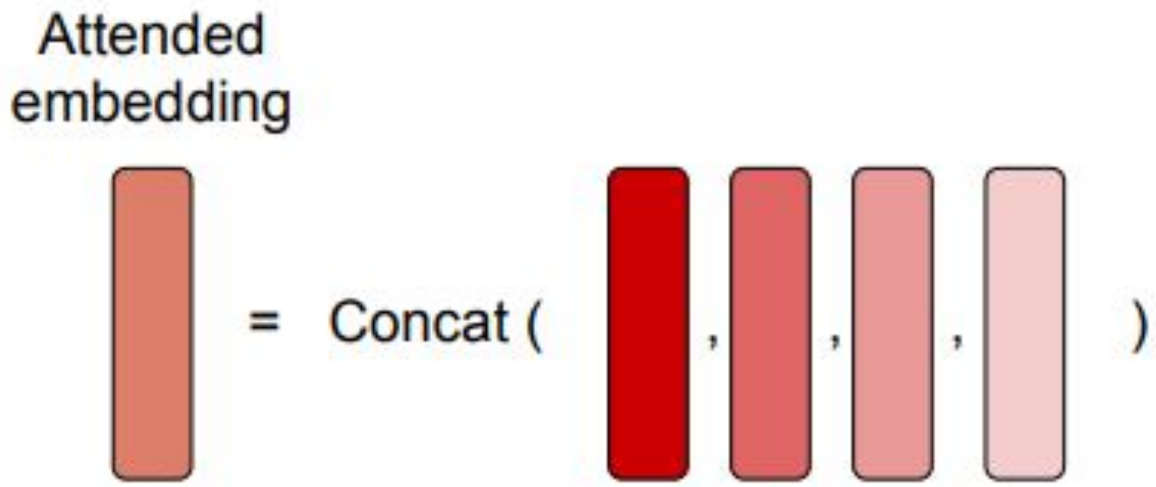


Models with attention

Model 3 : AttentionNet + Weighted Sum Model

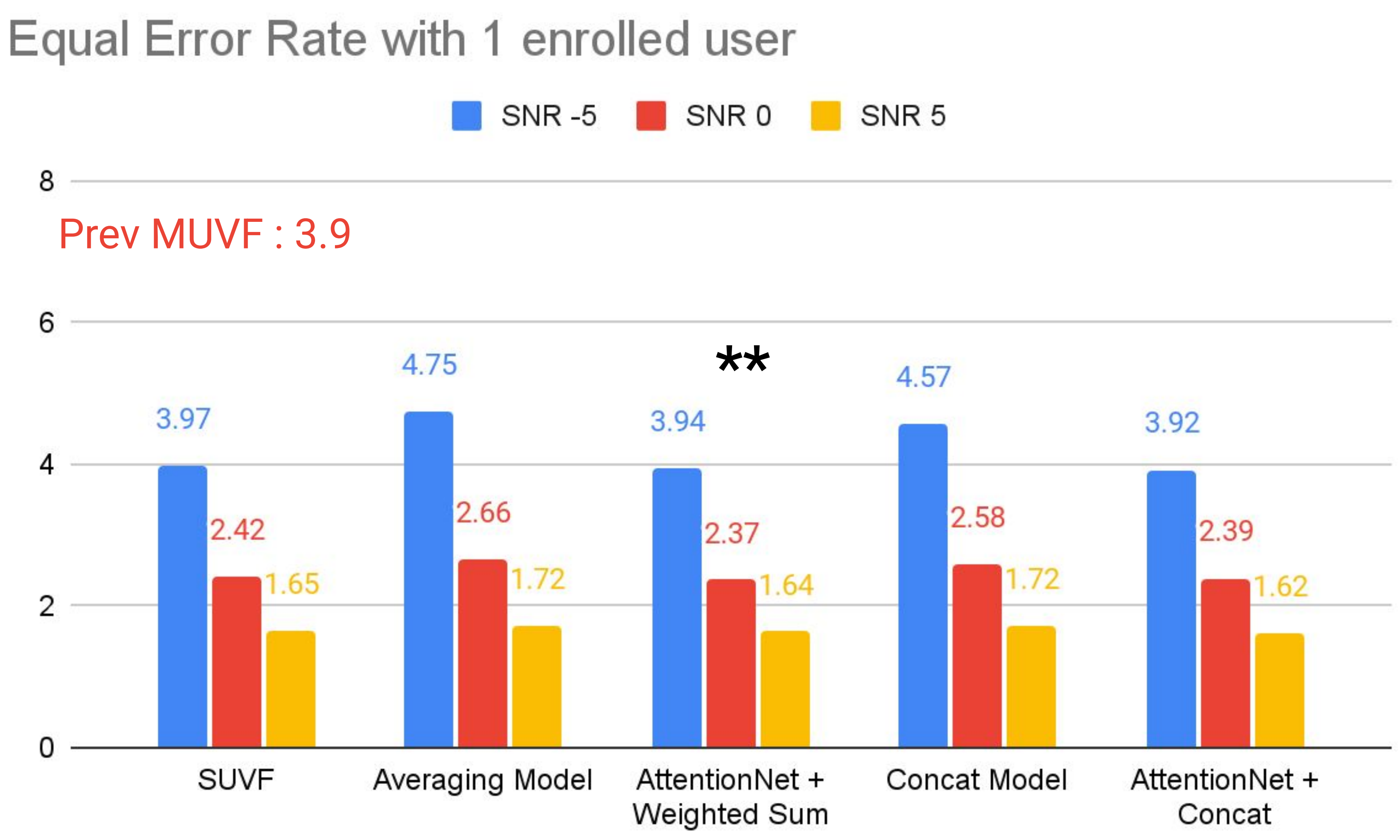


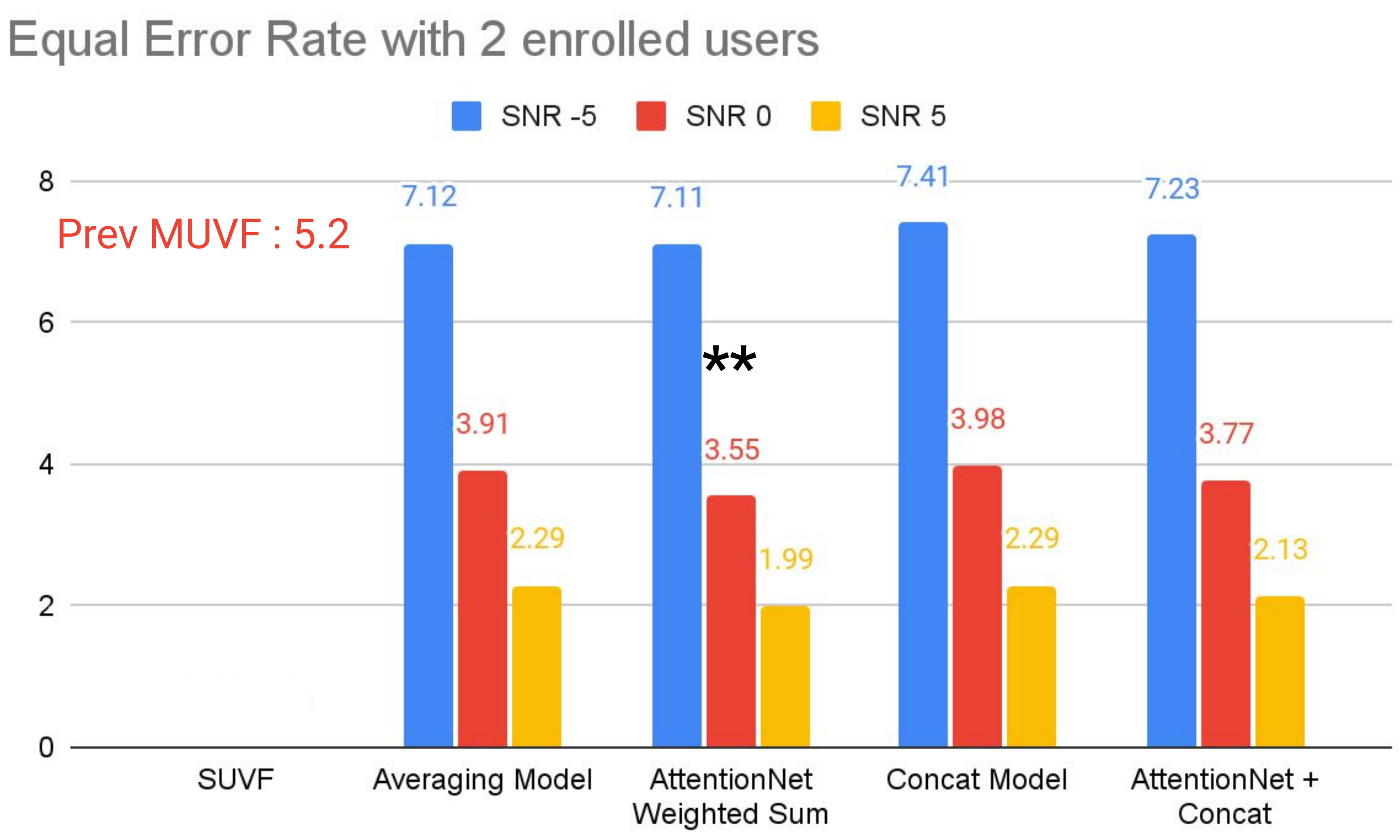
Model 4 : AttentionNet + Concat-Top-K Model



Note, colors indicate target speaker likelihood

Experiment 1 : Attention is **required** for accurate voice separation





- Compared with the no-attention variant, AttentionNet improves both 1-user and 2-user performance.
- Within the AttentionNet models, using a weighted sum (dot product) rather than concatenating the top 2 predicted speakers results in better 2-user performance.
    - The WeightedSum model (3.47 MB) is smaller than the Concat. Model (3.99 MB)

AttentionNet and *VoiceFilterNet* are trained by minimizing 3 loss functions

$$\boxed{\mathrm{L}_{\text{total}} = w_1 \mathrm{L}_{\text{asym}} + w_2 \mathrm{L}_{\text{noise}} + w_3 L_{att}}$$

Asymmetric reconstruction loss - ensures that the enhanced Spectrogram matches the clean spectrogram (Ground Truth)

$$\mathrm{L}_{\text{asym}} = \sum_t \sum_f \left(g_{asym}\left(S_{clean}(t,f) - S_{enh}(t,f), \alpha\right)\right)^2$$

Noise label prediction loss - ensures that predicted noise label is close to the ground truth label

$$\mathrm{L}_{\text{noise}} = \sum_i (n_{pred} - n_{gt})^2$$

Attention loss - minimizes the binary cross entropy between the attention weights and the ground truth embedding order.

$$L_{\text{att}} = \sum_t \mathrm{CrossEntropy}(\alpha^{(t)}, \mathbf{w}_{\text{gt}}) + \lambda ||\alpha^{(t)}||_\infty$$

Experiment 2 : Can we improve the training objective function?

Model 1 : Jointly trained VFNet and AttentionNet

$$\mathrm{L}_{\mathrm{total}} = w_1 \mathrm{L}_{\mathrm{asym}} + w_2 \mathrm{L}_{\mathrm{noise}} + w_3 \mathrm{L}_{\mathrm{attn}}$$

Current Parameter value

$$W_{t+1} = W_t - \alpha \frac{\partial L_{total}}{\partial t}$$

Updated Parameter value

Learning Rate

Model 2 : Dual Learning rate schedule

$$\mathrm{L}_{\mathrm{vf}} = w_1 \mathrm{L}_{\mathrm{asym}} + w_2 \mathrm{L}_{\mathrm{noise}}$$

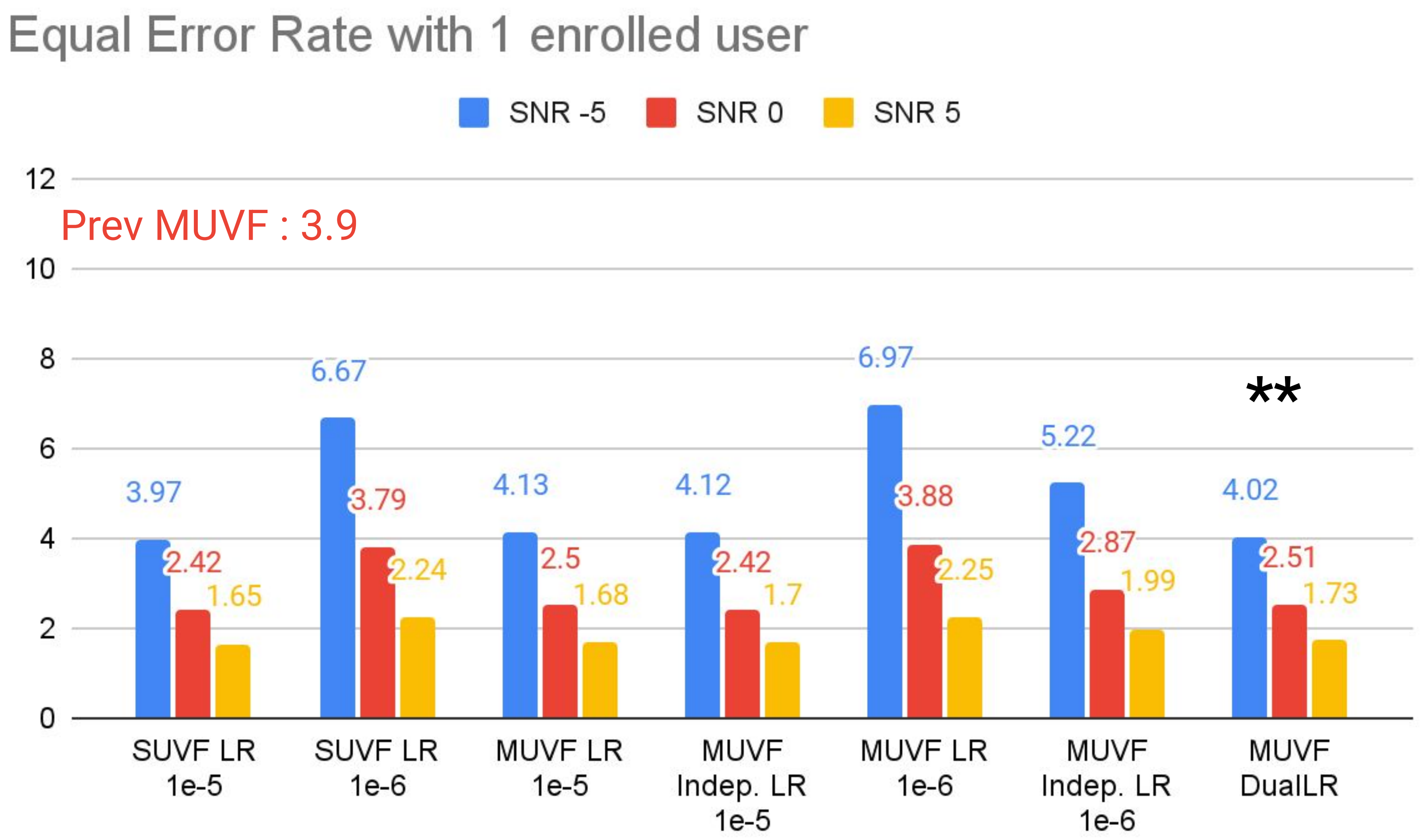
$$W_{t+1}^{vf} = W_t^{vf} - \alpha_{vf} \frac{\partial L_{vf}}{\partial t}$$

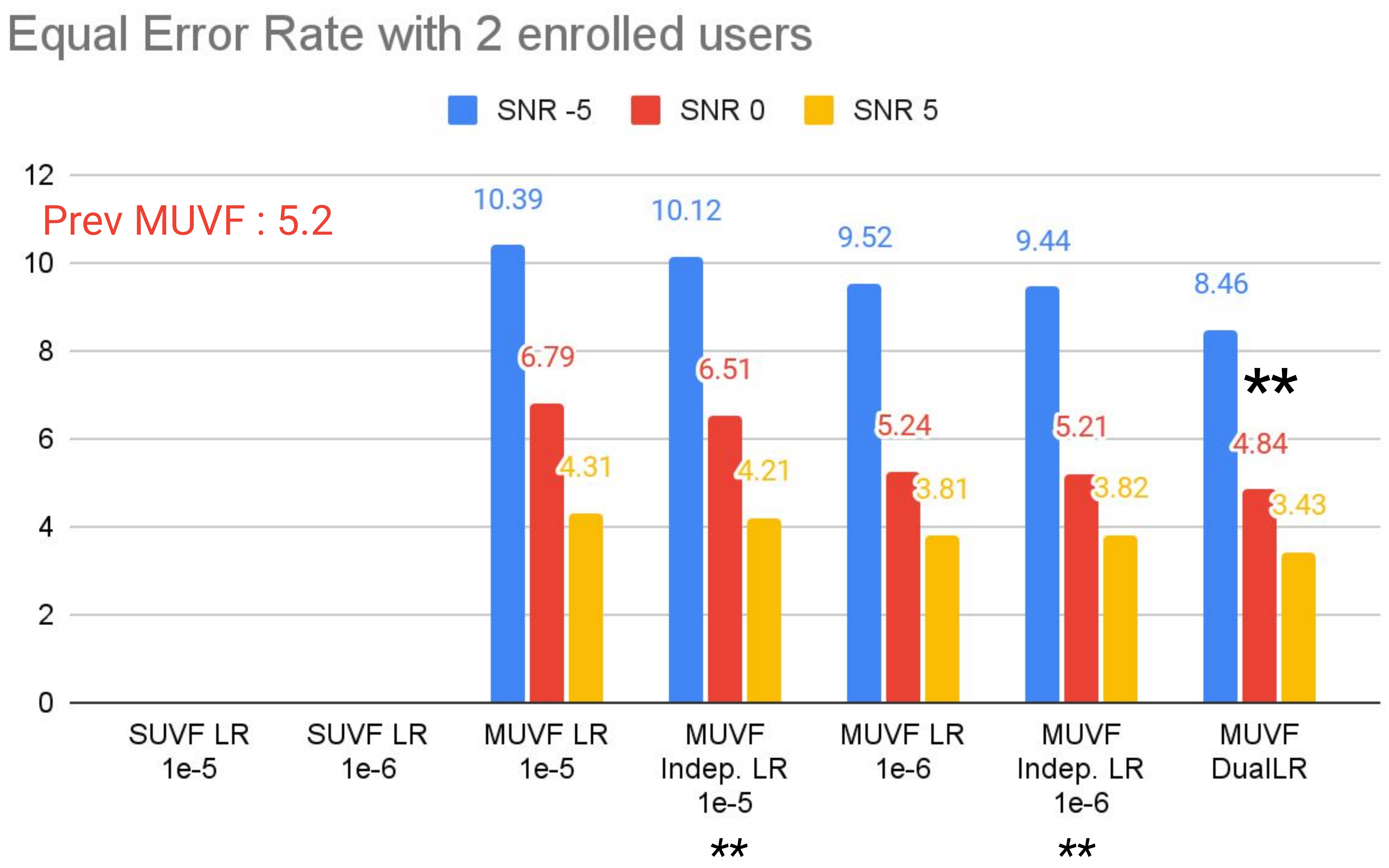
$$W_{t+1}^{attn} = W_t^{attn} - \alpha_{attn} \frac{\partial L_{attn}}{\partial t}$$

In the dual learning rate scheduler, VFNet and AttentionNet are trained with different learning rates.

Experiment 2 : Using a Dual Learning Rate Schedule improves performance

*** Independently trained with same LR*

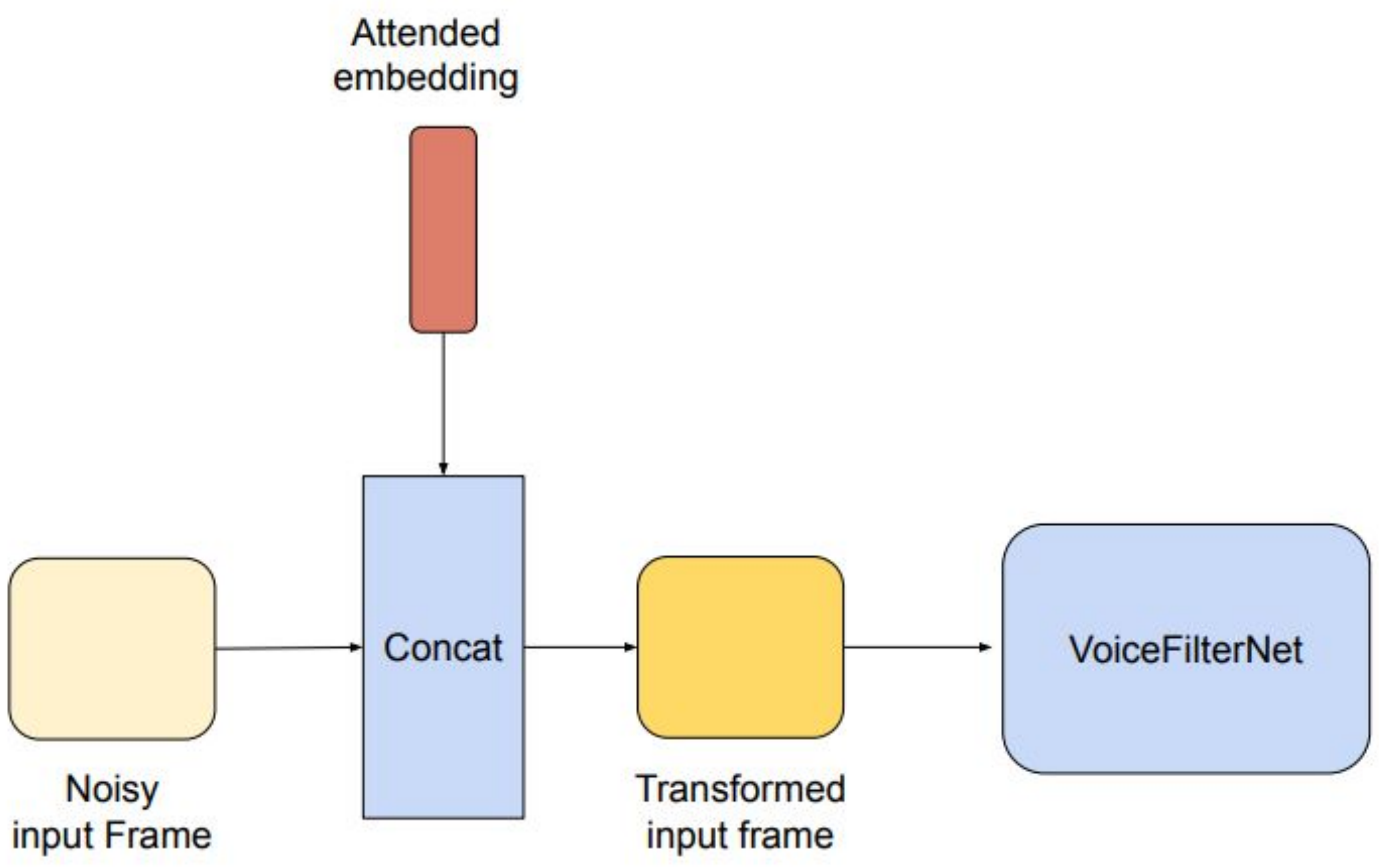




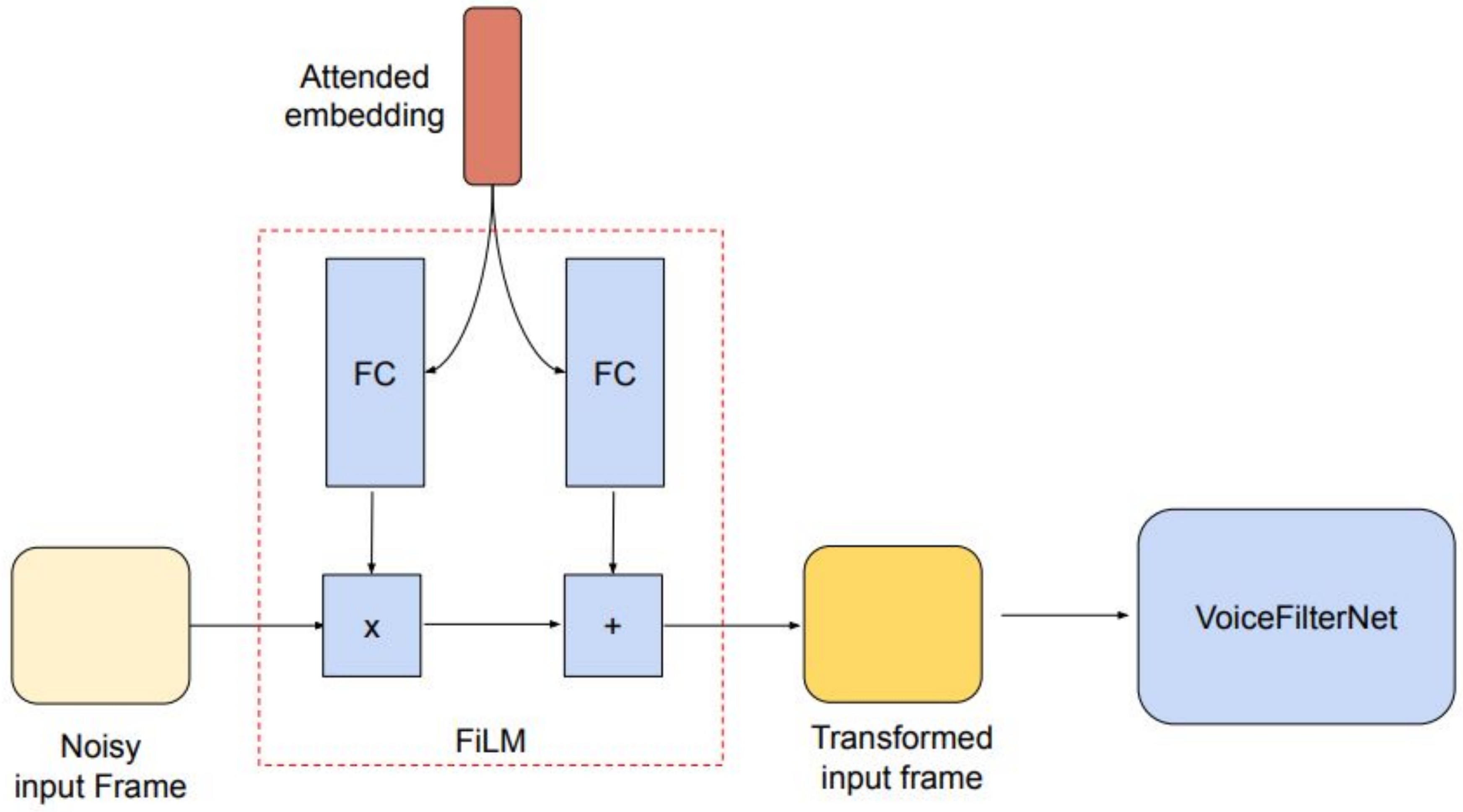
- Reducing the learning rate, increases the EER for SUVF and MUVF.
- Training *AttentionNet* independently but with same LR helps marginally.
- Using a 10^{-5} LR for the *VoiceFilterNet* and a 10^{-6} LR for the *AttentionNet* allows us to train the model for more steps, achieving better 1- and 2-enrolled user performance

Experiment 3 : Is conditioning via concatenation optimal?

Model 1 : Concat the attended embedding to each noisy input frame

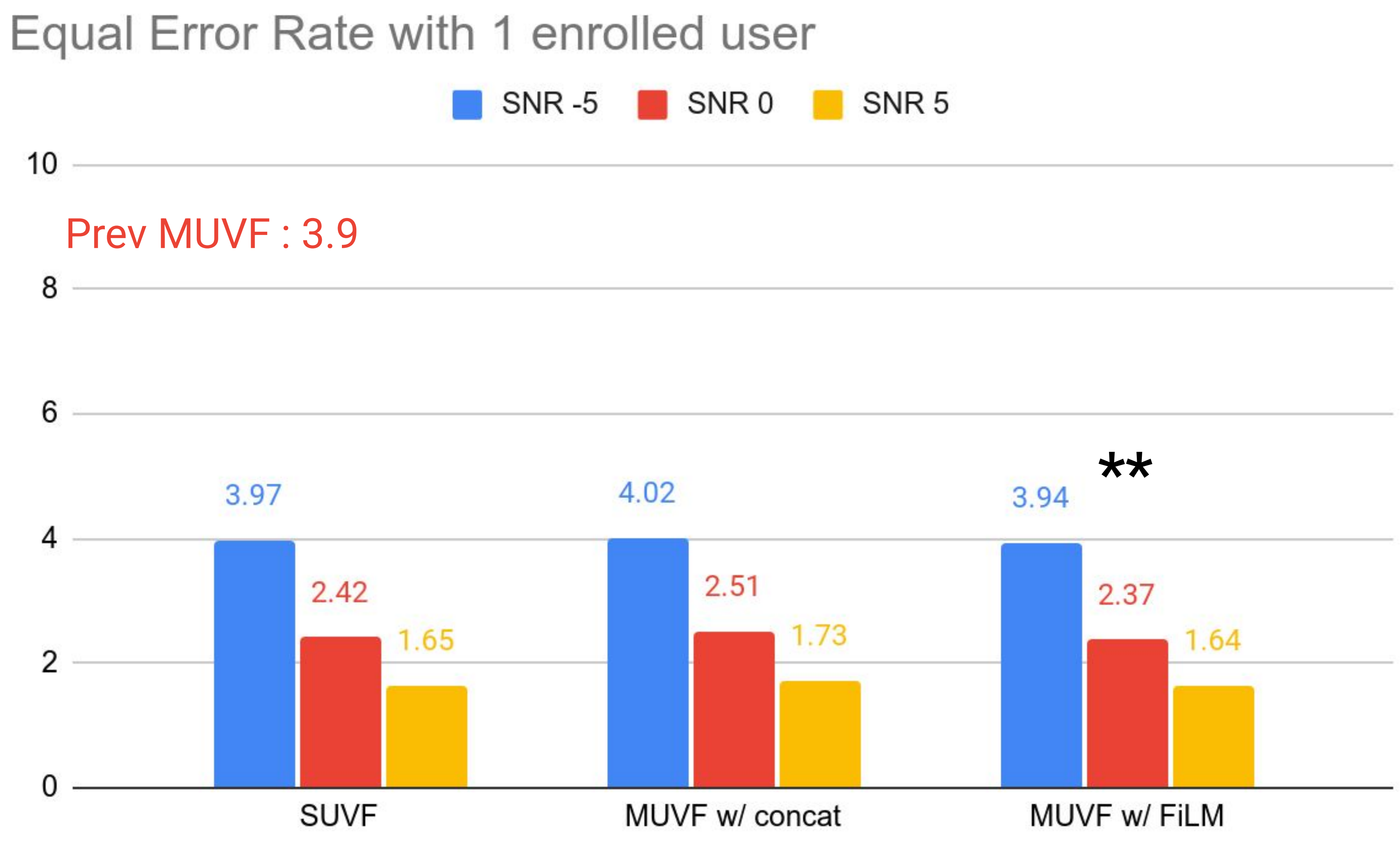


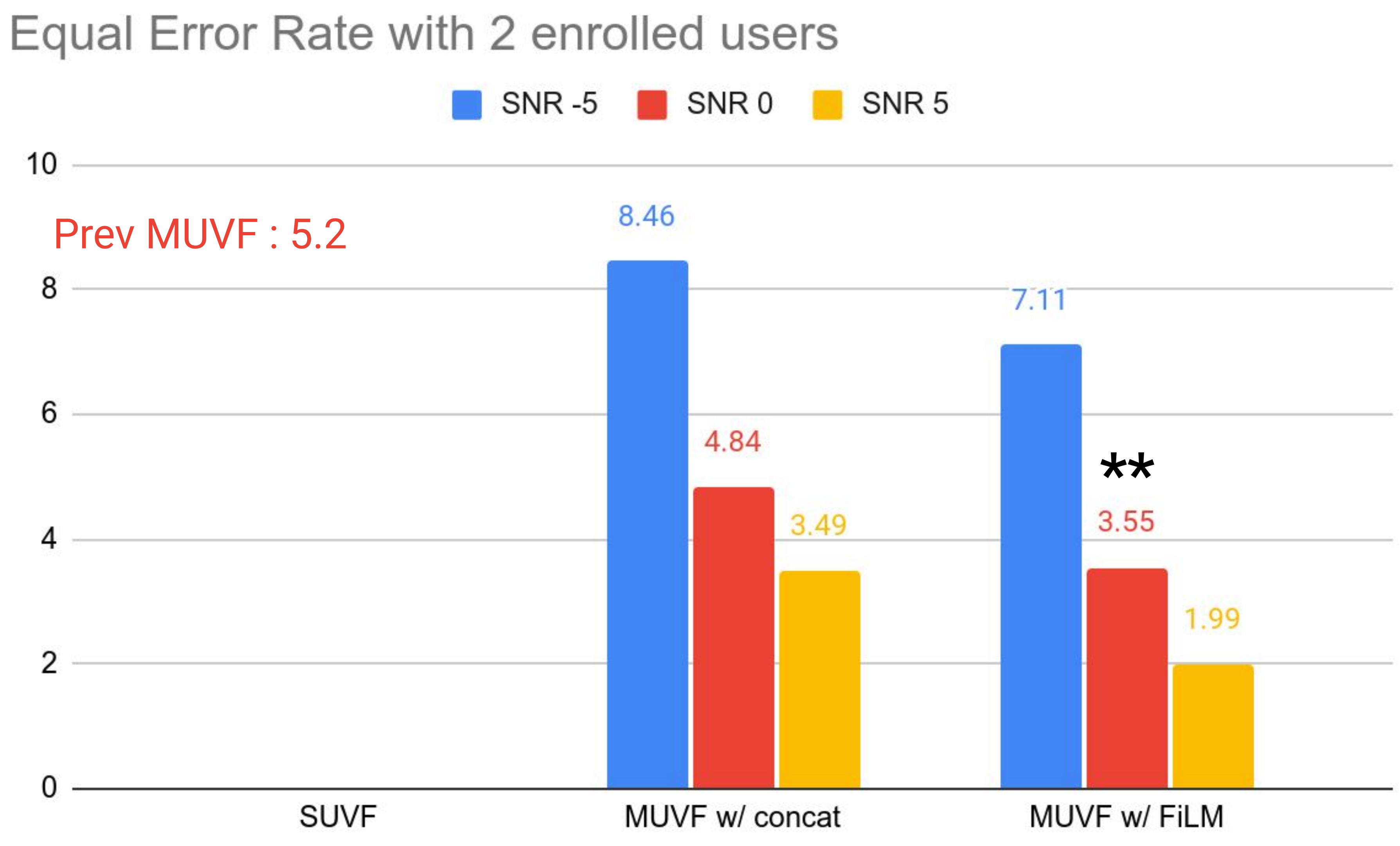
Model 2 : Use FiLM to transform each frame



FiLM applies an ***affine transformation*** to condition each input frame with the speaker embeddding in a feature-wise manner. The transformed input frame as the same size as the original input

Experiment 3 : Using a FiLM to condition the VoiceFilterNet is better





- Using FiLM to condition the VoiceFilterNet on the attended embedding significantly improves both 1- and 2-enrolled user performance.
- This is because, unlike concatenation, FiLM transforms the input depending on the values in the attended embedding.

Experiment 4 : Extending the multi-user VoiceFilter-Lite model to 4 enrolled users

Model Name	Number of enrolled speakers	EER(%) on Clean	EER(%) on Speech Noise		
			-5 dB	0 dB	5 dB
No VoiceFilter-Lite	-	0.71	12.40	8.29	5.13
Single-user VoiceFilter-Lite	1	0.71	3.97	2.42	1.65
Four-user VoiceFilter-Lite	1	0.71	3.88	2.37	1.60
	2	0.71	8.24	4.73	2.66
	3	0.72	9.78	5.38	2.99
	4	0.72	10.10	5.70	3.10

The same model architecture and training regime used for the two user model can be easily extended to support 4 enrolled users **without degrading** single user performance.

Summary

- Through a series of experiments, we found that:
  - **AttentionNet** is required for accurate speaker selection by computing the most likely speaker given a frame.
  - Training the AttentionNet with a **slower learning rate** than the VoiceFilterNet prevents overfitting and results in a better model.
  - Using **FiLM** to condition the VoiceFilterNet with the attended embedding also improves performance of the model.
- The multi-user VoiceFilter-Lite (MUVF) achieves identical single-user performance as the original VoiceFilter-Lite model (SUVF).
- We observe a degradation in performance with more enrolled users. This is because the *AttentionNet* has a difficult task of selecting the correct speaker from noisy input.
  - Our future work aims at addressing this discrepancy.

Thank you.