SPEAKER DISENTANGLEMENT OF SPEECH PRE-TRAINED MODEL BASED ON INTERPRETABILITY

Xiaoxu Zhu[1,*] *Junhua Li*[2,*] *Aaron J. Li*[3] *Yiming Ren*[4] *Baoxiang Li*[4,†]

[1]Department of Industrial Engineering, Tsinghua University, Beijing, China
[2]China National Aviation Fuel Group Limited, Beijing, China
[3]University of California, Berkeley, CA, USA
[4]Shanghai Artificial Intelligence Laboratory, Beijing, China

ABSTRACT

Self-supervised speech models learn representations that capture both content and speaker information. Yet this entanglement creates problems: content tasks suffer from speaker bias, and privacy concerns arise when speaker identity leaks through supposedly anonymized representations. We present two contributions to address these challenges. First, we develop InterpTRQE-SptME (**T**imbre **R**esidual **Q**uantitative **E**valuation Benchmark of **S**peech **p**re-**t**raining **M**odels **E**ncoding via **Interp**retability), a benchmark that directly measures residual speaker information in content embeddings using SHAP-based interpretability analysis. Unlike existing indirect metrics, our approach quantifies the exact proportion of speaker information remaining after disentanglement. Second, we propose InterpTF-SptME, which uses these interpretability insights to filter speaker information from embeddings. Testing on VCTK with seven models including HuBERT, WavLM, and ContentVec, we find that SHAP Noise filtering reduces speaker residuals from 18.05% to nearly zero while maintaining recognition accuracy (CTC loss increase under 1%). The method is model-agnostic and requires no retraining.

Index Terms— Speech pre-trained models, speaker disentanglement, interpretability, SHAP, privacy

1. INTRODUCTION

Recent advances in self-supervised speech pre-trained models such as wav2vec 2.0 [1], HuBERT [2], and WavLM [3] have revolutionized speech processing. These models learn rich representations that capture various aspects of speech including content, speaker identity, and paralinguistic information across different layers [4].

But there's a problem: these models mix content and speaker information in ways that hurt performance. When doing ASR, leftover speaker traits cause errors and biases. Privacy is another concern—speaker identity often leaks through "anonymized" representations, which matters for initiatives like VoicePrivacy [5].

Existing solutions like ContentVec [6] use elaborate training schemes with voice conversion and contrastive learning. But they have two major limitations: no direct way to measure how well they separate speakers, and they need custom architectures and training procedures.

We tackle both problems using interpretability. SHAP [7] explanations don't just show what models do—they can guide how to fix them. By analyzing which embedding dimensions contribute to speaker identification, we can measure and then remove speaker information. Our contributions are:

- InterpTRQE-SptME benchmark: the first direct metric for speaker residuals in content embeddings, using SHAP to quantify what previous work could only estimate indirectly
- InterpTF-SptME filtering: a practical method that uses interpretability insights to remove speaker information post-hoc, requiring no model retraining
- Evidence from seven models that our approach works universally, with SHAP Noise reducing residuals to 0% while maintaining recognition accuracy

2. RELATED WORK

Self-supervised learning has transformed speech representation learning in recent years. Wav2vec 2.0 [1] introduced masked prediction with contrastive loss, while HuBERT [2] employs iterative clustering to predict masked acoustic units. WavLM [3] extends these approaches with denoising objectives to improve speaker-related tasks, and DPHuBERT [8] focuses on efficiency through distillation and pruning.

Efforts to separate speaker and content information have explored various architectural constraints. AutoVC [9] enforces information bottlenecks to limit speaker leakage, while ContentVec [6] achieves state-of-the-art disentanglement through teacher-student training with explicit speaker

[0]*Equal contribution. †Corresponding author.

conditioning. However, these approaches evaluate disentanglement indirectly through downstream task performance, lacking direct measurement of residual speaker information.

Interpretability methods like SHAP [7] and LIME [10] have seen widespread adoption in NLP and computer vision but remain underexplored in speech processing. Existing work includes LIME for phoneme recognition [11] and gradient-based analysis for ASR [12], but interpretability has not been applied to understand or mitigate speaker entanglement in speech representations.

3. METHODOLOGY

3.1. InterpTRQE-SptME Benchmark

The InterpTRQE-SptME benchmark provides the first direct quantification of speaker information residuals in speech pre-trained model encodings. Unlike previous indirect metrics that rely on downstream task performance, our approach directly measures the proportion of speaker information in content embeddings through interpretability analysis.

Feature Extraction. From VCTK audio samples [13], we extract:

- Content embeddings: $E_c = M^{(l)}(x) \in \mathbb{R}^{T \times d_c}$, where l is the target layer, T is the number of frames, and d_c is the embedding dimension (768 or 1024)
- Speaker embeddings: $E_s = S(x) \in \mathbb{R}^{d_s}$, extracted using a pre-trained ECAPA-TDNN model [14] from SpeechBrain [15], where $d_s = 192$

Speaker Classification. After time-averaging content embeddings, we concatenate with speaker embeddings:

$$E_{concat} = [\text{mean}_T(E_c), E_s] \in \mathbb{R}^{d_c+d_s} \tag{1}$$

A 4-layer MLP predicts speaker identity from these concatenated features:

$$\hat{y} = f(E_{concat}) \tag{2}$$

SHAP Analysis. Gradient SHAP [7] reveals which features drive speaker classification. SHAP values ϕ_i represent each feature's contribution to the model output by calculating the average marginal contribution across all possible feature coalitions. Our residual metric is the proportion of decision coming from content embeddings:

$$R_M = \frac{\sum_{i=1}^{d_c} |\phi_i|}{\sum_{i=1}^{d_c} |\phi_i| + \sum_{j=1}^{d_s} |\phi_{d_c+j}|} \tag{3}$$

Perfect disentanglement would yield 0%, while high percentages indicate speaker information leakage. The SHAP values follow a distribution where positive values enhance speaker identification while negative values suppress it, as shown in Figure 2.

3.2. InterpTF-SptME: Interpretability-based Filtering

SHAP analysis reveals which dimensions encode speakers. We use this to design two filters:

SHAP Noise Method. Add noise scaled by each dimension's speaker contribution:

$$\hat{\phi}_c = \frac{\phi_c - \mu_{\phi_c}}{\sigma_{\phi_c}} \tag{4}$$

$$n_{shap} = \hat{\phi}_c \cdot \epsilon \cdot |\sigma| + \mu \tag{5}$$

$$E'_c = E_c + n_{shap} \tag{6}$$

where $\epsilon \sim \mathcal{N}(0, I)$ is Gaussian noise, σ controls the noise scale (negative values), and μ is the offset (set to 0 in experiments).

SHAP Cropping Method. Suppress dimensions that encode speaker information:

$$m_i = \begin{cases} 0, & \text{if } \phi_i \in \text{top-}r\% \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

$$E'_c = E_c \odot (m + \alpha \cdot (1 - m)) \tag{8}$$

where r is the cropping ratio and $\alpha \in (0, 1)$ is the normalized cutting weight that controls cropping strength.

4. EXPERIMENTS

4.1. Experimental Setup

Our experiments utilize the VCTK corpus [13], selecting 20 speakers with 7,758 utterances for evaluation. All audio follows standard preprocessing: 16kHz sampling rate, 16-bit mono WAV format, with energy normalization to -3dB.

We evaluate seven speech pre-trained models spanning different architectures and training objectives: HuBERT BASE [2] (layer 9), HuBERT LARGE [2] (layers 18 and 21), DPHuBERT [8] (layer 12), ContentVec [6] (layer 12), WavLM Base+ [3] (layer 12), Whisper-ppg (encoder output), and HuBERT-CH (Chinese fine-tuned, layer 9).

To assess speaker disentanglement comprehensively, we measure three aspects: timbre residual ratio quantified through InterpTRQE-SptME, content preservation via CTC loss using the Hubert-Large-Finetuned ASR model, and stability through batch-wise standard deviation of SHAP values.

The experimental framework leverages PyTorch and HuggingFace Transformers for model inference. SHAP analysis uses Captum's GradientShap implementation with 256 baseline samples randomly selected from the entire dataset to ensure consistency. The speaker classifier consists of a 4-layer MLP architecture (input→2048→1256→64→20 speakers) trained with cross-entropy loss using Adam optimizer (learning rate 1e-4, batch size 32) for 50 epochs until reaching 100% training accuracy. For ASR evaluation, we employ the Hubert-Large-Finetuned model from HuggingFace

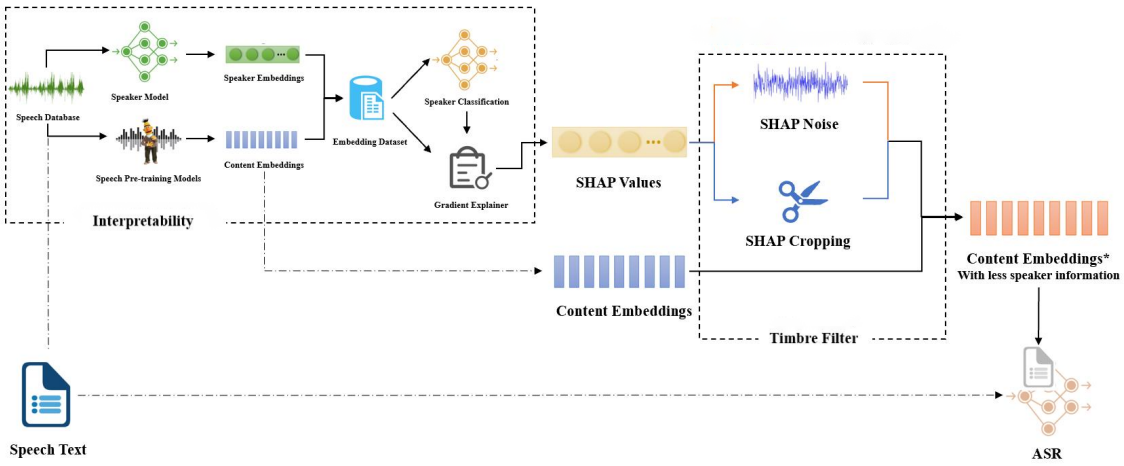


Fig. 1: Overview of our framework combining InterpTRQE-SptME benchmark (left) for quantifying speaker residuals and InterpTF-SptME filtering methods (right) for removing speaker information from content embeddings.

Table 1: Timbre residual quantification results on VCTK dataset

Model	Timbre Residual (%)
HuBERT BASE (L9)	13.72
HuBERT LARGE (L18)	10.58
HuBERT LARGE (L21)	18.65
DPHuBERT (L12)	7.73
ContentVec (L12)	**5.20**
WavLM Base+ (L12)	9.02
Whisper-ppg (Enc)	7.46
HuBERT-CH (L9)	13.93

(facebook/hubert-large-ls960-ft), which has been fine-tuned on LibriSpeech 960h. CTC loss computation occurs between model logits and ground-truth transcriptions from VCTK, with embedding substitution implemented through forward hooks at the target layer. Code and experimental scripts are available at `https://github.com/zhuxiaoxuhit/InterpTRQE-SptME`.

4.2. Benchmark Results

Table 1 shows residual speaker information across models. ContentVec performs best at 5.20%—not surprising given its design goal. Notice how HuBERT LARGE gets worse in deeper layers (10.58% at L18 jumps to 18.65% at L21), confirming that later layers mix in more speaker traits.

The quantification results reveal distinct patterns across models. ContentVec achieves the lowest residual at 5.20% through its explicit disentanglement architecture—voice conversion removes speaker information from teacher labels while contrastive loss enforces speaker invariance. Similar low residuals appear in Whisper-ppg (7.46%) and DPHuBERT (7.73%), reflecting their content-focused design objectives.

In the middle range, WavLM Base+ shows 9.02% residual, reflecting its multi-task training that balances content extraction with speaker-dependent tasks. HuBERT LARGE's layer 18 exhibits comparable performance at 10.58%, suggesting this intermediate layer maintains reasonable content-speaker separation.

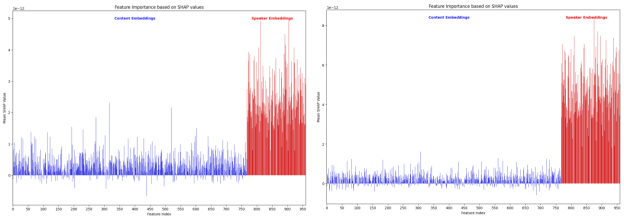
The highest residuals emerge in models without explicit disentanglement mechanisms. HuBERT BASE and HuBERT-CH show 13.72% and 13.93% respectively, while HuBERT LARGE's layer 21 reaches 18.65%. This pattern indicates that clustering-based training inherently preserves speaker characteristics, with deeper layers progressively accumulating more speaker-specific information.

Table 2: Filtering results on HuBERT LARGE layer 21

Method	Params	CTC Loss	Residual
Original	-	1.261	18.65%
SHAP Noise	$\sigma = -1.0$	1.325 (+5.1%)	**0%**
	$\sigma = -0.6$	1.273 (+0.9%)	2.21%
	$\sigma = -0.3$	1.263 (+0.1%)	7.23%
SHAP Crop	$\alpha = 0.99$	1.297 (+2.8%)	4.65%
	$\alpha = 0.58$	1.278 (+1.3%)	10.07%

4.3. Filtering Results

We tested filtering on HuBERT LARGE layer 21—the worst performer at 18.65% residual. Table 2 shows both methods work, with different trade-offs.



(a) HuBERT BASE (b) ContentVec

Fig. 2: SHAP value distribution comparison showing speaker (orange) vs content (blue) contributions. ContentVec shows significantly reduced speaker contribution.

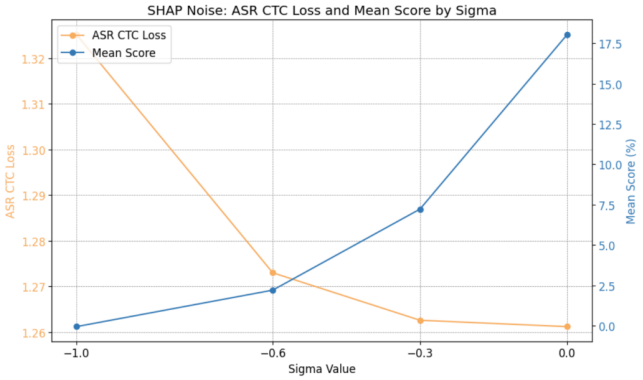


Fig. 3: Trade-off between speaker residual reduction (Mean Score) and content preservation (CTC Loss) for SHAP Noise filtering. σ controls noise scale (negative values indicate proportion of SHAP-weighted noise added).

Note: For SHAP Crop, α values are normalized by the maximum content embedding value of 17.1795.

SHAP Noise eliminates speaker residuals entirely at $\sigma = -1.0$, though recognition accuracy drops 5.1%. The sweet spot is $\sigma = -0.6$: just 2.21% residual with negligible recognition loss (+0.9%).

As shown in Figure 3, SHAP Noise exhibits a clear trade-off: exponential residual reduction with increasing $|\sigma|$, while CTC loss rises sharply after $\sigma = -0.8$. The optimal balance occurs at $\sigma = -0.6$.

4.4. Analysis and Discussion

Figure 2 visualizes the SHAP value distributions for HuBERT BASE and ContentVec, clearly showing ContentVec's superior disentanglement.

The superior performance of SHAP Noise stems from its treatment of negative SHAP values. Since timbre perception relies on relative frequency patterns and phase relationships, both positive and negative feature contributions carry speaker information. SHAP Cropping with $r = 1$ suppresses only positive-valued dimensions, effectively ignoring half the speaker-encoding features. SHAP Noise, by modulating all dimensions proportionally to their absolute SHAP values, addresses the complete speaker representation. This comprehensive approach enables SHAP Noise to achieve near-complete speaker removal (0% residual) compared to SHAP Cropping's 4.65% plateau.

Figure 3 illustrates the relationship between filtering strength and content preservation for SHAP Noise. The parameter $\sigma = -0.6$ emerges as an optimal operating point, reducing speaker residuals by 87.8% while increasing CTC loss by only 0.93%. This minimal impact on recognition accuracy demonstrates the method's practical viability for real-world applications.

4.5. Layer Sensitivity Analysis

We investigated the speaker information distribution across HuBERT LARGE layers. Testing layers 18 and 21 revealed significant differences in speaker residuals and ASR performance.

Analysis across HuBERT LARGE layers reveals increasing speaker information accumulation with depth. Layer 18 contains 10.58% speaker residual, which increases to 18.65% at layer 21. This progression challenges conventional wisdom about using the deepest layers for content-related tasks.

Examination of the Hubert-Large-Finetuned ASR model provides additional insight into layer utilization. The model fine-tunes only layers 18-21 while keeping earlier layers frozen. Layer 18 embeddings yield a CTC loss of 1.217, nearly matching the 1.219 achieved with raw audio, whereas layer 21 degrades to 1.261. These results indicate that the ASR model primarily exploits layer 18 representations.

Regarding filtering applicability, layers exhibit different sensitivities to perturbation. Layers 21-23 maintain functionality under noise injection, while layers 18-19 collapse with any modification. This robustness characteristic motivated our selection of layer 21 for demonstrating the filtering methods.

5. CONCLUSION

We demonstrated that interpretability methods can both quantify and mitigate speaker entanglement in speech models. InterpTRQE-SptME directly measures residual speaker information using SHAP analysis, while InterpTF-SptME filtering effectively removes it—reducing residuals from 18.65% to near zero with minimal recognition loss ($\leq$0.93%). The key insight is using SHAP explanations as actionable feedback to identify and suppress speaker-encoding dimensions. This model-agnostic approach requires no retraining and opens paths for addressing other attributes beyond speaker identity.

6. REFERENCES

[1] Alexei Baevski, Yongqiang Zhou, Abdelrahman Mohamed, et al., "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12449–12460, 2020.

[2] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, et al., "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[3] Sanyuan Chen, Chengyi Wang, Zhuo Chen, et al., "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[4] Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, et al., "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1179–1210, 2022.

[5] Natalia Tomashenko, Brij Mohan Lal Srivastava, Xin Wang, et al., "Introducing the voiceprivacy initiative," in *Interspeech*, 2020, pp. 1693–1697.

[6] Kaizhi Qian, Yang Zhang, Shiyu Chang, et al., "Contentvec: An improved self-supervised speech representation by disentangling speakers," in *International Conference on Machine Learning*, 2022, pp. 18003–18017.

[7] Scott M Lundberg and Su-In Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[8] Yifan Peng, Yui Sudo, Shakeel Muhammad, and Shinji Watanabe, "Dphubert: Joint distillation and pruning of self-supervised speech models," in *Interspeech*, 2023, pp. 1917–1921.

[9] Kaizhi Qian, Yang Zhang, Shiyu Chang, et al., "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5210–5219.

[10] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[11] Xiaoliang Wu, Peter Bell, and Ajitha Rajan, "Can we trust explainable ai methods on asr? an evaluation on phoneme recognition," *arXiv preprint arXiv:2305.18011*, 2023.

[12] Hemant Bharadhwaj, "Layer-wise relevance propagation for explainable deep learning based speech recognition," in *2018 IEEE International symposium on signal processing and information technology (ISSPIT)*. IEEE, 2018, pp. 168–174.

[13] Christophe Veaux, Junichi Yamagishi, and Kirsten MacDonald, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2017.

[14] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Interspeech*, 2020, pp. 3830–3834.

[15] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, et al., "Speechbrain: A general-purpose speech toolkit," *arXiv preprint arXiv:2106.04624*, 2021.