Self-Supervised Contrastive Learning for Content-Centric Speech Representation

Jinlong Li[1,2],Ling Dong[1,2],Wenjun Wang[1,2],Zhengtao Yu[*1,2], Shengxiang Gao[1,2]
1. Faculty of Information Engineering and Automation,
Kunming University of Science and Technology, Kunming, Yunnan, 650500, China
2.Yunnan Key Laboratory of Artifcial Intelligence, Kunming, Yunnan, 650500, China
jinlongli@stu.kust.edu.cn,ling.dong@kust.edu.cn
20203104003@stu.kust.edu.cn,ztyu@hotmail.com
gaoshengxiang.yn@foxmail.com

Abstract

Self-supervised learning (SSL) speech models have achieved remarkable performance across various tasks, with the learned representations often exhibiting a high degree of generality and applicability to multiple downstream tasks. However, these representations contain both speech content and some paralinguistic information, which may be redundant for content-focused tasks. Decoupling this redundant information is challenging. To address this issue, we propose a Self-Supervised Contrastive Representation Learning method (SSCRL), which effectively disentangles paralinguistic information from speech content by aligning similar content speech representations in the feature space using self-supervised contrastive learning with pitch perturbation and speaker perturbation features. Experimental results demonstrate that the proposed method, when fine-tuned on the LibriSpeech 100-hour dataset, achieves superior performance across all content-related tasks in the SUPERB Benchmark, generally outperforming prior approaches.

Keywords: Self-Supervised Fine-Tuning , Feature Disentanglement , Pre-trained Speech Model , Contrastive Learning

1 Introduction

Self-supervised speech models, such as HuBERT (Hsu et al., 2021) and WavLM (Chen et al., 2022), have advanced the field of speech processing by generating versatile representations through the exploitation of large-scale unlabeled data. These pre-trained models provide effective representations and initialization for downstream tasks (Evain et al., 2021; Chang et al., 2021), greatly facilitating applications like Automatic Speech Recognition (ASR), Phoneme Recognition (PR), and Speaker Identification (SID), among others. Compared with models trained from scratch, self-supervised models notably reduce training time and computational costs while delivering superior performance (Chang et al., 2021; Chan and Ghosh, 2022).

Despite the broad applicability offered by the high versatility of these models, they also present certain challenges in content-centric tasks. Particularly in tasks that require higher semantic consistency for similar content, the performance of SSL models may be constrained (Wang and Liu, 2021; Wang and Isola, 2020). This is primarily because learned speech representations often conflate linguistic content with paralinguistic cues. For instance, variations in pitch, background noise, and speaker identity can induce substantial feature disparities in the same semantic content, thereby diminishing the semantic relevance of semantically equivalent representations within the feature space. This misalignment not only impacts the accuracy of downstream tasks but also limits the model's generalization ability when the distribution of pre-training data diverges from that of the target domain data.

To address these issues, researchers have explored methods to disentangle content representations from speaker characteristics. For instance, the ContentVec model proposed by Kaizhi Qian et al. (Qian et al., 2022) achieves speaker disentanglement, thereby enhancing content-specific representations. Despite its effectiveness, this approach requires substantial data and computational resources. Peyser et al.

* Corresponding Author. email: ztyu@hotmail.com

introduced Disentangled Speech Representations (DSR) (Peyser et al., 2022), which separate content and contextual information through self-supervised learning, offering new directions for speech recognition and generation tasks. However, this method still demands significant computational resources. Chang et al. developed a self-supervised fine-tuning approach called Spin (Chang et al., 2023), which uses speaker-invariant clustering to improve content-related representations. While ContentVec requires substantially higher training time and resource costs, our method achieves comparable performance with only 45 minutes of fine-tuning on the 356-hour LibriSpeech dataset (Panayotov et al., 2015). However, this approach exclusively relies on speaker perturbation while neglecting prosodic variations across different intonations from the same speaker, which may degrade model performance in downstream tasks involving same-semantic different-prosody scenarios. Building on this, Meghanani et al. proposed the SCORE (Meghanani and Hain, 2024b) and LASER (Meghanani and Hain, 2024a) methods, which randomly decouple content-related representations from speed-perturbed audio and achieve satisfactory results with only one-third of the data required by Spin, albeit at the cost of some performance degradation compared to Spin. Therefore, existing decoupling frameworks exhibit inherent trade-offs between computational costs (e.g., training expenses in GPU hours) and disentanglement quality, while data-efficient fine-tuning strategies for resource-constrained scenarios remain systematically underexplored in both theoretical and practical dimensions.

To address the limitations of existing models in content-centric tasks, we propose a novel Self-Supervised Contrastive Representation Learning (SSCRL) framework for disentangling content-specific features from speech signals. Our method employs randomized pitch perturbation and speaker identity obfuscation on speech data, which are then fed into a pre-trained model and further fine-tuned using our designed self-supervised contrastive learning framework on the LibriSpeech 100-hour dataset. By explicitly enforcing semantic proximity between semantically equivalent features within the feature space, this approach effectively decouples linguistic content from paralinguistic attributes (e.g., vocal pitch and speaker characteristics), thereby enhancing the semantic coherence of learned representations. The method achieves this while maintaining low computational costs and minimal data requirements, narrowing the distance between semantically similar content in the feature space. Additionally, SSCRL demonstrates robustness in handling variations in pitch and speaker identity, offering an efficient and scalable solution for advancing content-focused speech processing.

Experiments on the SUPERB benchmark (Yang et al., 2021) demonstrate that SSCRL achieves competitive performance in content-related tasks such as Automatic Speech Recognition (ASR), Phoneme Recognition (PR), Query-by-Example Spoken Term Detection (QbE), Keyword Spotting (KS), Intent Classification (IC) and Slot Filling (SF). The main contributions of this work are as follows:

- We propose SSCRL, a low-cost yet effective method that achieves disentanglement of speech content and paralinguistic information within self-supervised fine-tuning framework.
- The SSCRL exhibits strong performance on the TIMIT (Hinton, 2012) and LibriSpeech datasets, demonstrating that the SSCRL method has robustness and generalization capabilities across different datasets.

2 Methods

Our approach draws inspiration from methods in image processing (Pan et al., 2023), where supervised contrastive learning was employed to fine-tune pre-trained models using labeled data, thereby achieving superior performance on downstream tasks. However, our method diverges from that of Pan et al. (Pan et al., 2023) in that we fine-tune the pre-trained model through self-supervised contrastive learning (Gunel et al., 2020; Zhang et al., 2021). During this fine-tuning phase, we do not use labeled data but instead learn content-related speech representations solely from the relationships between perturbed speech samples.

Our method is illustrated in Figure 1. Initially, our method employs tone-perturbed speech and speaker-perturbed speech as inputs to the pre-trained models. For the tone-perturbed speech, we adopted the method from ASR (Ko et al., 2015), implementing it using the SpeedPerturbation and PitchShift functions from the torchaudio.transforms library in Torchaudio (Yang et al., 2022). For the speaker-perturbed speech,

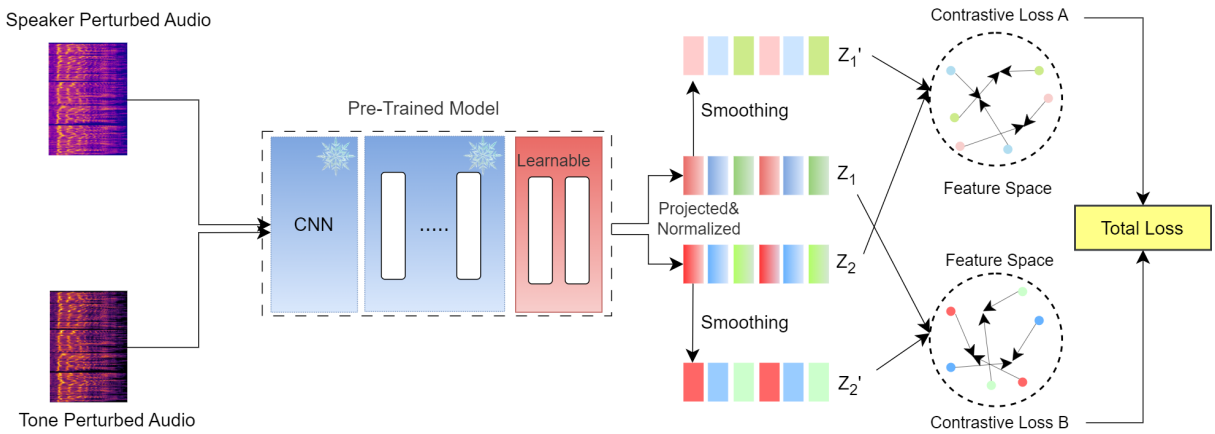

Figure 1: Architecture of the Self-Supervised Contrastive Representation Learning Fine-Tuning Method.

we used algorithms from (Choi et al., 2021; Eide and Gish, 1996) that alter speaker-related information while preserving more of the content-related information in the speech.

We paired the speaker-perturbed utterances and tone-perturbed utterances derived from the same source audio and fed them into a pre-trained model. Simultaneously, we extracted representations for both the speaker-perturbed utterances and the tone-perturbed utterances. These representations were then normalized and passed through a linear projection to obtain the final speaker-perturbed utterance representation Z_1 and tone-perturbed utterance representation Z_2, each with dimensions (batch size $\times$ seq length, D), D is the linear projection dimension. The two types of speech representations, Z_1 and Z_2, are subjected to unsupervised contrastive learning in a cross-manner, using this as a loss function to fine-tune the pre-trained model. During training, we found that directly using representations Z_1 and Z_2 can lead to unstable model training. Therefore, we adopted the Sinkhorn-Knopp algorithm proposed in (Cuturi, 2013), which normalizes a matrix into a probability distribution matrix that satisfies specific marginal constraints. This algorithm helps balance the similarity between samples when calculating the contrastive loss, making representations more effective for learning. Thus, after applying the Sinkhorn-Knopp algorithm for row and column normalization to Z_1 and Z_2, we obtain Z_1' and Z_2', which are then used in a cross-manner for self-supervised learning. The algorithm is shown in Equation (1).

$$Z' = \text{diag}(u^{(k)})\, Z\, \text{diag}(v^{(k)}) \tag{1}$$

Let $Z \in \mathbb{R}^{m \times n}$, where $u^{(k)}$ and $v^{(k)}$ are hyperparameters representing the row and column scaling factors at the k-th iteration, respectively.The function $\text{diag}(\cdot)$ constructs a diagonal matrix from a vector, with the vector elements on the diagonal and zeros elsewhere. They are typically initialized as $u^{(0)} = \mathbf{1}_m$ and $v^{(0)} = \mathbf{1}_n$, where $\mathbf{1}_m$ and $\mathbf{1}_n$ denote vectors of ones with dimensions m and n. The iterative formulas for $u^{(k)}$ and $v^{(k)}$ are shown in Equations (2) and Equations (3):

$$u^{(k+1)} = \frac{Q^{(k)} v^{(k)}}{\sum_{i=1}^{m} \left(Q^{(k)} v^{(k)}\right)_i} \tag{2}$$

$$v^{(k+1)} = \frac{Q^{(k)\top} u^{(k+1)}}{\sum_{i=1}^{n} \left(Q^{(k)\top} u^{(k+1)}\right)_i} \tag{3}$$

In the 0-th iteration, Q needs to be initialized as shown in Equation (4):

$$Q^{(0)} = \frac{1}{\epsilon} \exp\left(\frac{Z}{\epsilon}\right) \tag{4}$$

Here, ϵ is a hyperparameter used to control the smoothness of the matrix. Z' represents the final result after normalization by the algorithm. The algorithm yields Z'_1 and Z'_2. The final contrastive loss function is shown in Equation (5):

$$\mathcal{L}_{total} = \frac{1}{2}\left(\mathcal{L}_A + \mathcal{L}_B\right) \tag{5}$$

The formula for $\mathcal{L}_A$ is shown in Equation (6):

$$\mathcal{L}_A = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{\exp\left(\frac{Z_i \cdot Z'_i}{\tau}\right)}{\sum_{j\neq i}\exp\left(\frac{Z_i \cdot Z'_j}{\tau}\right)}\right) \tag{6}$$

Here, τ is the temperature parameter used to control the smoothness of the distribution. N is the first dimension of Z. Z' is the normalized matrix obtained through the Sinkhorn algorithm. Similarly, the formula for $\mathcal{L}_B$ can be derived similarly as shown in Equation (6).

3 Experiments

3.1 Configuration and Parameters

Pre-trained Models: We used the base models of WavLM (Chen et al., 2022) and HuBERT (Hsu et al., 2021), fine-tuning only the last two layers while keeping others frozen.

Experimental Data: In our experiments, we fine-tuned the pretrained models on the train-clean-100 hour LibriSpeech dataset (Panayotov et al., 2015), employing the dev-clean and dev-other datasets for validation. For testing purposes within the SUPERB Benchmark framework, we utilized the test-clean dataset for Automatic Speech Recognition (ASR) and Phoneme Recognition (PR) tasks. For Query-by-Example Spoken Term Detection (QbE), Keyword Spotting (KS), Intent Classification (IC), Slot Filling (SF), and Speaker Identification (SID) tasks, we used the QUESST14 (Anguera et al., 2015), Speech Commands (Warden, 2018), Fluent Commands (Lugosch et al., 2019), SNIPS (Coucke et al., 2018), and VoxCeleb1 (Nagrani et al., 2017) datasets, respectively. Our findings align with those obtained by methods such as SCORE, where increasing the LibriSpeech dataset (Panayotov et al., 2015) to 360 hours for model fine-tuning did not yield performance improvements. Furthermore, we incorporated the TIMIT (Hinton, 2012) dataset into the S3PRL framework to evaluate the effectiveness of our approach in ASR and PR tasks when there is a mismatch between the training and target datasets.

Experimental Setup: The learning rate was linearly increased from 0 to 10^{-4} over the first 2500 steps and then decayed to 10^{-6} over the next 2500 steps, for a total of 5000 fine-tuning steps. The Sinkhorn algorithm used $k = 3$ iterations and a smoothing parameter $\epsilon = 0.02$. Gradient clipping was set to 3, and the embedding dimension D was fixed at 256. For contrastive loss, we used a temperature hyperparameter $\tau = 0.1$ and applied PyTorch's clamp function (Imambi et al., 2021) to ensure training stability. All experiments were performed on a single RTX3090 GPU, and the fine-tuning process took approximately 3 hours. The model checkpoint at step 5000 was selected for downstream task fine-tuning within S3PRL.

3.2 Performance Evaluation on SUPERB Benchmark

All our downstream tasks were tested using the open source S3PRL toolkit (S3PRL). S3PRL (Self-Supervised Speech Pre-training and Representation Learning Toolkit) is a toolkit for speech processing that focuses on self-supervised learning methods. It provides a variety of pre-trained models and evaluation frameworks. Within this framework, we report the results for ASR, PR, QbE, KS, IC, and SF. The results are shown in Table 1.

From Table 1, we can clearly observe that our method excels in three metrics for both the IC and SF tasks on both the WavLM and HuBERT models, showing a significant advantage over other methods. In the ASR task, our method achieves a word error rate (WER) of 5.78% on the WavLM model, placing second only to ContentVec$_{500}$. In the HuBERT model, it achieves a WER of 6.16%, outperforming all other methods. For the QbE task, our method surpasses all other models in the WavLM model with a

Table 1: Performance Metrics on the SUPERB Benchmark for ASR, PR, QbE, KS, IC, and SF Tasks Using the LibriSpeech Dataset. The metrics include Accuracy (Acc%), Phoneme Error Rate (PER%), Word Error Rate (WER%), Maximum Term Weighted Value (MTWV%), F1 Score, and Concept Error Rate (CER%). The boldface represents the SOTA performance, while the underline denotes the top-3 performance.

Index	Method	ASR	PR	QbE	KS	IC	SF	
		WER↓	PER↓	MTWV↑	ACC↑	ACC↑	F1↑	CER↓
#01	FBANK	23.18	82.01	0.58	41.38	9.65	69.64	52.98
#02	HuBERT	6.42	5.41	7.36	96.30	98.34	88.53	25.20
#03	WavLM	6.21	4.84	8.70	**96.79**	98.63	**89.38**	22.86
#04	Wav2vec2.0	6.43	5.74	2.33	96.23	92.35	88.30	25.27
#05	data2vec	4.94	4.69	5.76	96.56	97.63	88.59	24.77
#06	ContentVec$_{500}$	**5.70**	4.54	5.90	96.40	99.10	89.60	23.60
#07	HuBERT + Spin$_{256}$	6.34	4.39	9.12	96.53	98.34	89.00	24.32
#08	WavLM + Spin$_{256}$	5.88	**4.18**	8.79	96.20	98.52	88.84	24.06
#09	HuBERT + SCORE	6.35	4.84	8.10	96.04	96.78	85.95	29.47
#10	WavLM + SCORE	6.15	4.72	9.18	96.29	97.86	88.63	25.10
#11	HuBERT + LASER	6.18	4.61	8.91	95.84	98.62	86.09	28.68
#12	WavLM + LASER	5.92	4.28	9.27	95.74	98.99	87.77	26.19
#13	HuBERT + SSCRL	6.16	4.83	8.73	96.65	**99.23**	89.24	23.28
#14	WavLM + SSCRL	5.78	4.61	**9.67**	96.13	99.15	88.90	**22.85**

#01−**#08** Reported in (Chang et al., 2023; Baevski et al., 2020; Baevski et al., 2022). **#09**−**#10** Reported in (Meghanani and Hain, 2024b). **#11**−**#12** Reported in (Meghanani and Hain, 2024a).
We re-implemented the KS, IC, and SF metrics for models **#09**−**#12** to ensure a fair comparison.

result of 9.67%. In the KS and PR tasks, our method also achieves favorable results compared to other approaches.

In general, our method achieves efficient and superior performance using just 100 hours of training data, demonstrating its effectiveness and robustness in content-related tasks.

3.3 Evaluating Generalization on Out-of-Domain Data

To evaluate the generalization capability of SSCRL across different datasets, we conducted ASR and PR experiments on the TIMIT dataset using WavLM and HuBERT models fine-tuned on LibriSpeech data. Notably, no fine-tuning was performed on the TIMIT dataset. For the ASR task, we trained for 100,000 steps, while for the PR task, we trained for 50,000 steps. Performance for both tasks was evaluated based on the final layer. The results are shown in Table 2.

From Table 2, we can see that our method achieves WER of 27.38% and 24.66% on the ASR task using the HuBERT and WavLM models, respectively. This places our method second only to the ContentVec$_{500}$ model but surpasses all other methods. For the PR task, our performance with the HuBERT model is second only to HuBERT + Spin$_{256}$. On the WavLM model, our results are slightly behind only the WavLM + SCORE approach. These findings highlight that SSCRL effectively extracts domain-invariant, content-focused representations, achieving consistent performance across diverse datasets with only 100

Table 2: Performance Metrics for PR and ASR on the TIMIT Dataset. The underline denotes the boldface represents the SOTA performance, while the top-2 performance.

Index	Method	FT Data Hours	ASR	PR
			WER↓	PER↓
#01	HuBERT	0	31.49	14.61
#02	WavLM	0	28.31	14.63
#03	ContentVec$_{500}$	76K	**20.49**	15.63
#04	HuBERT + Spin$_{256}$	356	28.93	**14.1**
#05	WavLM + Spin$_{256}$	356	26.56	14.61
#06	HuBERT + SCORE	100	31.08	14.44
#07	WavLM + SCORE	100	28.53	14.31
#08	HuBERT + LASER	100	30.31	14.76
#09	WavLM + LASER	100	28.53	14.71
#10	HuBERT + SSCRL	100	27.38	<u>14.2</u>
#11	WavLM + SSCRL	100	<u>24.66</u>	14.59

#01−**#11** were implemented using the S3PRL toolkit.

hours data. The ability to maintain strong results on both ASR and PR tasks under varying domain distributions further underscores SSCRL's robustness and adaptability.

3.4 Ablation Implementation For Disturbed Audio

We also verified the impacts of two different perturbed audios on model fine-tuning respectively.

Table 3: Performance Metrics for QbE, PR and ASR across various input speech conditions. TP denotes tone-perturbed speech, while SP denotes speaker-perturbed speech.

Model	Augmentation	QbE	ASR	PR
		MTWV↑	WER↓	PER↓
HuBERT	SP+TP	8.73	6.16	4.83
	-SP	8.71	6.31	4.85
	-TP	8.44	6.37	4.89
WavLM	SP+TP	9.67	5.78	4.61
	-SP	9.69	5.92	4.69
	-TP	9.73	5.99	4.82

As can be seen from Table 3, through comparative testing, it can be found that both the HuBERT and WavLM models are significantly affected by tone-perturbed, especially in ASR and PR tasks, which highlights the importance of prosodic information. Despite facing similar challenges, WavLM demonstrates stronger robustness and better adaptability to tone changes.

3.5 Discrete Unit Quality Evaluation

Table 4: Discrete unit quality evaluation. Cls Pur, Phn Pur, and PNMI denote cluster purity, phone purity, and phone-normalized mutual information.The "L" refers to the optimal layer based on the metrics reported by the model. The underline denotes the top-2 performance, while the boldface represents the SOTA performance.

Index	Method	L	Cls Pur↑	Phn Pur↑	PNMI↑
#01	HuBERT	7	0.154	0.639	0.630
#02	WavLM	11	**0.178**	0.624	0.640
#03	Data2vec	4	0.173	0.652	0.630
#04	ContentVec$_{100}$	12	0.169	0.650	0.643
#05	ContentVec$_{500}$	8	0.154	0.639	0.629
#06	HuBERT + Spin$_{256}$	12	0.150	0.641	0.655
#07	HuBERT + Spin$_{2048}$	12	0.151	0.654	0.666
#08	WavLM + Spin$_{256}$	12	0.137	0.644	0.658
#09	WavLM + Spin$_{2048}$	12	0.153	0.650	0.666
#10	HuBERT + SSCRL	12	0.158	0.652	0.674
#11	WavLM + SSCRL	12	0.148	**0.659**	**0.685**

#01−**#09** Reported in (Chang et al., 2023).

We evaluated the performance of the SSCRL method using three metrics: Clustering Purity(Cls Pur), Phoneme Purity(Phn Pur), and Phoneme Normalized Mutual Information(PNMI). These metrics, proposed in HuBERT, aim to assess the quality of discrete units after model discretization. Clustering Purity measures the extent to which each clustering result is predominantly composed of samples with a single label. Phoneme Purity evaluates how well phoneme information is retained in audio representations after discretization or clustering. Phoneme Normalized Mutual Information(PNMI) quantifies the overall alignment between clustering results and phoneme categories.

The results for all three metrics are presented for the best-performing layer. We randomly selected 10 hours of data from the train-clean-100 subset of the LibriSpeech dataset and fine-tuned our models using this subset. We then trained a K-Means clustering model with 256 clusters on the fine-tuned representations and calculated these three metrics on the dev-clean and dev-other datasets using the K-Means Clustering algorithm (Kodinariya et al., 2013).

As shown in Table 4, the SSCRL method achieves higher PNMI scores than other models on both our WavLM and HuBERT models. For the Phn Pur metric, our method outperforms other models on the WavLM model and is second only to the HuBERT + Spin2048 model on the HuBERT model. For the Cls Pur metric, our method is competitive with the Spin model. These results clearly demonstrate that our method effectively enhances the quality of discrete units.

3.6 Visualization of Perturbed Audio Features

We randomly selected an audio clip from the dev-other dataset, applied pitch perturbation and speaker perturbation to it, then extracted original and perturbed audio features using both the pre-fine-tuned and post-fine-tuning WavLM-base models. Finally, we mapped these features onto a 2D plane using Principal Component Analysis (PCA) (Greenacre et al., 2022) dimensionality reduction algorithm, with the results shown in Figure 2.

Figure 2(a) shows the two types of audio features extracted by our pre-fine-tuned WavLM-base model

Figure 2: PCA visualization of raw and perturbed speech representations in 2-Dimensional space

from this audio clip, while Figure 2(b) presents the two audio features extracted by our fine-tuned model. The figure clearly demonstrates that before fine-tuning, the original audio features and speaker perturbed features exhibited noticeable uneven distribution in the feature space, indicating the original model's difficulty in distinguishing pitch perturbation and speaker perturbation features without fine-tuning. After fine-tuning with our method, the perturbed audio features show uniformly distributed and aligned patterns compared to the original speech features in the feature space, further demonstrating that the fine-tuned model exhibits enhanced robustness against perturbed audio while maintaining discriminative power for speech characteristics. This disentanglement in the feature space suggests that the model can effectively separate domain-specific perturbations from invariant speech representations.

3.7 Visualization of Features Aligned with Phonemes

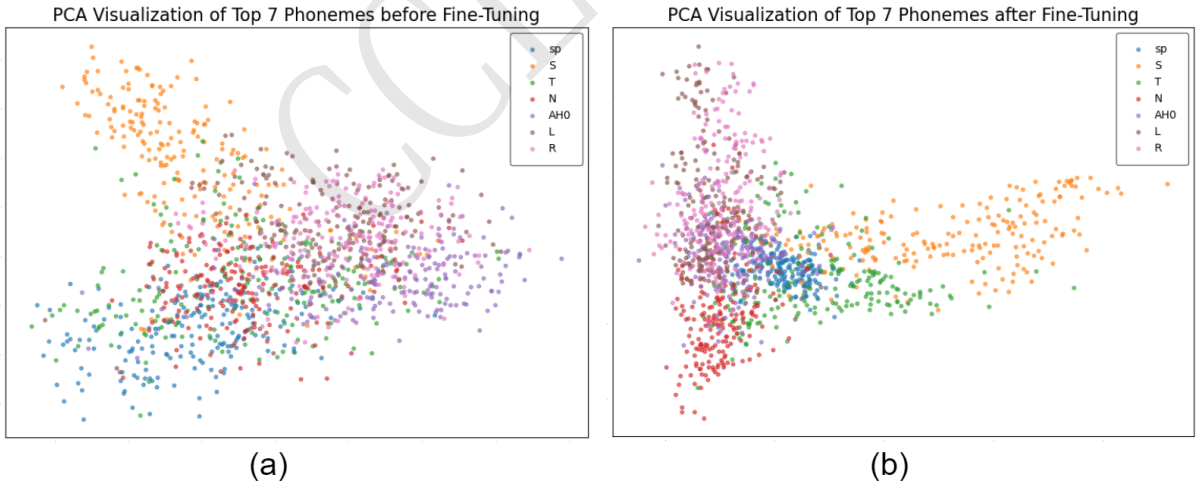

Figure 3: PCA visualization of speech representations in 2-Dimensional space

To demonstrate the efficacy of our method in feature space, we separately extracted representations using both the pre-finetuned and post-finetuned WavLM-Base models. These representations were obtained from the final layer of each model, subsequently processed through Principal Component Analysis

(PCA) (Greenacre et al., 2022) for dimensionality reduction, and finally mapped to a two-dimensional space with phoneme labels. The comparative analysis is presented in Figure 3.

As illustrated in Figure 3, panel (a) presents the representations extracted by the pre-finetuning WavLM-Base model, while panel (b) displays those obtained from the post-finetuning counterpart. Notably, following the application of our fine-tuning methodology, homogeneous phonemes demonstrate closer proximity within the feature space, exhibiting tighter clustering patterns. In contrast, the non-fine-tuned WavLM model reveals comparatively scattered phonemic representations with more disorganized spatial distributions.

3.8 Speaker Identification Accuracy

To validate the capability of our method in disentangling speaker information representation, we conducted an in-depth study on the speaker recognition task to assess the effectiveness of the SSCRL method in decoupling speaker information from speech content representations. In our experiments, consistent with the Spin approach, we utilized the last six layers of the HuBERT model to analyze speaker recognition performance. To ensure a fair comparison, we matched the number of training steps used in Spin, training for 50,000 steps.

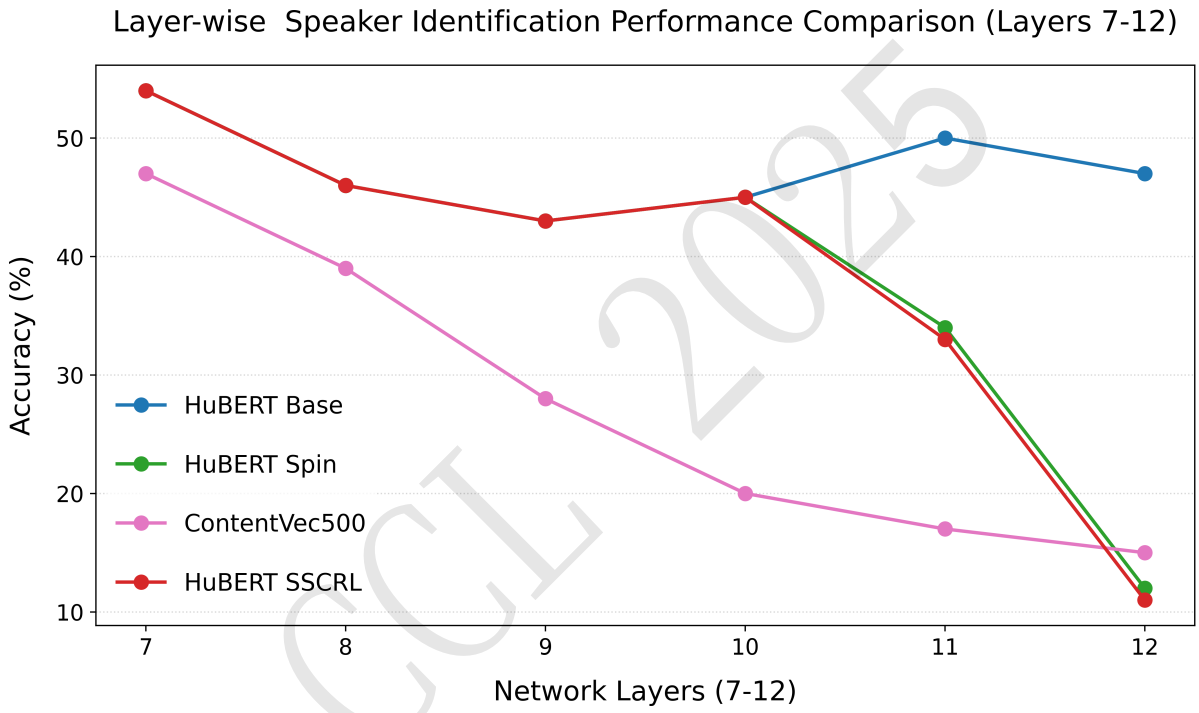

Figure 4: Speaker Identification accuracy analysis on HuBERT-base model

As clearly demonstrated in Fig. 4, after fine-tuning with our SSCRL method, the recognition accuracy of HuBERT's final two layers drops to 11%, outperforming ContentVec500 while achieving comparable results to Spin methods. This outcome evidences that SSCRL effectively reduces speaker information and tone information interference in speech representations, demonstrating its effectiveness in content representation disentanglement. Notably, these results are achieved using only the 100-hour LibriSpeech dataset.

4 Conclusion

This paper introduces Self-Supervised Contrastive Representation Learning (SSCRL), a fine-tuning approach that disentangles speaker information from speech content, improving semantic consistency in content-centric tasks. SSCRL achieves competitive performance across multiple benchmarks, including

lower WER and enhanced Intent Classification and Slot Filling accuracy, while using only 100 hours of training data. Its strong generalization across diverse datasets such as TIMIT further highlights its robustness. Additionally, SSCRL effectively clusters semantically similar features, validating its ability to enhance content-specific representations.

For future work, we plan to explore advanced self-supervised fine-tuning techniques to enhance the adaptability and performance of pre-trained models on diverse downstream tasks, pushing the boundaries of self-supervised learning in ASR and related applications.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grants: 62466030, U24A20334, 62376111), Yunnan Provincial Key R&D Program (202303AP140008, 202502AD080014), and the Open Fund of Yunnan Key Laboratory of Artificial Intelligence (CB24069D018A).

References

Xavier Anguera, Luis-J Rodriguez-Fuentes, Andi Buzo, Florian Metze, Igor Szöke, and Mikel Penagarikano. 2015. Quesst2014: Evaluating query-by-example speech search in a zero-resource setting with real-life queries. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5833–5837. IEEE.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning*, pages 1298–1312. PMLR.

David M Chan and Shalini Ghosh. 2022. Content-context factorized representations for automated speech recognition. *arXiv preprint arXiv:2205.09872*.

Xuankai Chang, Takashi Maekaku, Pengcheng Guo, Jing Shi, Yen-Ju Lu, Aswin Shanmugam Subramanian, Tianzi Wang, Shu-wen Yang, Yu Tsao, Hung-yi Lee, et al. 2021. An exploration of self-supervised pretrained representations for end-to-end speech recognition. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 228–235. IEEE.

Heng-Jui Chang, Alexander H Liu, and James Glass. 2023. Self-supervised fine-tuning for improved content representations by speaker-invariant clustering. *arXiv preprint arXiv:2305.11072*.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Hyeong-Seok Choi, Juheon Lee, Wansoo Kim, Jie Lee, Hoon Heo, and Kyogu Lee. 2021. Neural analysis and synthesis: Reconstructing speech from self-supervised representations. *Advances in Neural Information Processing Systems*, 34:16251–16265.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Ellen Eide and Herbert Gish. 1996. A parametric approach to vocal tract length normalization. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 1, pages 346–348. IEEE.

Solène Evain, Ha Nguyen, Hang Le, Marcely Zanon Boito, Salima Mdhaffar, Sina Alisamir, Ziyi Tong, Natalia Tomashenko, Marco Dinarelli, Titouan Parcollet, et al. 2021. Lebenchmark: A reproducible framework for assessing self-supervised representation learning from speech. *arXiv preprint arXiv:2104.11462*.

Michael Greenacre, Patrick JF Groenen, Trevor Hastie, Alfonso Iodice d'Enza, Angelos Markos, and Elena Tuzhilina. 2022. Principal component analysis. *Nature Reviews Methods Primers*, 2(1):100.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

GE Hinton. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.

Sagar Imambi, Kolla Bhanu Prakash, and GR Kanagachidambaresan. 2021. Pytorch. *Programming with TensorFlow: solution for edge computing applications*, pages 87–104.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. Audio augmentation for speech recognition. In *Interspeech*, volume 2015, page 3586.

Trupti M Kodinariya, Prashant R Makwana, et al. 2013. Review on determining number of cluster in k-means clustering. *International Journal*, 1(6):90–95.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. 2019. Speech model pre-training for end-to-end spoken language understanding. *arXiv preprint arXiv:1904.03670*.

Amit Meghanani and Thomas Hain. 2024a. Laser: Learning by aligning self-supervised representations of speech for improving content-related tasks. *arXiv preprint arXiv:2406.09153*.

Amit Meghanani and Thomas Hain. 2024b. Score: Self-supervised correspondence fine-tuning for improved content representations. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12086–12090. IEEE.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. 2017. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*.

Haolin Pan, Yong Guo, Qinyi Deng, Haomin Yang, Jian Chen, and Yiqun Chen. 2023. Improving fine-tuning of self-supervised models with contrastive initialization. *Neural Networks*, 159:198–207.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.

Cal Peyser, Ronny Huang Andrew Rosenberg Tara N Sainath, Michael Picheny, and Kyunghyun Cho. 2022. Towards disentangled speech representations. *arXiv preprint arXiv:2208.13191*.

Kaizhi Qian, Yang Zhang, Heting Gao, Junrui Ni, Cheng-I Lai, David Cox, Mark Hasegawa-Johnson, and Shiyu Chang. 2022. Contentvec: An improved self-supervised speech representation by disentangling speakers. In *International Conference on Machine Learning*, pages 18003–18017. PMLR.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504.

P. Warden. 2018. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. *ArXiv e-prints*, April.

Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.

Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Artyom Astafurov, Caroline Chen, Christian Puhrsch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z Yang, et al. 2022. Torchaudio: Building blocks for audio and speech processing. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6982–6986. IEEE.

Yifan Zhang, Bryan Hooi, Dapeng Hu, Jian Liang, and Jiashi Feng. 2021. Unleashing the power of contrastive self-supervised visual models via contrast-regularized fine-tuning. *Advances in Neural Information Processing Systems*, 34:29848–29860.