QUANTIFYING SEPARATION IN REPRESENTATIONS FOR SPEECH TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised speech modelling aims to learn useful general-purpose speech representations for a wide variety of downstream tasks. Despite their superior capability to capture complex data properties, one of the drawbacks is that information relevant for different tasks is typically entangled in the representation space, thereby limiting general task performance and imposing extra difficulty for universal speech representation learning. A formal measure of the degree of separation of task-relevant information in representations has the potential to mitigate this but is under-investigated in the literature. To address this gap, this work introduces the concept of Task Separation, which extends the concept of disentanglement from latent factors to downstream task use. It aims to quantify the degree of separation of representation content between tasks. Using Self-supervised Learning (SSL) features linear models are added to probing tasks. An information-theoretic measure of Task Separation is derived from the parameter space of such linear probers. Experiments examine the degree of separation between four speech tasks and different SSL presentations: speaker identification (SID), speech emotion recognition (SER), phone recognition (PR) and automatic speech recognition (ASR). Experiments show that: 1) the separation score for tasks between SID and SER, PR and ASR, respectively, is at a moderate level, ranging from 0.32 to 0.67, suggesting the difficulty of separating task-relevant information of SID from the other three tasks; 2) with a relatively high separation score, the linear model parameters from SER, PR and ASR can filter out the relevant information for SID effectively; 3) the distribution derived from the parameter space of linear probers can be used to interpret the optimal design of a minimum hierarchical model to combine two speech tasks. These findings enrich the scientific understanding of speech representations from the view of downstream tasks, with a formal evaluation protocol contributing to the goal of universal speech representation learning.

1 INTRODUCTION

Learning general-purpose speech representations that are useful for a wide variety of downstream tasks has been one of the long-standing goals in the modelling of speech, with the latest advancement of Self-Supervised Learning (SSL) methods (Baevski et al., 2020; Hsu et al., 2021; Chen et al., 2022; Liu et al., 2023; 2025). Despite the promising performance of self-supervised modelling of speech, most methods are trained without explicit separation of task-relevant information in the representation space, thereby potentially limiting certain task performance. For instance, recent works (Pasad et al., 2023; 2024) on the layerwise analysis of speech SSL models have found that different speech properties are centred at distinct layers; a more recent study on the comprehensive layerwise evaluation of speech SSL models has confirmed that different layer representations are beneficial for distinct tasks (Yang et al., 2024) and the authors further suggested the potential to improve all tasks by separating the speech properties in distinct layers. Presumably, a hierarchical model can achieve this purpose; however, it remains a challenge in practice, and most methods are limited to simple datasets(Hsu et al., 2017; Yingzhen & Mandt, 2018; Barami et al., 2026).

In representation learning, real-world data is usually assumed to be generated by mixing multiple latent factors. Learning to separate or disentangle these latent factors has been one of the fundamental problems in this field (Bengio et al., 2013). The disentangled representations are believed to be useful for the downstream tasks of interest (Locatello et al., 2019; Bengio et al., 2013). However,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

one of the primary issues for disentanglement is that this important concept lacks a clear definition (Higgins et al., 2018), resulting in the scientific measure of success being difficult. For instance, one of the classic views on disentanglement assumes the latent factors behind data variation are statistically independent(Eastwood & Williams, 2018; Ridgeway & Mozer, 2018; Bengio et al., 2013), and the objective is to recover them using learning models (Higgins et al., 2017; Kim & Mnih, 2018; Locatello et al., 2019). In addition to the latent factor view, the independence assumption has been made from other perspectives to address disentanglement (Hyvärinen & Hoyer, 2000; Gresele et al., 2021; Matthes et al., 2026). The lack of a well-accepted definition imposes extra difficulty for the evaluation of disentanglement, and the standard measures so far are mainly based on the statistical independence assumption among latent factors on simple datasets (Eastwood & Williams, 2018; Ridgeway & Mozer, 2018). However, in many real-world datasets, such as speech, the latent factors are usually unknown, let alone how to measure the separation.

To address the above gap, this work aims to quantify the degree of separation of relevant information for speech tasks in the representations. The concept of Task Separation, short for *the separation of relevant information of speech tasks*, is introduced, which extends the classic view of disentanglement from the latent-factor level to downstream tasks. By employing linear models to solve the probing tasks, an information-theoretical measure of Task Separation is derived from the parameter space of the linear probers. Such a measure was examined on four commonly used speech tasks, including Speaker Identification (SID), Speech Emotion Recognition (SER), Phone Recognition (PR) and Automatic Speech Recognition (ASR), with input features from two representative but distinct speech SSL models of wav2vec 2.0 (Baevski et al., 2020) and (Chen et al., 2022). The metric of Task Separation is further investigated in multi-stage training and joint optimisation of hierarchical models, respectively. The main contributions of this work are:

- The introduction of Task Separation, an alternative view to disentanglement at the task level with an information-theoretical measure (Section 3) and comprehensive investigation on four commonly used speech tasks and four SSL features, all based on real-world datasets (Section 4.2);
- Empirical evidence enriches the scientific understanding of speech representations from the perspective of downstream tasks, where the proposed metric can inform the design of hierarchical speech models for different tasks, contributing to the goal of universal speech representation learning with a formal evaluation protocol (Section 4.3).

2 Related work

2.1 Self-supervised speech modelling

Recent works on the modelling of speech have been revolutionised by SSL methods, due to the outstanding capabilities across a wide variety of downstream tasks(Yang et al., 2021). In contrast to supervised counterparts, SSL models make use of supervision from the data itself rather than human-annotated labels, and they are typically trained to solve a pre-text task (Mohamed et al., 2022), such as reconstruction (Van Den Oord et al., 2017; Huang et al., 2022), contrastive learning (Oord et al., 2018; Baevski et al., 2020) and masked prediction (Hsu et al., 2021; Chen et al., 2022). Pioneering models like wav2vec 2.0 (Baevski et al., 2020) and HuBERT(Hsu et al., 2021) or its successor WavLM (Chen et al., 2022) are optimised with distinct pre-text tasks, and both have demonstrated excellent performance; very recent works have further enhanced the techniques for model training (Liu et al., 2023; 2025). To pursue the goal of universal speech representations, in contrast to existing studies on learning methods, this work presents a comprehensive analysis of relevant information in the representations across speech tasks, investigating speech representations from the view of downstream tasks, following (Peng et al., 2026).

2.2 Disentangled representation learning

Learning to disentangle or recover the latent factors behind data variation has been one of the fundamental problems of representation learning, and the representations after disentanglement are usually assumed to be useful for related tasks (Bengio et al., 2013; Locatello et al., 2019). Disentanglement has been extensively addressed in the problem of identifiability. To obtain an identifiable model, independence assumptions have been made in the literature. For instance, non-linear

Independent Component Analysis (ICA) (Hyvarinen et al., 2019; Hyvarinen & Morioka, 2016) assume that the latent factors are statistically independent, and the models become identifiable if they are able to separate these factors. However, this statistical independence assumption is too strong for many real-world data, which has inspired other works to relax it. Instead of latent factors, Independent Subspace Analysis (ISA)(Hyvärinen & Hoyer, 2000) assumes the existence of subspaces for the representations and imposes a statistical independence assumption at the subspace level. Recently, Independent Mechanistic Analysis (IMA)(Gresele et al., 2021; Matthes et al., 2026) was proposed with the mechanistic independence assumption for the mixing function. Both ISA and IMA follow the same spirit to relax the independence assumption at the latent factor level. Different from the above existing works, this paper presents an alternative view on disentanglement from the downstream task level, with an information-theoretical measure derived in Section 3.2 and empirical evidence for its usefulness in Section 4.3.

Metrics. For the quantitative evaluation of disentangled representations it is typically assumed that the latent factors are statistically independent (Eastwood & Williams, 2018; Ridgeway & Mozer, 2018). Under this assumption different metrics have been proposed, such as intervention-based(Higgins et al., 2017; Kim & Mnih, 2018), predictor-based(Eastwood & Williams, 2018; Ridgeway & Mozer, 2018) and information-based(Chen et al., 2018; Sepliarskaia et al., 2019), according to the taxonomy in (Carbonneau et al., 2022). Very recent work extends the evaluation for sequential data (Barami et al., 2026). Instead of a better metric this work aims to examine speech representation from the view of downstream tasks, via a proxy measure for the separation of task-relevant information.

3 Quantifying separation in speech representations

This section presents the proposed approach to quantifying the separation in speech representations to select and combine the representations to construct the task-relevant information, i.e. to solve the optimisation task. At first we provide an overview of the classic view and measure for disentanglement of latent factors (Eastwood & Williams, 2018; Ridgeway & Mozer, 2018) in Section 3.1, followed by the proposed measure of Task Separation in Section 3.2. Under the novel alternative view of disentanglement at the task level, i.e. Task Separation in this work, one can quantify the relationship among speech tasks in the representations by examining the corresponding parameter space.

3.1 Measuring disentanglement of latent factors

Most real-world data is believed to be generated based on a mixture of multiple latent factors. One of the working assumptions is that the changes among them are independent (Bengio et al., 2013). Learning representations that can disentangle the latent factors has become one of the long-standing goals in representation learning. To measure the degree of disentanglement among the latent factors, one of the standard approaches employs supervised models to learn the mapping between latent factors and the learned representations (Eastwood & Williams, 2018; Ridgeway & Mozer, 2018).

Let $\boldsymbol{z}$ denote a speech frame representation extracted from some neural networks, e.g. SSL models, where $\boldsymbol{z} \in \mathbb{R}^D$. The traditional view of disentanglement assumes that $\boldsymbol{z}$ is generated from the result of mixing some latent factors $\boldsymbol{y}$, where $\boldsymbol{y} \in \mathbb{R}^K$ (Bengio et al., 2013). Under this view, a measure can be derived by using a function f with parameters $\boldsymbol{\theta}$ to predict $\boldsymbol{y}$ from $\boldsymbol{z}$, i.e. $\hat{\boldsymbol{y}} = f(\boldsymbol{\theta}; \boldsymbol{z})$, where $\boldsymbol{\theta}$ typically consists of weights $\boldsymbol{W}$ and biases $\boldsymbol{b}$, and $\boldsymbol{W} \in \mathbb{R}^{D\times K}, \boldsymbol{b} \in \mathbb{R}^K$ if the function f is linear. With discrete latent factors, the prediction of the target distribution $\hat{p}(\boldsymbol{y})$ can be obtained from the linear transformation with function g, i.e. $\hat{p}(\boldsymbol{y}) = g(\boldsymbol{W}^\top \boldsymbol{z} + \boldsymbol{b})$ (Ridgeway & Mozer, 2018). Note that $\boldsymbol{W}$ is obtained by concatenating the weight vector w $\in \mathbb{R}^D$ from K regressors in (Eastwood & Williams, 2018). In contrast to the one-versus-rest logistic regression used as g in (Ridgeway & Mozer, 2018), this work adopts the softmax function due to the objective of solving the task at this point, as will be covered in Section 3.2. Since $\boldsymbol{W}$ is learned by solving the prediction task, its elementwise magnitude therefore reflects the task-relevant information in the form of selecting and combining the representations $\boldsymbol{z}$, which is defined as

$$\boldsymbol{R} = \boldsymbol{W}^{\text{abs}}, \tag{1}$$

where $\boldsymbol{R} \in \mathbb{R}^{D \times K}$ is the relative importance matrix obtained by taking the elementwise absolute value of $\boldsymbol{W}$, and $\boldsymbol{R}_{i,j}$ indicates the importance of the i-th element $\boldsymbol{z}_i$ in predicting the j-th latent factor $\boldsymbol{y}_j$. The degree of the separation for the i-th representation $\boldsymbol{z}_i$ in predicting the K different latent factors (Eastwood & Williams, 2018) is defined as:

$$\mathrm{d}_i = (1 - H_K(\boldsymbol{P}_{i,:})), \tag{2}$$

where d_i is the factor-level disentanglement score for $\boldsymbol{z}_i$. $H_K(\boldsymbol{P}_{i,:})$ is the normalised entropy defined as $H_K(\boldsymbol{P}_{i,:}) = -\sum_{j=1}^{K} \boldsymbol{P}_{i,j} \log_K \boldsymbol{P}_{i,j}$ and $\boldsymbol{P}_{i,j} = \boldsymbol{R}_{i,j} / \sum_{k=1}^{K} \boldsymbol{R}_{i,k}$ is the probability of $\boldsymbol{z}_i$ being important to predict $\boldsymbol{y}_j$. As suggested by Eastwood & Williams (2018) the value of d_i will be close to 0 if the representation $\boldsymbol{z}_i$ is equally important to predict all K latent factors. In contrast the value of d_i will be 1 when $\boldsymbol{z}_i$ is exclusively important to predict a single latent factor. The total disentanglement score $d_{\text{disentanglement}}$ over the D dimensionalities is defined as

$$d_{\text{disentanglement}} = \sum_{i=1}^{D} \rho_i \mathrm{d}_i, \tag{3}$$

where $\rho_i = \sum_j \boldsymbol{R}_{i,j} / \sum_{i,j} \boldsymbol{R}_{i,j}$ is the weight for the representation $\boldsymbol{z}_i$. The weighted average can better account for the dimensions that are irrelevant in predicting $\boldsymbol{y}$, as the associated ρ_i tends to be close to 0 (Eastwood & Williams, 2018).

3.2 Measuring separation of tasks

One of the drawbacks of the above measure for disentanglement is the requirement for knowledge about the ground-truth latent factors behind the data variation. While these factors are available for synthetic data (Eastwood & Williams, 2018), they are usually unknown for most real-world data, particularly speech. Despite the limited knowledge about data generation, in many cases, one is more interested in the information to solve certain tasks rather than accounting for data variation. A very common way to extract task-relevant information from the representations is to employ a supervised model. With this intuition, the proposed concept of Task Separation, an alternative view of disentanglement from the factor-level to the downstream task level, can be formally quantified using parameterised models with supervised training, similar to that in (Eastwood & Williams, 2018; Ridgeway & Mozer, 2018).

To derive the measure for Task Separation we follow the steps to define factor-level disentanglement, as described in the above Section 3.1. It differs in the way it constructs the relative importance matrix $\boldsymbol{R}$. While Eastwood & Williams (2018); Ridgeway & Mozer (2018) employ supervised models to predict the K latent factors, this work utilises the supervised models to solve K downstream tasks. For a given task with target labels $\boldsymbol{y} \in \mathbb{R}^N$, and parameters $\boldsymbol{W} \in \mathbb{R}^{D \times N}, \boldsymbol{b} \in \mathbb{R}^N$, the relative importance vector $\boldsymbol{r}$ for $\boldsymbol{z}$ to solve the task is defined as an average of the magnitude over the target label space:

$$\boldsymbol{r} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{W}_{:,i}^{\text{abs}}, \tag{4}$$

where $\boldsymbol{r} \in \mathbb{R}^D$ and $\boldsymbol{W}_{:,i}^{\text{abs}}$ is the elementwise absolute value of i-th column of the weight matrix $\boldsymbol{W}$. Although different speech tasks are usually associated with target labels $\boldsymbol{y}$ with distinct dimensionality N, i.e. the number of predicted labels is usually different across tasks, the relative importance vector $\boldsymbol{r}$ has the same dimensionality conditioned on the same input representations. For K speech tasks, one can construct the relative importance matrix $\boldsymbol{R} \in \mathbb{R}^{D \times K}$ from K individual $\boldsymbol{r}$. Following the above definitions in Equations 2 and 3, the measure of Task Separation $\mathrm{d}_{\text{separation}}$ can be derived, which is omitted for simplicity.

In addition to the measure of disentanglement for different latent factors, the authors in (Eastwood & Williams, 2018) introduced two more metrics, including Completeness and Informativeness, together forming the Disentanglement-Completeness-Informativeness (DCI) framework. Since the metric of Completeness is argued to be less useful (Ridgeway & Mozer, 2018), its task-level extension is not included in this work. While the metric of Informativeness is measured by an average

of errors made from different functions f in (Eastwood & Williams, 2018), this work reports the performance per task to present a fine-grained analysis for tasks.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset. The IEMOCAP (Busso et al., 2008), an emotional English speech corpus with labels of speaker identity, emotion state, phone and word, is used for the training and evaluation of four speech tasks, including SID, SER, PR and ASR. The data splits for PR and ASR follow the five-fold cross-validation setup of SER in s3prl [1], with the removal of utterances containing non-words only. The resulting total number of utterances is 5499. Since the setup of SID requires the overlap of speakers among training and evaluation sets, the whole utterances are split into five folds per speaker. For all tasks per data split, 20% of the training set is randomly split as the dev set. To derive the separation score $d_{\text{separation}}$ defined in Section 3.2 from the above four tasks, it is necessary to ensure the probing models are conditioned on the same data. For a given task, a manual check confirms that the training set of the five folds covers the whole dataset. The weight matrix $\boldsymbol{W}$ per task is extracted by averaging the five models of the cross-validation setup.

Implementation. The implementations for model training are based on [2], with support for SID, PR and ASR on IEMOCAP. Two representative but distinct speech SSL models of WavLM [3] and wav2vec 2.0 [4] are adopted to extract features from the raw speech data. Previous studies have suggested that different speech properties are centred at distinct layers in SSL models (Pasad et al., 2023; 2024). To have a comprehensive investigation of the impact of input features, representations before the first and from the last Transformer layer, denoted as Layer 0 and 12 respectively, are utilised for each SSL model, resulting in four different SSL features as input. To mitigate the influence of magnitude on $d_{\text{separation}}$, the input features are standardised with the elementwise mean and standard deviation calculated from the training set as in (Eastwood & Williams, 2018). The learning rate for SID is set to 0.01, while 0.001 is used for SER, PR and ASR, and training finishes after 45000, 30000, 40000 and 40000, respectively. The batch size is set to 4 across all tasks. The number of target outputs for SID and SER is 10 and 4, respectively. For PR, the output targets consist of 48 phones and 1 extra blank token, following (Graves et al., 2006). Similarly, the model is trained to predict 26 letters, 1 blank token and 1 whitespace for ASR. The whitespace token is used as a separator for words in the model prediction and is removed in the performance evaluation. The performance of SID and SER is measured by classification accuracy, while Phone Error Rate (PER) is measured for PR, Character Error Rate (CER) and Word Error Rate (WER) for ASR, respectively.

4.2 MEASURING TASK LEVEL DISENTANGLEMENT

The above four tasks were trained and evaluated on the five-fold cross-validation sets, with four different SSL features as input per fold. The performance per task was an average on the five test sets. Please refer to Appendix A for the comprehensive results and further verification of the correctness of the task setup.

The separation score $d_{separation}$ is calculated on the combinations of the four tasks above. The results of the combination between two tasks (denoted with "-" in between) are shown in Table 1, please refer to Appendix A for the numbers of more tasks.

The difficulty in separation. It can be seen from Table 1 that different combinations of tasks result in distinct degrees of separation. Overall, SID achieves a moderate separation score with SER, PR and ASR, respectively, ranging from 0.32 to 0.67 across different SSL features. This suggests the existence of overlap between SID and the three tasks in selecting and combining the representations to encode task-relevant information, and furthermore, the difficulty in separating them. The pairwise

[1] https://github.com/s3prl/s3prl/tree/main/s3prl/downstream/emotion

[2] https://github.com/s3prl/s3prl

[3] https://huggingface.co/s3prl/converted_ckpts/resolve/main/wavlm_base.pt

[4] https://huggingface.co/s3prl/converted_ckpts/resolve/main/wav2vec_small.pt

Table 1: Measure of separation $d_{separation}$ between two tasks with different SSL features as input.

SSL model	Layer	SID-SER	SID-PR	SID-ASR	SER-PR	SER-ASR	PR-ASR
WavLM	12	0.32	0.34	0.37	0.04	0.05	0.01
WavLM	0	0.50	0.63	0.67	0.12	0.14	0.02
wav2vec 2.0	12	0.46	0.36	0.41	0.06	0.05	0.01
wav2vec 2.0	0	0.37	0.60	0.58	0.16	0.14	0.01

combinations of SER, PR and ASR achieve a low score, which is consistent across different SSL features. This tendency suggests that while the selection and combination of the representations to solve the tasks of SER, PR and ASR is similar, SID manifests a distinct pattern from them. To compare how tasks utilise the representations, a histogram of the relative importance vector $\boldsymbol{r}$ defined in Equation 4 per task is illustrated in Figure 1. More illustrations can be found in Appendix C.

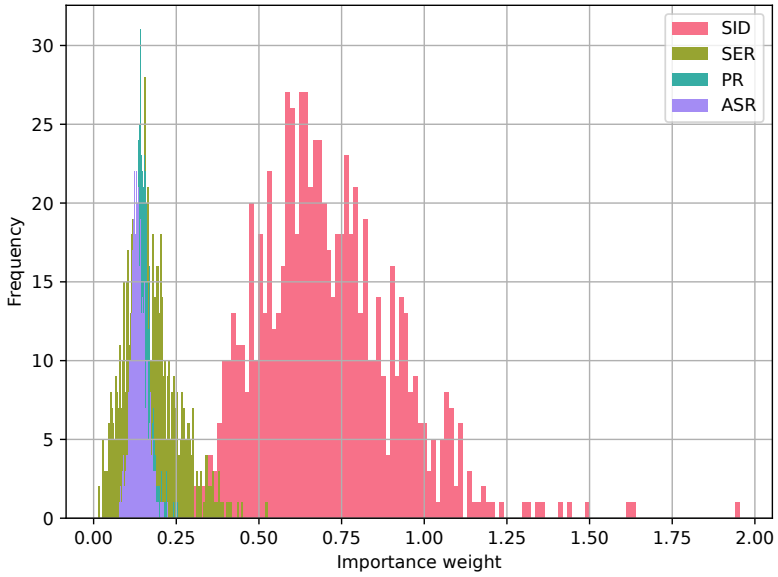


Figure 1: Histogram of the relative importance vector per task, with input features from the 12th layer of WavLM and 128 bins.

Distribution of the relative importance vector. With the input features from the 12th layer of WavLM, it can be seen from Figure 1 that the relative importance vector for SID has the least overlap with that from SER, PR and ASR. Furthermore, the overlap is easy to spot for SER, PR and ASR, where PR is mostly covered by ASR and both PR and ASR are mostly covered by SER. This is consistent with the pattern in Table 1.

4.3 Implications for Tasks

Based on the results in Section 4.2, some conjectures on the proposed metric can be made: 1) for tasks with a relatively high separation score, the parameter space from one task can function as a filter to remove the relevant information of the other task; 2) for tasks with the distribution of the relative importance vector covered by the other task, a hierarchical model can be constructed for joint optimisation. This subsection presents empirical evidence to validate the two hypotheses.

4.3.1 MULTI-STAGE TRAINING

To validate the first hypothesis, paired tasks were selected to investigate how one task performs by taking the output from the other task. Specifically, SID was trained on the output of the linear layer for SER, PR and ASR, denoted as SER_SID, PR_SID, and ASR_SID, respectively. Similarly, SER was trained on the output from SID, PR, and ASR; PR on ASR and ASR on PR. The outputs of SID and SER are excluded for PR and ASR, as the latter require frame-level representations rather than utterance-level vectors as input. Based on the results in Table 4, WavLM features were adopted to provide the initial model input, as they perform better than wav2vec 2.0 on most tasks; furthermore, the results for ASR in this and the next subsection are based on CER as WER may not be informative for showing the small difference. The relative improvement for the task taking the output from the other with SSL features per layer is calculated, which is shown in Table 2.

Table 2: Relative improvement (in %) for the multi-stage systems over their standalone counterparts.

Layer	SER_SID	PR_SID	ASR_SID	SID_SER	PR_SER	ASR_SER	ASR_PR	PR_ASR
12	-66.62	-43.44	-52.54	-40.57	-17.81	-22.72	-67.23	-45.65
0	-75.71	-28.77	-40.82	-42.71	-6.34	-10.41	-7.84	-8.40

Information filtering. It can be seen from Table 2 that the output from one task can act as a filter to remove the relevant information for the other task if their separation score is relatively high, as suggested by the significantly degraded performance of SER_SID, PR_SID, ASR_SID, SID_SER, on both layer representations of WavLM, respectively. For tasks with a low separation score, the performance degradation is relatively small, as suggested by the results of PR_SER and ASR_SER. It is interesting to note that ASR_PR and PR_ASR achieve a high degradation of -67.23% and -45.65% on Layer 12, respectively. One of the reasons may be that the last-layer representations from WavLM already perform well on the two tasks because, when using the representations from the first layer, the degradation is reduced to -7.84% and -8.40%, respectively.

4.3.2 JOINT OPTIMISATION

To validate the second hypothesis, paired tasks were selected to investigate the performance changes after joint optimisation of the two-level hierarchical model. Specifically, SID was trained jointly with SER, PR, and ASR, denoted as SER-SID, PR-SID and ASR-SID, respectively. Here, the notation SER-SID is distinguished from SID-SER, where the former represents a hierarchical model in which the low level is for the task of SER, and the output from its linear layer is concatenated with the SSL features as input for the high-level task of SID. For fair comparison, the hierarchical models consist of only two linear layers, which also function as a minimal sufficient hierarchical model. When constructing a hierarchical model like SER-PR, the output from SER was expanded to match the input length for PR. The resulting combinations consist of 12 hierarchical models. Note that for the models with the SID component, it requires speaker overlap among the data splits, but a caveat is that this is not the typical setup of SER, PR and ASR. The SSL features in this subsection are only based on the last layer of WavLM, as it performs well on most tasks, as shown in Table 4. The performance of SER, PR and ASR on the data splits with speaker overlap is shown in Table 6. The relative improvement is calculated on the performance of the hierarchical model with joint optimisation over its standalone components, which is shown in Table 3.

Hierarchical model design. Overall the results in Table 3 show that the joint optimisation of hierarchical models can bring positive improvement to tasks and, in some cases, to both tasks. It is interesting that the design of the hierarchy can impact the model performance. For example, in contrast to SID-SER, the design of SER-SID can improve both tasks. This may be explained by Figure 1 that the overlap between both tasks is mostly covered by the distribution of SID, which is consistent with existing works suggesting the inclusion of emotion information in speaker representations (Aldeneh & Provost, 2021; Ulgen et al., 2024). To combine utterance-level tasks like SID and SER with frame-level content tasks like PR and ASR, respectively, training the model with utterance-level tasks at the low level instead of the high level is generally better for both tasks, as suggested by the less degraded performance for PR and more improvement for ASR. This is consistent for SER, PR

Table 3: Relative improvement (in %) of joint optimisation over their standalone components.

SID-SER		**SID-PR**		**SID-ASR**		**SER-PR**		**SER-ASR**		**PR-ASR**	
SID	SER	SID	PR	SID	ASR	SER	PR	SER	ASR	PR	ASR
2.04	-0.93	**3.03**	-1.48	**2.46**	**1.60**	**0.23**	-1.85	-0.49	**0.76**	-1.15	-1.22
SER-SID		**PR-SID**		**ASR-SID**		**PR-SER**		**ASR-SER**		**ASR-PR**	
SID	SER	SID	PR	SID	ASR	SER	PR	SER	ASR	PR	ASR
5.26	**0.25**	**1.35**	-10.75	**6.33**	**0.79**	-0.77	-3.82	-1.39	**0.63**	-2.60	-1.72

and ASR with the pattern shown in Figure 1, where the distribution of the importance vector for PR is mostly covered by ASR, and ASR by SER. Despite the relatively high separation score in Table 1 and less overlap in Figure 1 between SID and ASR, respectively, the output from the utterance-level SID task is useful for improving the frame-level ASR task. Nevertheless it aligns with the design of a sequential disentanglement model for temporal data like speech (Hsu et al., 2017; Yingzhen & Mandt, 2018). Although the proposed metric is measured using target labels, future work may explore a self-supervised setup using techniques such as contrastive learning and data augmentation for hierarchical speech representation learning.

5 Conclusion

This work proposes an information-theoretic quantification as an alternative view of disentanglement from latent factors to downstream task use. The proposed metric is first validated in a multi-stage training setup, where the model output from a given task can effectively filter out the relevant information of the other task. It is further examined in the optimisation of a minimum sufficient hierarchical model. The associated distribution of parameters to calculate the metric manifests a hierarchical structure in terms of coverage, providing interpretation for the optimal hierarchical model design to combine two tasks. This empirical evidence enriches the scientific understanding of speech representations from the view of downstream tasks, establishing a formal evaluation protocol for future universal speech representation learning.

AI use statement

In this work, we have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, clean and reformat datasets, support qualitative and thematic data analysis, or interpret results. We take responsibility for the final content of this work, including text and claims.

Reproducibility statement

The experiments are documented in detail in this paper. Concretely, the corpus and SSL features are publicly available as addressed in Section 4.1. All the implementations are enhanced by minimising the impact of randomness and are based on an open-source benchmark for speech SSL models with modifications to support more tasks on the corpus. The tasks used in this work are either from the existing implementations or implemented by the authors with references and detailed descriptions (c.f. Section 4.1, 4.2 and 4.3). The correctness of some setups for some tasks is sanity-checked as shown in Table 5 of Appendix A. The calculation of the proposed information-theoretical measure is explained step-by-step in Section 3. In addition, all the experimental results can be found either in the main text or in the Appendix. We believe this information is useful and sufficient to reproduce the experimental results.

REFERENCES

Zakaria Aldeneh and Emily Mower Provost. You're not you when you're angry: Robust emotion features emerge by recognizing speakers. *IEEE Transactions on Affective Computing*, 14(2): 1351–1362, 2021.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *In Proceedings of the 34th Advances in Neural Information Processing Systems*, 33:12449–12460, 2020.

Tal Barami, Nimrod Berman, Ilan Naiman, Amos H Hason, Rotem Ezra, and Omri Azencot. Disentanglement beyond static vs. dynamic: A benchmark and evaluation framework for multi-factor sequential representations. *Advances in Neural Information Processing Systems*, 38, 2026.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335–359, 2008.

Marc-André Carbonneau, Julian Zaidi, Jonathan Boilard, and Ghyslain Gagnon. Measuring disentanglement: A review of metrics. *IEEE transactions on neural networks and learning systems*, 35 (7):8747–8761, 2022.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

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

Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations. In *International conference on learning representations*, 2018.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Luigi Gresele, Julius Von Kügelgen, Vincent Stimper, Bernhard Schölkopf, and Michel Besserve. Independent mechanism analysis, a new concept? *Advances in neural information processing systems*, 34:28233–28248, 2021.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. *Advances in neural information processing systems*, 30, 2017.

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

Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *Advances in neural information processing systems*, 35:28708–28720, 2022.

Aapo Hyvärinen and Patrik Hoyer. Emergence of phase-and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural computation*, 12(7):1705–1720, 2000.

Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in neural information processing systems*, 29, 2016.

Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd international conference on artificial intelligence and statistics*, pp. 859–868. Pmlr, 2019.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International conference on machine learning*, pp. 2649–2658. PMLR, 2018.

Alexander Liu, Sang-gil Lee, Chao-Han Huck Yang, Yuan Gong, Yu-Chiang Frank Wang, James R Glass, Rafael Valle, and Bryan Catanzaro. Uniwav: Towards unified pre-training for speech representation learning and generation. In *International Conference on Learning Representations*, volume 2025, pp. 64099–64116, 2025.

Alexander H Liu, Heng-Jui Chang, Michael Auli, Wei-Ning Hsu, and Jim Glass. Dinosr: Self-distillation and online clustering for self-supervised speech representation learning. *Advances in Neural Information Processing Systems*, 36:58346–58362, 2023.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Stefan Matthes, Zhiwei Han, and Hao Shen. Mechanistic independence: A principle for identifiable disentangled representations. In *International Conference on Learning Representations*, volume 2026, pp. 109930–109969, 2026.

Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, et al. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1179–1210, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ankita Pasad, Bowen Shi, and Karen Livescu. Comparative layer-wise analysis of self-supervised speech models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Ankita Pasad, Chung-Ming Chien, Shane Settle, and Karen Livescu. What do self-supervised speech models know about words? *Transactions of the Association for Computational Linguistics*, 12: 372–391, 2024.

Wenjie Peng, Chen Chen, and Thomas Hain. Difference in task performance on sparse speech representations. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28278–28291, 2026.

Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. *Advances in neural information processing systems*, 31, 2018.

Anna Sepliarskaia, Julia Kiseleva, Maarten de Rijke, et al. Evaluating disentangled representations. *arXiv preprint arXiv:1910.05587*, 2019.

Ismail Rasim Ulgen, Zongyang Du, Carlos Busso, and Berrak Sisman. Revealing emotional clusters in speaker embeddings: A contrastive learning strategy for speech emotion recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12081–12085. IEEE, 2024.

Aaron Van Den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017.

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, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. Superb: Speech processing universal performance benchmark. In *Interspeech 2021*, pp. 1194–1198, 2021. doi: 10.21437/Interspeech.2021-1775.

Shu-wen Yang, Heng-Jui Chang, Zili Huang, Andy T Liu, Cheng-I Lai, Haibin Wu, Jiatong Shi, Xuankai Chang, Hsiang-Sheng Tsai, Wen-Chin Huang, et al. A large-scale evaluation of speech foundation models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32: 2884–2899, 2024.

Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *International Conference on Machine Learning*, pp. 5670–5679. PMLR, 2018.

A Downstream task performance

Performance on different data splits. The complete evaluation performance on the test sets per task is shown in Table 4, where the final number is an average of the results from five-fold cross-validation. In contrast to SID, the data splits for SER, PR and ASR don't have speaker overlaps and the total number of utterances is identical for the four tasks.

Table 4: Performance on test set across the tasks with different SSL features as input.

SSL model	**Layer**	**SID**	**SER**	**PR**	ASR	
		Acc ↑	Acc ↑	PER ↓	CER ↓	WER ↓
WavLM	12	47.36	62.36	31.12	33.75	75.26
WavLM	0	76.79	57.08	69.09	72.99	98.58
wav2vec 2.0	12	66.56	56.54	56.65	65.19	98.31
wav2vec 2.0	0	78.96	56.80	63.37	70.50	98.33

Validation of task setups. Note that from Table 4 for the tasks of PR and ASR, the performance is low, as suggested by the high PER, CER and WER. One of the reasons may be that the models with a single linear layer are very weak. To verify this assumption, the two-layer BLSTM was adopted as a replacement for the single linear layer. Experimental results in Table 5 show that the performance can indeed be further improved with added complexity in the models, therefore validating the correctness of the setups for both tasks. Note that this work doesn't aim for better task performance but for a measure of the separation in representations for tasks.

Table 5: Ablations of model complexity for the performance of PR and ASR.

Model	**PR**	**ASR**	
	PER ↓	CER ↓	WER ↓
Linear layer × 1	31.12	33.75	75.26
BLSTM layer × 2	**25.97**	**21.96**	**46.52**

Performance on the same data splits. The evaluation performance on the test set for the non-SID tasks using the data splits with speaker overlap is shown in Table 6 below, which is used as the baseline results for the joint optimisation of hierarchical models as mentioned in Section 4.3.2.

Table 6: Performance on the test set across the tasks with the data splits from SID and input features from Layer 12 of WavLM.

SER	**PR**	**ASR**
Acc ↑	PER ↓	CER ↓
66.61	30.77	33.76

B Results on the combination of more tasks

The combination of more than two tasks for the separation score was calculated and is shown in Table 7 below. The pattern is similar to that in Table 1, where adding SID tends to increase while excluding it tends to decrease the score.

C Visualising the relative importance vector

Table 7: Measure of separation $d_{separation}$ derived from three or more tasks with different SSL features as input.

Input features		**Tasks**				
SSL model	Layer	SID-SER-PR	SID-SER-ASR	SID-PR-ASR	SER-PR-ASR	SID-SER-PR-ASR
WavLM	12	0.26	0.27	0.28	0.03	0.22
WavLM	0	0.49	0.51	0.59	0.09	0.47
wav2vec 2.0	12	0.33	0.36	0.31	0.03	0.28
wav2vec 2.0	0	0.41	0.39	0.52	0.10	0.38

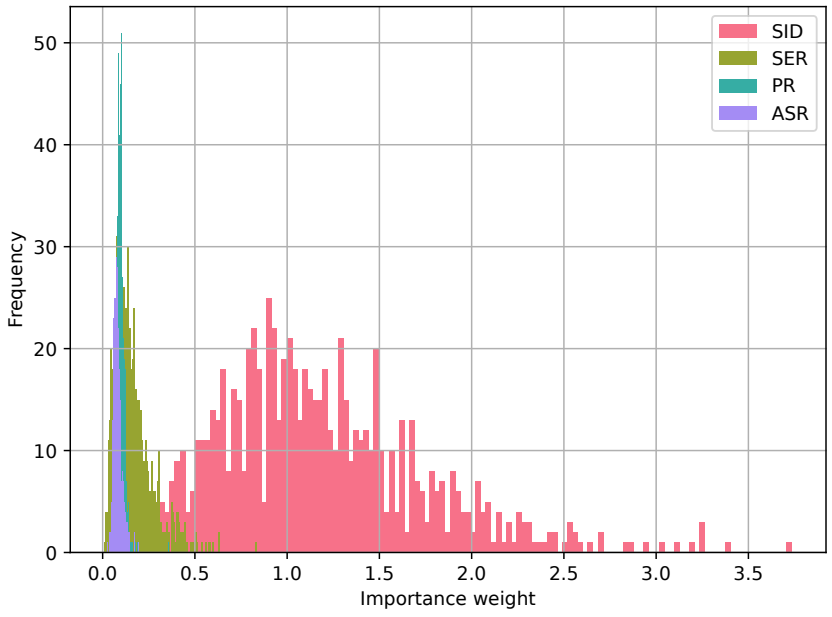


Figure 2: Histogram of the relative importance vector per task, with input features from the 0th layer of WavLM and 128 bins.

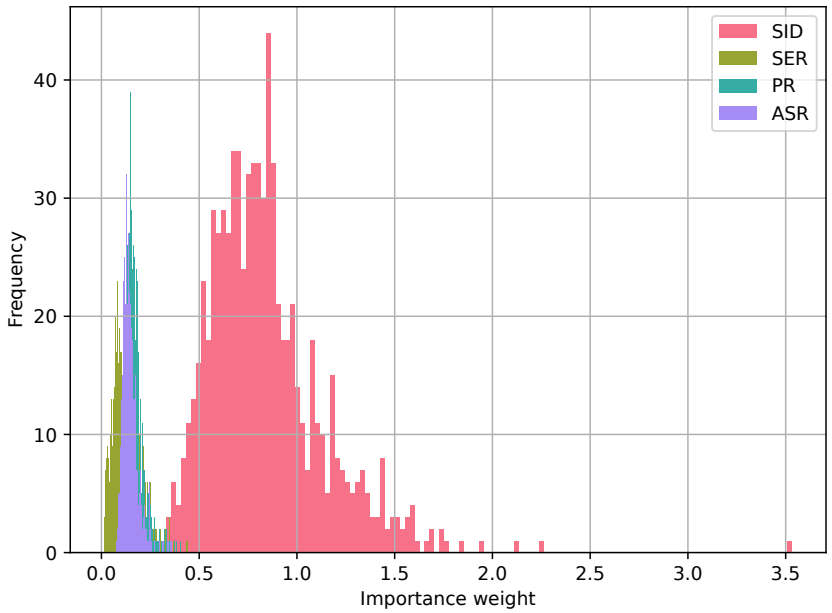


Figure 3: Histogram of the relative importance vector per task, with input features from the 12th layer of wav2vec 2.0 and 128 bins.

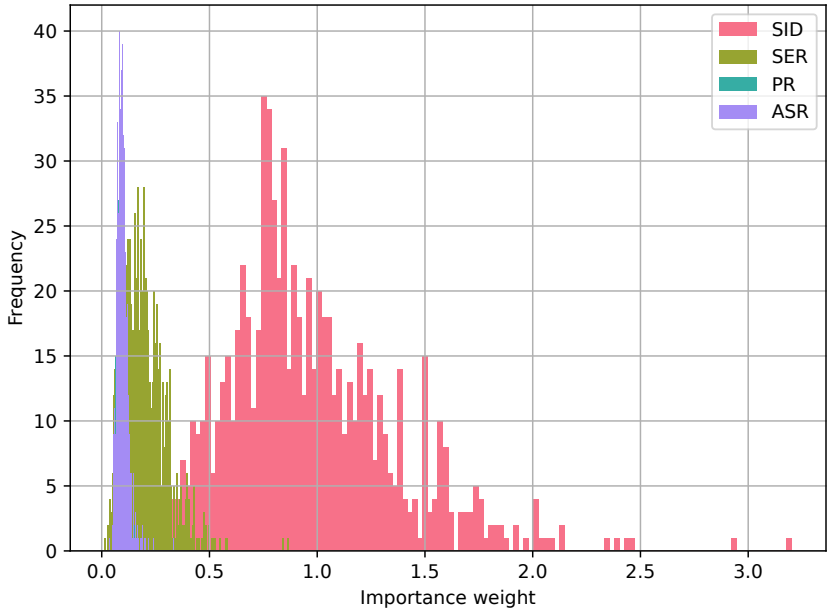


Figure 4: Histogram of the relative importance vector per task, with input features from the 0th layer of wav2vec 2.0 and 128 bins.