Automatic Data Augmentation Selection and Parametrization in Contrastive Self-Supervised Speech Representation Learning

Salah Zaiem, Titouan Parcollet, Slim Essid

Automatic Data Augmentation Selection and Parametrization in Contrastive Self-Supervised Speech Representation Learning

Salah Zaiem[1], *Titouan Parcollet*[2], *Slim Essid*[1]

[1]LTCI, Télécom Paris, Institut Polytechnique de Paris
[2]LIA, Avignon Université

salah.zaiem@telecom-paris.fr

Abstract

Contrastive learning enables learning useful audio and speech representations without ground-truth labels by maximizing the similarity between latent representations of similar signal segments. In this framework various data augmentation techniques are usually exploited to help enforce desired invariances within the learned representations, improving performance on various audio tasks thanks to more robust embeddings. Now, selecting the most relevant augmentations has proven crucial for better downstream performances. Thus, this work introduces a conditional independance-based method which allows for automatically selecting a suitable distribution on the choice of augmentations and their parametrization from a set of predefined ones, for contrastive self-supervised pre-training. This is performed with respect to a downstream task of interest, hence saving a costly hyper-parameter search. Experiments performed on two different downstream tasks validate the proposed approach showing better results than experimenting without augmentation or with baseline augmentations. We furthermore conduct a qualitative analysis of the automatically selected augmentations and their variation according to the considered final downstream dataset.
Index Terms: self-supervised learning, data augmentation.

1. Introduction and related works

Self-supervised learning (SSL) enables the use of large amounts of unlabeled data to obtain substantial performance improvements in a wide range of downstream tasks, without relying on costly and maybe imprecise manual annotations. Various approaches have thus been introduced and applied to speech data, including predictive coding [1, 2], multi-task learning [3, 4], encoding techniques [5] or contrastive learning [6, 7].

Contrastive Learning. Specifically, contrastive learning is one of the leading paradigms in speech self-supervised representation learning, especially towards solving paralinguistic classification tasks [8, 9]. COLA (COntrastive Learning for Audio) [6] is an audio-adapted version of these models. It consists in learning representations through assigning high-similarity to segments extracted from the same audio file and low-similarity to segments from different files. The learned representations are then fed to downstream models solving tasks. However, unlike similar approaches in the computer vision literature [10], COLA does not explore the use of data augmentation to enforce further invariances in the representations. This work explores this use and its variation with the considered downtream task.

Data augmentation in SSL settings. In this context, the creation of different versions, often called ”views”, of a given data point through data augmentation is an essential part of various self-supervised approaches [10, 11]. On speech data, Kharitonov *and al.* [12] have shown that using data augmentation to alter the data during Contrastive Predictive Coding (CPC) [13, 14] training improves the downstream ASR performance. Two works may be considered as close to the purpose of this paper. First, in image classification settings, adapting the augmentation distribution used in the contrastive pretraining to the downstream classification task has proven effective [15, 16]. This is particularly true when certain differences, to which the representations are trained to be invariant, are crucial for distinguishing the downstream classes. Second, experiments led on contrastive representations (COLA based) on sound classification show that augmenting the cut segments leads to better results, and that the set of best performing augmentations is downstream task dependent [17]. Nonetheless, while ablation studies are conducted on the selected augmentations, no prior justification of the choices are developed, making the selection relying on computationally heavy empirical exploration. Finally, a few works have attempted to define how views should be created in contrastive learning settings [18, 19], and thus which and how augmentations should be used. However, and to the best of our knowledge, there is no attempt to theoretically motivate data augmentation in self-supervised settings on speech or audio data. This work will rely on COLA approach as it is one of the closest to vanilla contrastive learning, and it did not explore the use of data augmentation on speech. It is, nonetheless, perfectly transferable to other contrastive approaches. If we were to rely only on empirical testing, evaluating a single set of augmentation distribution would require two full trainings. In the specific case of this paper, a single pretraining takes 2 days on a V100 GPU. The method we present prevents this, allowing for an efficient selection of an appropriate data augmentation distribution. The contributions of this work are thus threefold :

1. To highlight the impact of data augmentation on contrastive self-supervised speech representation learning.
2. To propose a method that selects a distribution on the choice of augmentations and their parametrization according to the downstream task of interest, validated on two different downstream tasks. The selected augmentations are qualitatively linked to the recording conditions.
3. To release the code base, implemented with SpeechBrain [20] for replication and further improvements. [1]

Figure 1 presents an overview of the led experiments, summarizing the three steps conducted for every downstream task. First, an augmentation distribution is selected (Section 2). Second, representations are learned through contrastive pretraining using the selected augmentation distribution (Section

[1]https://github.com/salah-zaiem/augmentations

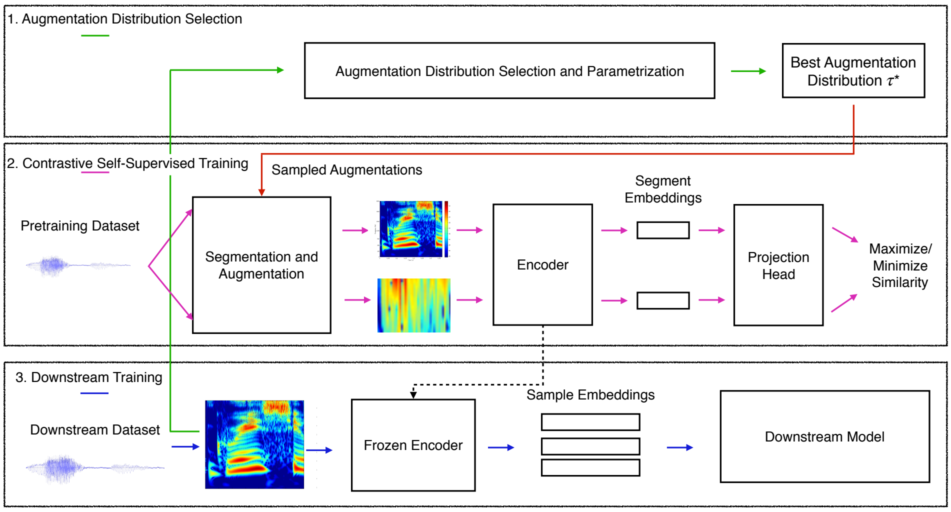

Figure 1: *The three steps of the validation process. (a) select the best augmentation distribution. (b) contrastive pretraining alterating the input points with the selected augmentation. (c) use the learned speech representations as input for downstream finetuning.*

3.1). Finally, the learned representations are fed to the downstream model to solve the considered task (Section 3.2).

2. Selecting the Augmentation Distribution

This section details the method developed to find a data augmentation distribution for the contrastive learning part, suitable to the final downstream task of interest. It starts by detailing the theoretical motivations behind the method, before delving into the technical details of the implementation.

2.1. Theoretical Motivation

During self-supervised training, the representations are learned through solving automatically generated pretext tasks. Lee *and al.* [21] theoretically proved a link between the downstream task performance and the conditional independence (CI) between the pretext task labels and the training samples given the downstream labels. In previous works [22, 4], we have shown that this relation holds even without the theoretical assumptions in [21], introducing a practical method to compute the conditional independence. Precisely, let X, Y and Z be, respectively, the downstream data points, the downstream labels and the pretext labels whose prediction is used as a pretext task. We have shown that the more is Z conditionally independent of X given Y, the more using the prediction of Z as a pretext task leads to a better downstream performance. To quantify the utility of a given pretext task, we use the kernelized independence test Hilbert Schmidt Independence Criterion (HSIC) [23]. Intuitively, the HSIC value is high if similar speech samples have similar pretext labels. The more the HSIC value is high, the more dependent are X and Z conditionally on Y. In [22, 4], we demonstrated that choosing the pretext labels Z minimizing $HSIC(Z, X|Y)$ leads to better downstream performances.

In this work, we extend these findings to the contrastive learning settings through the following steps. First, the key consists in considering that in the contrastive learning setting, the pretext task of assigning high similarity to segments originating from the same file can be seen as the prediction, given a random augmented segment, of the file it was generated from. An augmentation distribution τ is defined by a set of parameters defining how a chain of augmentations is sampled during training to be applied to the upcoming data points. More precisely, every distribution τ is represented as a vector of $P = 14$ parameters, where every parameter $(\tau(p))_{1 \leq p \leq P}$ is either the probability of applying an augmentation or a boundary for a uniform law from which a augmentation's internal parameter (e.g. room scale) is sampled. With X the speech samples and τ a distribution of augmentations, we define $X' = f(X, \tau)$ with f a function that randomly cuts segments from the speech samples and applies augmentations sampled from τ on them. Given a downstream dataset of samples (X, Y) and an augmentation distribution τ, we can generate N augmented segments per speech sample to get the augmented set of data points X'. To find the optimal augmentation distribution τ^* we resort to minimizing the HSIC quantity with the augmented dataset $X' = f(X, \tau)$ according to:

$$\tau^* = \arg\min_{\tau} HSIC(f(X, \tau), Z|Y) \tag{1}$$

with (X, Y) the downstream datapoints and labels, and Z the pretext labels corresponding here for every augmented view of a speech sample to the ID of the speech sample it originates from.

2.2. Implementation

In this work, we chose to limit ourselves to the set of augmentations used in [12] for two reasons. First, they have shown effective with the contrastive predictive coding approach improving the final discrimination performances. Second, they are easily implemented within PyTorch using the WavAugment library. Hence, five augmentations are considered: time dropping[24], pitch shifting [25], reverberation, clipping and band rejection[24]. The first parameters concern the probability of applying each one of the considered augmentations. The second set of parameters are related to those of the chosen augmentations in terms of signal effects; these are described in Table 1.

Since the considered augmentations are not differentiable, to minimize the HSIC test described above, we resort to a random search, sampling random distributions and selecting the one with the lowest dependance scoring. It is important to note here, that this phase does not involve any training, and is largely more efficient than thorough testing of the distributions, as a computation takes 3 hours on 20 CPUs. More precisely, for every considered downstream task, we first sample $p = 100$ parametrizations $(\tau_i)_{i\in[1,p]}$. For every parametrization τ_i, we compute the HSIC quantity in Eq.(1) following two steps. First, computing the augmented set $X'_i = f(X, \tau_i)$, by computing $N = 20$ views of every speech sample in X. Then, computing $HSIC(X', Z|Y)$ following the technique described in [22]. For every downstream task, the augmentation distribution with the lowest conditional dependance value is selected and will be used during the pretraining to train the encoder that will be exploited as a feature extractor in the downstream training.

Table 1: *Parameters considered, descriptions and ranges*

Name	Description	Range
Room scale min	Min room size	[0,30]
Room scale max	Max room size	[30,100]
Band Scaler	Scales the rejected band	[0,1]
Pitch Shift Max	Amplitude of a pitch shift	[150,450]
Pitch Quick pr.	Speeds pitch shifting	[0,1]
Clip Min	Minimal clip factor	[0.3, 0.6]
Clip Max	Maximal clip factor	[0.6, 1]
Timedrop max	Size of a time dropout	[30-150] ms

3. Experimental setup

This section describes the experiments led to validate the proposed approach and the selected augmentation distributions. It starts by describing the details of the contrastive learning phase before reporting the downstream finetuning conditions.

3.1. Contrastive Learning

As shown in Figure 1, during the contrastive pre-training, we start by extracting two random segments from every speech sample of a given batch. These segments are then alterated using the considered augmentation distribution before being fed to the encoder. Our pretraining model takes as input the speech samples as 64-Mel band spectrograms. The frame size is 25ms and hop size 10ms. As in COLA, the encoder is an EfficientNet-B0 [26], a lightweight convolutional neural network. We cut from the input speech samples 1-second long segments that are augmented using the considered augmentation distribution. Fixing the length of the extracted segments allows the use of EfficientNet-B0 even though it has been originally proposed for computer vision, as fixed length Mel-spectrograms have a 2D structure similar to image inputs. The encoder applies a global time-pooling at its final layer to get a 1280-dimensioned embedding h that represents the whole segment and that will be the one used for downstream finetuning. During the pretraining phase, this embedding is then projected with a dense layer followed by a layer normalization and a hyperbolic tangent activation to a 512-sized vector v. Learning consists in maximizing the similarity of segments originating from the same file, while minimizing that of those that do not. As suggested by the final results obtained with COLA, the similarity is computed using the bilinear similarity. More precisely, if g is the function regrouping the encoder and the projection head, x_1 and x_2 two speech segments and W the bilinear parameters, then the similarity function is $s(x_1, x_2) = g(x_1)^T\ W\ g(x_2)$. The input is a batch of size B of distinct speech files that we denote $(x_i)_{i\in[1,B]}$, and a selected augmentation distribution τ from which we can sample at each iteration two augmentation functions A_τ and A'_τ. From each speech sample, two random segments of length 1 second are cut. The first is altered using A_τ while the second undergoes the A'_τ alteration, leading to two sets $(\tilde{x}_i)_{i\in[1,B]}$ and $(\tilde{x}'_i)_{i\in[1,B]}$. Finally, the loss function for pretraining is the multi-class cross entropy over the bilinear similarity scores:

$$\mathcal{L} = -\log \frac{e^{s(\tilde{x}_i, \tilde{x}'_i)}}{e^{s(\tilde{x}_i, \tilde{x}'_i)} + \sum_{j\neq i} e^{s(\tilde{x}_i, \tilde{x}'_j)}}. \quad (2)$$

Pretraining dataset. The train set of the English Common Voice dataset (version 8.0) [27] is used for SSL pretraining (2185 hours). Common Voice is a collection of speech utterances from worldwide users recording themselves from their own devices. Hence, the closeness to natural settings makes it a suitable choice for self-supervised learning. We remove from Common Voice the sentences lasting more than 10 seconds, as they often contain long silence parts due to open microphones. It is important to note that since the COLA embeddings were originally introduced to set non-speech tasks as well, they were trained on AudioSet [28], which contains speech and non-speech utterances. Since we will be only working on speech downstream tasks, we chose to use a speech-only pretraining. We also use a 1024 batch size. All the models are pre-trained for 100 epochs with ADAM and a 10^{-4} learning rate.

3.2. Downstream finetuning

Two downstream tasks are considered in this work: speaker identification and language identification. Two reasons motivate this choice. First, among the list of tasks COLA was applied on, we chose the two downstream tasks exhibiting the largest room for improvement. Second, we wanted two tasks that would require different aspects of the considered speech signal, thus maybe requiring different sets of augmentations. A study validating this assumption is provided in Section 4.1. VoxCeleb1 [29] is used for the speaker recognition task. The training set contains $148,642$ utterances from 1251 different speakers. We use the available identification split for testing. VoxForge [30] is used for language identification. 6 European languages are present in the 176,438 samples of the dataset, two tenths are kept for validation and testing.

During the downstream finetuning the projection head is discarded and replaced with a linear classifier directly on top of the encoder. The contrastive encoder is frozen during the finetuning phase as we want it to be used solely as a fixed feature extractor to properly assess the impact of our data augmentation selection on the obtained representation. In the COLA paper, the final class prediction is obtained through averaging the predictions of non-overlapping cut segments of a given test utterance. However, we found it more effective to use the mean over the embeddings of overlapping segments. We proceed in this manner: during training and testing, we cut a 1-s long segment every 200ms , encode every segment separately, and then use the mean over the encoded representations as a sequence embedding to the classifier. We train on the downstream task for 10 epochs with ADAM with a 10^{-3} learning rate and the additive angular margin loss [31] with margin 0.2 and scale 30.

4. Results and Discussion

Table 2 shows the results obtained on the two considered downstream tasks. The "COLA" column shows the results obtained in the original paper. The "Without" column is our implementation of the algorithm without any augmentation during pretraining. "Basic" shows the results reached using the baseline WavAugment augmentation parameters. Finally, the results obtained using the augmentation choice based on the proposed

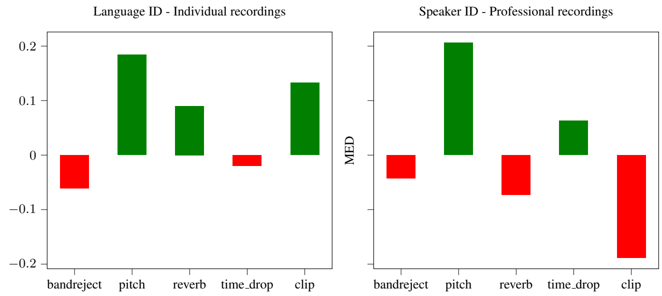

Figure 2: *Difference of the probability of picking an augmentation between the best and worst scoring augmentations, depending on the downstream dataset. Green bars show augmentations that are more likely to get picked for the best scoring distributions for that task. For instance, the far right bars indicate that clipping is an encouraged augmentation on VoxForge, and is discouraged on VoxCeleb1.*

technique can be found in the "Selected" column. The first observation is that the selected augmentation technique outperforms the baselines on the two considered tasks. For speaker identification, the accuracy obtained with the selected distribution is 46% higher than the non-augmented COLA, and 4% higher than the baseline augmentations. An important point is that in the baseline augmentation setup (i.e. "Basic"), all the augmentations are systematically performed on the input points, thus considerably slowing the pretraining. Indeed, with WavAugment augmentations being CPU-processed, we witnessed that dividing by half the conducted augmentations by lowering their probability, leads to 20% faster trainings.

4.1. Discussion

In this part, we will discuss the automatically selected data augmentations, and analyze their dependence on the downstream dataset. We will study first the dependence of the probabilities of applying a given augmentation according to the downstream dataset of interest. Then, we will consider the choice of a few interpretable parameters. This is done through the following procedure: for every downstream task, we start by selecting $k = 10$ best and worst augmentation distributions according to our HSIC scoring. The "Mean Extremal Difference" or "MED" is finally obtained by computing the difference between the two means originating from these two groups *i.e.*, best and worst. More precisely, for an augmentation parameter p:

$$MED(p) = \frac{1}{k}(\sum_{i=0}^{k} \tau_i^{best}(p) - \tau_i^{worst}(p)) \qquad (3)$$

with τ_i^{best} being the i-th best distribution, τ_i^{worst} being the i-th worst and $\tau(p)$ being the value of parameter p in τ.

Figure **??** depicts these values for the probabilities of applying each of the five considered alterations in this work. Green bar means are for positive values, indicating that this augmentation is more likely to be applied in the supposedly best distributions. We observe that clipping and reverberation are more selected for language identification on VoxForge than for speaker identification on VoxCeleb. We think that this is mainly due to the type of recording rather than to the nature of the task. VoxForge samples come from individual contributors that record themselves speaking their native language. The varying recording conditions lead to clipping or heavy reverberation issues, which may be the reason behind the selection of these augmentations in this case. Figure 3 shows the mean difference defined above on 3 parameters, which are time dropping and room scale boundaries. Concerning reverberation, it is worth noting that room scales are smaller for VoxForge than for VoxCeleb1, which is once again coherent with the recording conditions, as the first ones are recorded at home, compared to studio conditions. Samples of augmented speech files with various distributions are provided for quantitative comparison by the readers.[2]

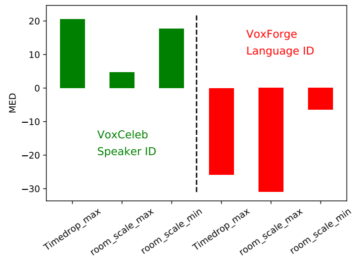

Figure 3: *MED for selected parameters, for every downstream task. Reverb room sizes are coherent with the difference in recording conditions between the two datasets.*

Table 2: *Results for the two considered downstream tasks. COLA column shows the result of the original paper. "Basic" shows the result with the basic WavAugment recipe. "Selected" shows our approach results.*

Down. Task	**COLA**	**Our Implementations**		
		Without	Basic	Selected
Language ID	71.3	84.9	84.3	**85.2**
Speaker ID	29.9	32.0	45.1	**46.9**

5. Acknowledgements

This work has benefited from funding from l'Agence de l'Innovation de Défense, and was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011012801R1).

6. Conclusion

Self-supervised learning of speech representations is a computationally intensive technology, especially when using data augmentation within contrastive schemes. We introduced a novel informed method enabling the automatic selection and parametrization of the crucial data augmentation pipeline. Our findings open a range of possibilities in signal alterations exploration for self-supervision.

[2] `salah-zaiem.github.io/augmentedsamples/`

7. References

[1] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020.

[2] A. T. Liu, S.-w. Yang, P.-H. Chi, P.-c. Hsu, and H.-y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020.

[3] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," 2020.

[4] S. Zaiem, T. Parcollet, and S. Essid, "Pretext tasks selection for multitask self-supervised speech representation learning," 2021. [Online]. Available: https://arxiv.org/abs/2107.00594

[5] R. Algayres, M. S. Zaiem, B. Sagot, and E. Dupoux, "Evaluating the reliability of acoustic speech embeddings," in *INTERSPEECH 2020 - Annual Conference of the International Speech Communication Association*, Shanghai / Vitrtual, China, Oct. 2020.

[6] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," 2020. [Online]. Available: https://arxiv.org/abs/2010.10915

[7] J. Shor, A. Jansen, R. Maor, O. Lang, O. Tuval, F. de Chaumont Quitry, M. Tagliasacchi, I. Shavitt, D. Emanuel, and Y. Haviv, "Towards learning a universal non-semantic representation of speech," in *Interspeech 2020*. ISCA, oct 2020. [Online]. Available: https://doi.org/10.21437Finterspeech.2020-1242

[8] J. Shor, A. Jansen, W. Han, D. Park, and Y. Zhang, "Universal paralinguistic speech representations using self-supervised conformers," 2021. [Online]. Available: https://arxiv.org/abs/2110.04621

[9] H. Al-Tahan and Y. Mohsenzadeh, "CLAR: Contrastive Learning of Auditory Representations," Tech. Rep., 2021. [Online]. Available: https://github.com/

[10] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," 2020. [Online]. Available: https://arxiv.org/abs/2002.05709

[11] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," 2020. [Online]. Available: https://arxiv.org/abs/2006.07733

[12] E. Kharitonov, M. Rivière, G. Synnaeve, L. Wolf, P.-E. Mazaré, M. Douze, and E. Dupoux, "Data augmenting contrastive learning of speech representations in the time domain," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 215–222.

[13] M. Rivière, A. Joulin, P.-E. Mazaré, and E. Dupoux, "Unsupervised pretraining transfers well across languages," 2020.

[14] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *ArXiv*, vol. abs/1807.03748, 2018.

[15] T. Xiao, X. Wang, A. Efros, and T. Darrell, "What Should Not Be Contrastive in Contrastive Learning," 2020.

[16] Y. Li, R. Pogodin, D. J. Sutherland, and A. Gretton, "Self-supervised learning with kernel dependence maximization," 2021.

[17] M. Emami, D. Tran, and K. Koishida, "Augmented Contrastive Self-Supervised Learning for Audio Invariant Representations," dec 2021. [Online]. Available: http://arxiv.org/abs/2112.10950

[18] S. Arora, H. Khandeparkar, M. Khodak, O. Plevrakis, and N. Saunshi, "A Theoretical Analysis of Contrastive Unsupervised Representation Learning," *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 9904–9923, feb 2019.

[19] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning," 2020, cite arxiv:2005.10243Comment: submitted to ECCV 2020. [Online]. Available: http://arxiv.org/abs/2005.10243

[20] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "Speechbrain: A general-purpose speech toolkit," 2021.

[21] J. D. Lee, Q. Lei, N. Saunshi, and J. Zhuo, "Predicting what you already know helps: Provable self-supervised learning," 2020.

[22] S. Zaiem, T. Parcollet, and S. Essid, "Conditional independence for pretext task selection in self-supervised speech representation learning," in *Interspeech 2021*. ISCA, aug 2021. [Online]. Available: https://doi.org/10.214372Finterspeech.2021-1027

[23] A. Gretton, K. Fukumizu, C. Teo, L. Song, B. Schölkopf, and A. Smola, "A Kernel Statistical Test of Independence," in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20. Curran Associates, Inc., 2007.

[24] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech 2019*. ISCA, sep 2019. [Online]. Available: https://doi.org/10.214372Finterspeech.2019-2680

[25] K. Lent, "An efficient method for pitch shifting digitally sampled sounds," *Computer Music Journal*, vol. 13, no. 4, pp. 65–71, 1989. [Online]. Available: http://www.jstor.org/stable/3679554

[26] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2019. [Online]. Available: https://arxiv.org/abs/1905.11946

[27] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," 2020.

[28] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.

[29] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," *Interspeech 2017*, Aug 2017.

[30] K. MacLean, "Voxforge," 2018.

[31] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," 2018. [Online]. Available: https://arxiv.org/abs/1801.07698