

Audio Self-supervised Learning: A Survey

Shuo Liu, Adria Mallol-Ragolta, Emilia Parada-Cabaleiro, Kun Qian, *Senior Member, IEEE*,
 Xin Jing, Alexander Kathan, Bin Hu, *Senior Member, IEEE*, and Björn W. Schuller, *Fellow, IEEE*

Abstract—Inspired by the humans’ cognitive ability to generalise knowledge and skills, Self-Supervised Learning (SSL) targets at discovering general representations from large-scale data without requiring human annotations, which is an expensive and time consuming task. Its success in the fields of computer vision and natural language processing have prompted its recent adoption into the field of audio and speech processing. Comprehensive reviews summarising the knowledge in audio SSL are currently missing. To fill this gap, in the present work, we provide an overview of the SSL methods used for audio and speech processing applications. Herein, we also summarise the empirical works that exploit the audio modality in multi-modal SSL frameworks, and the existing suitable benchmarks to evaluate the power of SSL in the computer audition domain. Finally, we discuss some open problems and point out the future directions on the development of audio SSL.

Index Terms—Self-supervised learning, audio and speech processing, multi-modal SSL, representation learning, unsupervised learning

I. INTRODUCTION

ACCORDING to Piaget’s theory of cognitive development [1], [2], since their birth up to approximately 18 months, children acquire knowledge from sensory and motor experiences. During this stage, i.e., the ‘sensorimotor’ stage, through basic actions such as sucking, grasping, looking, and listening, the early representational thought emerges [3]. Along with the acquisition of knowledge, over the different developmental stages until the last one, i.e., the ‘formal operational’ (adolescence and adulthood) stage, children’s reasoning progressively moves towards the acquisition of abstract ideas and the use of deductive logic, i.e., subtracting specific information from a general principle [4]. During this process, in order to understand the world, the so-called ‘schemas’, i.e., *higher-order cognitive structures that have been hypothesised to underlie many aspects of human knowledge and skill* [5], emerge. According to Piaget, children development is interpreted through an equilibration mechanism which explains how new information is balanced according to old knowledge. Equilibration involves ‘assimilation’ (the process of taking in

Manuscript received Jan, 2022.

Shuo Liu, Adria Mallol-Ragolta, Xin Jing, Alexander Kathan are with the Chair of Embedded Intelligence for Health Care & Wellbeing, University of Augsburg, Augsburg 86159, Germany (e-mail: {shuo.liu, adria.mallolragolta, xin.jing, alexander.kathan}@uni-a.de).

Emilia Parada-Cabaleiro is with the Institute of Computational Perception, Johannes Kepler University Linz, Linz 4040, Austria (e-mail: emilia.paradacabaleiro@jku.at).

Kun Qian and Bin Hu are with the School of Medical Technology, Beijing Institute of Technology, China (e-mail: {qian, bh}@bit.edu.cn)

Björn W. Schuller is leading the Chair of Embedded Intelligence for Health Care & Wellbeing, University of Augsburg, Augsburg 86159, Germany. He is also the head of GLAM – the Group on Language, Audio, & Music, Imperial College London, London SW7 2AZ, UK. (e-mail: björn.schuller@imperial.ac.uk).

new information to fit in with the pre-existing schemas) and ‘accommodation’ (the process of modifying the pre-existing schemas as a result of new information) [1], [6]. In this view, learning is possible if complex structures are based on simpler ones, i.e., when a natural development between structures exists instead of a simple external reinforcement [1]. Indeed, the interesting aspect of learning (and one of the main goals in education) is to create dynamic structures which can lead to generalisation, i.e., the ability to apply learnt knowledge and skills for understanding a different context. This is known as ‘transfer of learning’ [7]. Inspired by the cognitive process of developing dynamic structures with the capability of generalisation, the Self-Supervised Learning (SSL) paradigm has been presented [8]–[11] — a machine and deep learning technique rapidly evolving in the last years. SSL targets at learning a model that is able to produce universal representations. This is approached by first solving some pretext tasks (also known as upstream tasks in literature), i.e., a procedure which, similarly to the sensorimotor stage, enables to artificially learn representations directly from the data attributes without the need of human-annotations [12]. Then, with a pre-trained model generated on the pretext task, feature representations are extracted to understand new data, i.e., similarly to cognitive development, a pre-trained model (previous knowledge) can be used through generalisation to understand a new context, a process known as downstream task [13].

SSL mitigates two difficulties that currently limit the application of deep learning: the need of human annotations and the difficulty in designing effective network architectures for specific tasks. First, the current success of deep learning reckons on big data which typically consumes uninhibited human efforts in annotations. This faces the issue of annotation bias as well as the fact that annotation procedures often cannot optimally preserve data privacy. As SSL learns representations from the data itself without the need of labels [14] (sometimes creates pseudo-labels for self-supervision), it overtakes the challenges derived by the use of human annotations. Second, as long as the pretext model can generate proper representations of the data, these can be used for multiple downstream tasks, reducing, at the same time, the difficulty in designing reliable downstream models. For instance, a Multi-layer Perceptron (MLP) is commonly used for this step, reaching state-of-the-art results for different research areas in computer science. As the main effort of SSL concentrates on the development of well-trained upstream models, it guarantees to extract data representations with a sufficient level of generalisation and distinctiveness. Furthermore, as a way to increase distinctiveness of the learnt representations, when solving pretext tasks, negative examples can be additionally

provided in order to contrast the target sample with negative examples [15]. This process formalises the SSL into a contrastive learning framework [16]–[19]. Finally, it has been shown in many works, such as [20] that the amount of labelled data needed to fine-tune a model during the downstream task considerably decreases when taking upstream models into account, which makes using SSL effective in the design of efficient architectures.

The fitness of upstream and downstream tasks, i.e., how much the knowledge learnt from pretext tasks is applicable to the downstream tasks, is partially determined by the data relevance used in both steps. From a cognitive point of view, this is comparable to the aforementioned ‘transfer of learning’, as a speaker of a given language would find it easier to learn a related language (near transfer) than an unrelated one (far transfer). Thus, near transfer of knowledge is expected to ensure the downstream tasks to particularly benefit from the upstream training. However, far transfer may also occur. Therefore, downstream tasks that use data from a different domain can still benefit from learning representations of sufficient generalisability. The versatility of SSL has yielded to a superior performance in several research fields, such as, Natural Language Processing (NLP) [21] and Computer Vision (CV) [8], [11], as well as in a variety of deep learning methods, e.g., graphical neural networks [22] and reinforcement learning [23], to name a few. Nevertheless, processing audio sources increases further the difficulty of applying deep learning methods, as in real-world, this modality is typically characterised by many uncertainties. Speech, for instance, due to within- and cross-speaker variations, such as those produced by disfluencies, as well as differences in language, acoustic environments, or recording setups, presents usually a considerable variability. This makes it difficult deducing relevant latent structures without taking into account any supervision guidance. In addition, unlike for images, overlapping noises are typical of recordings. Through its masking properties, surrounding noises limit (and even impede) understanding, in some cases distorting the spectrogram of the the audio content of interest. Indeed, as each pixel (time-frequency bin) of the spectrogram can be deteriorated, noise reduction and removal is still an open challenge in the field [24]. Similarly, in comparison to NLP tasks, which (despite their inherent difficulties) process texts that are comprised of limited possible words and characters, the infinite possibilities of audio that represent the same meaning creates more uncertainties in audio understanding. These might be indicators that explain why SSL has achieved lower performance for audio signal processing than for CV and NLP.

Specifically, a proper SSL model should be able to extract representations that are: (i) distributed, i.e., more expressive as the dimensionality increases; (ii) abstract, i.e., aggregate more abstract features which are invariant to local changes in the input; (iii) disentangled, i.e., each factor of the representation vector should be interpretable [9]. As SSL requires from a model both, generalisation and discrimination (in parallel), using SSL for audio processing becomes particularly challenging. Although several survey articles aimed to give an overview of the existing literature on SSL have been presented

to the research community, due to the prominent use of SSL in CV and NLP, these works show a clear bias towards these two fields [8], [11], [21]. However, despite the challenges, recent research has shown an always increasing interest in applying SSL to audio sources. As this rapidly developing area has not been systematically investigated yet, to fill this gap, we present a survey on SSL with a special emphasis on the recent progress, by this including for the first time an overview of SSL in audio within unified frameworks. By providing an overview of the existing techniques as well as a disambiguation between approaches, this work is especially thought to support both beginners and more experienced researchers interested in the use of SSL for audio signal processing.

The rest of the manuscript is organised as follows. We first introduce SSL in Section II, aiming to unify its frameworks and cover the basic useful blocks and operations that lead to its success. Then, we describe how this unified framework can fit for audio processing in Section III with discussion focus on training objectives, network architectures, and the training framework. Next, SSL approaches exploiting audio as one of the modalities will be discussed in Section IV. In Section V, we summarise the downstream tasks considered in the literature and list the databases and benchmarks that are used for evaluating the performance of pretext tasks. At last, we discuss several aspects of SSL, including its relations and difference to other similar deep learning techniques, before drawing a conclusion and pointing out the potential research directions.

II. SELF-SUPERVISED LEARNING: A GENERAL OVERVIEW

SSL aims at learning latent representations from large-scale data by solving designed pretext tasks, rather than using human annotations. To this end, different views of an object, which are of high natural correlation, are created. An SSL model is trained to generalise, to some extent, its representations in a latent high-dimensional space. By comparing the representations of the same object to other objects (defined as negative samples) in training, an SSL model is expected to produce representations that are of better distinctiveness. Dependent on whether negative samples are taken into the training process, SSL frameworks can be categorised into two classes: Predictive and contrastive (cf. Figure 1). We will introduce these two basic frameworks and their variants in Section II-B1 and Section II-B2, respectively. In addition, as predictive models can also utilise contrastive loss as its training objectives, contrastive predictive coding (CPC), based on auto-regressive predictive coding (APC) and masked predictive coding (MPC), will be described in Section II-B3. We will also discuss the benefit of employing negative samples in training SSL models and the potential drawbacks. As the process of generating the views of an object is the fundamental step for SSL, we will start by summarising the different methods in the following.

A. Views Generation

A typical way to get multiple views of the same context is recording it by the use of multiple sensors, for example, using

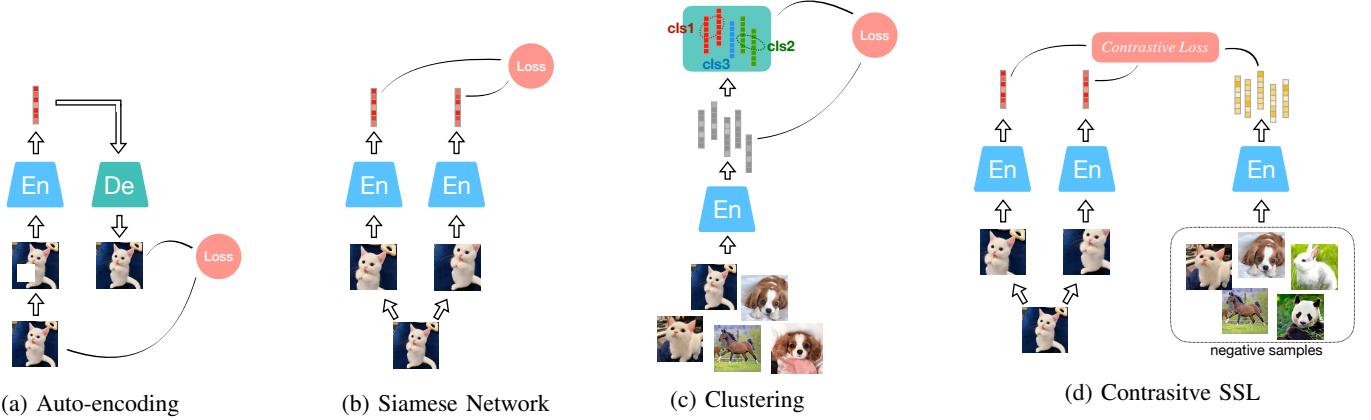


Fig. 1: Predictive SSL frameworks (a-c) and contrastive SSL framework (d). For each framework, the diagram shows the components, including pseudo-labels that are used to construct training objectives. (a) Auto-encoding contains an encoder (En) and a decoder (De). The encoder learns representations from a distorted signal input while the decoder aims at recovering the clean signal from the learnt representations; (b) a Siamese network processes two views of the same data point, hence the latent representation of one sub-network is seen as pseudo-label of the other sub-network; (c) clustering is applied for grouping the learnt representations – the clustering centroids are used as pseudo-labels for training; (d) contrastive SSL constructs the contrastive loss through negative samples.

multiple cameras to shoot a scene from different angles [25]. The views are not limited to be of the same modality but can also belong to a different one, such as an utterance with its text transcriptions [25], [26] or audio and image frames from the same video source [27], [28]. As using multi-modality and cross-modality is typical in SSL, in Section IV, the works specific to audio representation will be presented.

An alternative way to create effective views in CV-related tasks is through data transformation, including image cropping [20], [29], rotation [30], colouring [31], and distortion [20], amongst others. In several audio works, after converting an audio waveform to its spectrogram or Mel representations, similar paradigms can be applied [32], [33]. In this case, data augmentation techniques such as warping and SpecAugment [34] can also be used for creating another view of the audio signal.

Considering the spatial or temporal coherence and consistency in a signal, local features of different patches of an image or segments of an audio can be considered as multiple views of the same type of data [25], [35]. This approach is widely adopted for sequential input, such as video as a sequence of image frames [25], where two frames in a short temporal range are considered a positive pair, while frames that are far away in the same sequence or from other sequences are taken as negative samples. For an audio sequence, the positive pair can be segmented from the same recording, and the negative pair can be extracted from different recordings [36]–[38].

Another solution is to exploit the relationship between local features and global information, which aims at maximising their mutual information [39]. The global representation, often defined as context vector, aggregates the information of the entire context. An SSL model then learns to represent local features by capturing meaningful information relevant to the aggregated global representation. Deep InfoMax [39] codes an image into a global context vector, and contrasts its distance to the spatial patches of the same image against

the distance to spatial patches of different images. More often, the mechanism is used for a sequential data structure. As in [40], a context vector collects the global information along the temporal direction in an auto-regressive manner. The context vector is then compared to the learnt representations of each local frame. Note that, in this work, contrastive loss (introduced in Section II-B2) has been used in the framework of an auto-regressive coding model, leading to a contrastive predictive coding that can learn effective fixed-length audio representations from speech input. We leave its discussion for the next section, to which the reader is referred for more details. Finally, [41] investigates the effect of the redundancy in two views of the positive pair, suggesting that the views with less mutual information should be selected for training. The idea is to compress the redundancy in the embeddings of the two views that are not relevant to the labels [41]. Similarly, [42] provides also empirical evidences about the fact that the contrastive loss is not only attributed to mutual information.

B. SSL Frameworks

1) Predictive Models: Predictive SSL, different from contrastive SSL, optimises the similarity or correlations between the representations of two views of the same object, without considering their similarity to that of negative samples in training objectives. Typical frameworks are auto-encoding and Siamese Networks. A clustering method requires no additional views generation, but explicitly groups the learnt representations based on the underlying similarity between each input. We describe the three typical predictive SSL frameworks in detail in the following.

Auto-encoding is based on the use of auto-encoders [60], as depicted in Figure 1a. A standard auto-encoder learns a compressed latent embedding that represents the input of the encoder and expects to reconstruct the original input from the latent representation, i.e., the decoder output. The dimensionality of the latent representation must be carefully designed,

TABLE I: An overview of the recent typical self-supervised learning methods. **FOS** abbreviates field of study. The type of frameworks refers to Figure 1. “Other images” in the **Source** column indicates other images of the mini-batch.

Model	FOS	Framework	Encoder	Pseudo-labels	Loss	Negative samples	
						Source	Strategy
TCN embedding [25]	CV	(d)	Inception network + CNN	Different but simultaneous viewpoint	Triplet loss	Images of different time	End-to-end
SimCLR [20]	CV	(d)	ResNet	Data augmentation	NT-Xent loss	Other images	End-to-end
SimCLR v2 [43] (semi)	CV	(d)	Variants of ResNet	Data augmentation	NT-Xent loss	Other images	End-to-end
MoCo [29]	CV	(d)	ResNet	Data augmentation	InfoNCE loss	Other images	Momentum
MoCo v2 [44]	CV	(d)	ResNet	Data augmentation	InfoNCE loss	Other images	Momentum
MoCo v3 [45]	CV	(d)	Vision Transformers	Data augmentation	InfoNCE loss	Other images	End-to-end
RotNet [46]	CV	(a)	ConvNet	Rotation directions	Prediction loss	-	-
Colorization [31]	CV	(a)	AlexNet, VGG-16, ResNet-152	Colour of missing patch	Regression loss KL divergence	- -	- -
DIM [39]	CV	(d)	-	-	JSD, DV or InfoNCE loss	-	End-to-end
Word2Vec [47]	NLP	(a)	Auto-encoder	Context words	Prediction loss	-	-
Speech2Vec [48]	Audio	(a)	Auto-encoder	Context audio	Prediction loss	-	-
Audio2Vec [49]	Audio	(a)	Auto-encoder	Context audio	Prediction loss	-	-
BERT [50]	NLP	(a)	MPC	Masked words	Prediction loss	-	-
ALBERT [51]	NLP	(a)	MPC	Masked words Sentence order	Prediction loss	-	-
NPC [52]	Audio	(a)	MPC	Masked frames	Prediction loss	-	-
BYOL [53]	CV	(b)	ResNet	Data augmentation	MSE loss	-	-
Barlow Twins [54]	CV	(b)	ResNet	Data augmentation	Eq. (3)	-	-
SimSiam [55]	CV	(b)	ResNet	Data augmentation	Negative cosine similarity	-	-
DeepCluster [56]	CV	(c)	AlexNet, VGG-16	Clustering centroids	Negative log-softmax loss	-	-
Local Aggregation [57]	CV	(c)	AlexNet, VGG-16	Soft-clustering centroids	Negative log-softmax loss	-	-
SwAV [58]	CV	(c)	Variants of ResNet-50	Online-clustering centroids	Modified cross-entropy	-	End-to-end
CPC [40]	CV, audio NLP	(d)	APC	-	InfoNCE loss	Other images	End-to-end
CPC v2 [59]	CV	(d)	APC	-	InfoNCE loss	Other images	End-to-end

as it determines the representation reliability. When setting a too large latent dimensionality, an auto-encoder risks to learn an identity function, i. e., maps the input directly to the output, and hence becomes useless. Various techniques to prevent auto-encoders from learning an identity function do exist, e. g., denoising auto-encoders [61], which partially corrupt the input data by randomly zeroing some input values and are trained to recover the original undistorted input. For the denoising to be successful, the model’s ability to retrieve useful high-level representations becomes essential. The zero-out step can be replaced by other data augmentation techniques, such as geometric transformations including cropping, reordering, and colourisation, to name a few, which often appear in SSL studies.

Nevertheless, such models do not always require to predict the entire original sample, i. e., the prediction can be restricted to only recover the distorted part. This is typically the case for sequential data as in Word2Vec [47], [62] that is used to map one-hot representations of words to word embeddings. In Word2Vec, two formulations are used to learn underlying

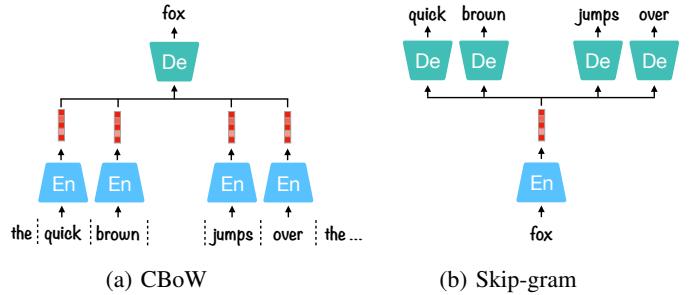


Fig. 2: Two architectures of Word2Vec. (a) CBoW predicts a single word from the previous and future words. The context words are fed into an encoder (En) to aggregate a context vector, which is used to produce the target word using a decoder (De); (b) Skip-gram makes the opposite prediction from CBoW, i. e., predicting previous and future words from a single centre word.

word representations: Continuous Bag-of Words (CBoW) and Skip-gram, depicted in Figure 2a and 2b, respectively. CBoW

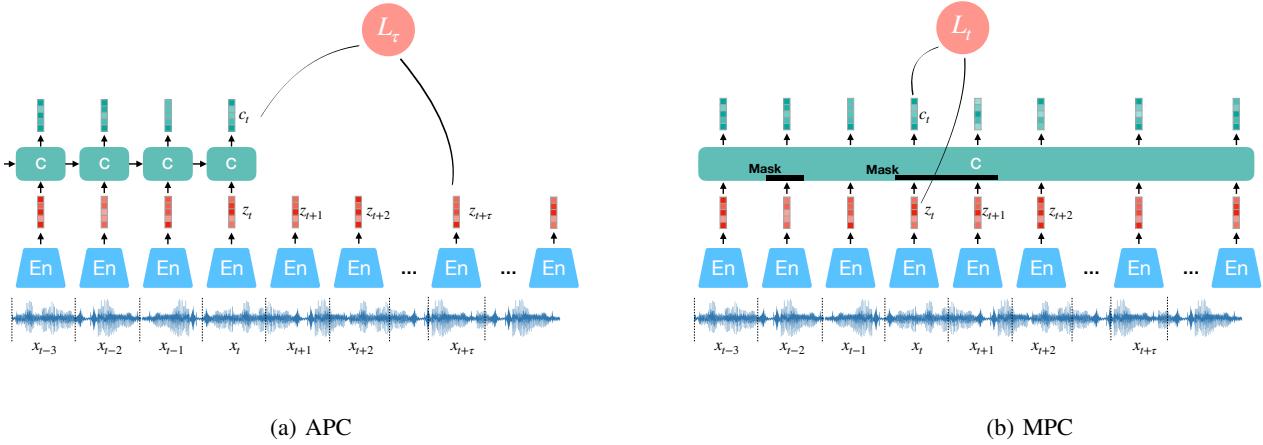


Fig. 3: Diagrams of auto-regressive predictive coding (APC) and masked predictive coding (MPC).

is trained to predict a single word from its context words, whereas Skip-gram does the opposite, aiming at predicting the left and right context words of a single input word. CBoW performs better in learning syntactic relationships between words, however, it is prone to overfit frequent words. Differently, Skip-grams are better at capturing semantic relationships and suffer less from overfitting, leading to a more effective solution in learning representations for general purposes. Both formulations have been considered for learning fixed-length vector representations of audio segments, such as in Audio2Vec [49] and Speech2Vec [48]. To this end, the input audio or speech waveform is transformed into its time and Mel-frequency representations, keeping a two-dimensional input format as for Word2Vec.

The success of Word2Vec or alike is based on the consistency of the context surrounding the component to predict. Similarly, an auto-regressive model can be used to learn representations by making predictions of future information conditioning on the past context. Auto-regressive Predictive Coding (APC) [63] codes on wave samples (cf. Figure 3a). An additional context network aggregates the resulting representations up to the current time step. Hence, the context network is usually a recurrent neural network (RNN) for modelling the temporal information. Its output context vector is then used to predict the next audio representations, for example, τ steps ahead of the current time step. APC is optimised by minimising L1 loss. The APC method takes into account only uni-directional information of a sequence and needs a combination of separately trained models for forward and backward directions in order to achieve a representation from both directions (past and future). Inspired by BERT [50], Masked Predictive Coding (MPC) trains directly a bidirectional architecture. This is made by masking parts of the input signals, which are subsequently predicted by conditioning on the context from both directions (cf. Figure 3b). Transformer encoders and bidirectional RNNs have been considered as context networks for realising MPC. Similarly, the recently proposed Non-autoregressive predictive coding (NPC) [52] also applies a mask on its model input, but it learns representations based on local dependencies of an input sequence,

rather than globally. The MPC approaches can learn effective representations of sequential data in a non-autoregressive way, and hence achieve considerable speed-up in training. Finally, other auto-encoding predictive models for SSL also aim to predict the relative position of signal parts [46], [64], including solving a Jigsaw puzzle [65], [66] or reordering the pieces of a shuffled sequence input [51], [67], [68].

Siamese Models have a typical ‘two towers’ architecture as shown in Figure 1b. Each tower processes a view of a data sample. Considering the natural similarity between the two views of the same sample, the encoded representations in the high-dimensional latent space should be close to each other. Hence, during training, the representations from one tower can be seen as the training target, i. e., pseudo-labels, for the other tower. The neural encoders of both towers share the same or similar architecture – their parameters can be shared or independent. When taking negative samples in the training objectives, a contrastive loss is formulated which pulls the representations of different views from the same data close together, while pushing the one from negative samples far away. The model can be optimised by applying standard back-propagation.

However, without using negative samples, the Siamese model is prone to *mode collapse*, i. e., when the model’s output is very similar (or even identical) for different inputs. Therefore, a specific training strategy is required for developing this kind of predictive model. Although the research of Siamese models starts from using contrastive learning as in SimCLR [20] and MoCo [29], recent works, such as BYOL [69], SimSiam [55], and Barlow Twins [54], use only positive pairs without contrasting negative pairs. These approaches still maintain competitive (and even superior) SSL performance. Thus, in this part, we will focus on introducing the Siamese models trained without negative samples, while the contrastive model will be discussed in Section II-B2.

BYOL [69], short for Bootstrap Your Own Latent, trains two sub-networks separately denoted as online network and target network as shown in Figure 4a. Both sub-networks contain an encoder f and a projection layer g . The online network has an additional predictor layer p build on MLP and

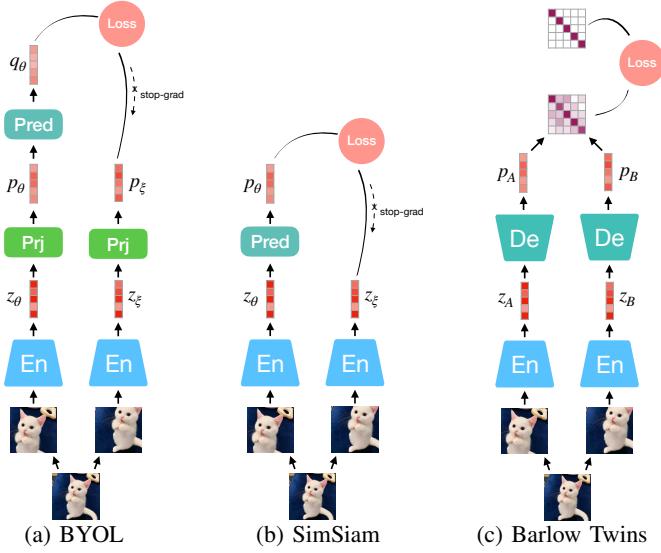


Fig. 4: Diagrams for Predictive Models using Siamese architectures.

is optimised to pull close the distance between the predicted representation and the projected embedding from the target network. To get rid of mode collapse, the two networks are asynchronously optimised in an iterative way. The target network is randomly initialised and then, its parameters are updated using an *exponential moving average* (EMA) strategy during training, similar as presented in MoCo [29] and defined as:

$$\xi \leftarrow \tau\xi + (1 - \tau)\theta, \quad (1)$$

where θ and ξ stand for the parameters of the online and target network, and $\tau \in [0, 1]$ is a given decay rate for updating. The online network follows the guidance of the slowly-updated target network, and is optimised by minimising the Mean Square Error (MSE) between the two network outputs:

$$\begin{aligned} L &= \|\bar{q}_\theta - \bar{p}_\xi\|_2^2 \\ &= 2 - 2 \frac{\langle q_\theta, p_\xi \rangle}{\|q_\theta\|_2 \cdot \|p_\xi\|_2}, \end{aligned} \quad (2)$$

where \bar{q}_θ and \bar{p}_ξ are l2-normalised q_θ and p_ξ , i.e., $q_\theta/\|q_\theta\|_2$ and $p_\xi/\|p_\xi\|_2$. The two views are exchanged as the input of online and target networks once to create a symmetric loss, denoted as \tilde{L} , leading to a complete training loss of $L + \tilde{L}$. The slow update of the target network progressively aggregates the parameters from the online network. This enables to produce more stable representations, which are used as the guidance to train the online network, by this, progressively yielding better representations. As updating the online parameters is a sensitive procedure that requires very careful fine-tuning, in order to avoid mode collapse, the authors additionally exploit LARS [70] as optimiser for training. Updating the parameters in different layers and with different strength is expected to gradually lead to meaningful convergence.

SimSiam [55] (shown in Figure 4b) shares a similar structure as BYOL, but is removing the projection layers in both networks. Its target network is also not optimised using back-propagation, but copied directly from the online network.

Updating the target network, also found to be effective, avoids the use of momentum encoding in BYOL.

The extra learnable predictor and a stop-gradient operation are considered as the essential parts that prevent the model from collapsing into trivial representations [71]. The theoretical analysis and experimental study in [71] has also revealed that as long as the predictor is updated more often or has a larger learning rate, EMA is not necessary for successful convergence without mode collapse. However, optimising the predictor too often or with too large a learning rate leads to its failure for SimSiam [55]. The final predictor is expected to be optimal, i.e., achieving minimal $l2$ error in predicting the output of the target network from the online network output. Moreover, weight decay has been shown to be very helpful in achieving stable convergence. Although the use of batch normalisation [72] was hypothesised to be crucial for preventing collapse in BYOL¹, in previous work [53] batch normalisation has been successfully replaced with group normalisation and weight standardisation, thus, refuting the need of batch statistics for BYOL.

Barlow Twins (BT) [54], is a neural network architecture inspired by the redundancy reduction principle described in the work of the neuroscientist H. Barlow [73]. As depicted in Figure 4c, BT contains two identical networks that process two distorted versions of a same sample to produce their representations. The model measures the cross-correlation matrix between the two learnt representations, which is expected to be close to the identity matrix. BT simplifies the training procedure with respect to BYOL and SimSiam, which requires asymmetric components, such as a predictor layer, as well as operations, including gradient stopping and EMA. A BT model benefits from very high-dimensional output vectors and its loss function is formulated as:

$$L = \underbrace{\sum_i (1 - C_{ii})^2}_{\text{invariance term}} + \lambda \underbrace{\sum_i \sum_{j \neq i} C_{ij}^2}_{\text{redundancy reduction term}}, \quad (3)$$

where the cross-correlation matrix computed between the outputs of the two networks along batch direction, is defined as:

$$C_{ij} = \frac{\sum_b p_{b,i}^A p_{b,j}^B}{\sqrt{\sum_b (p_{b,i}^A)^2} \sqrt{\sum_b (p_{b,j}^B)^2}}. \quad (4)$$

By minimising the training objective, the invariance term pushes the diagonal elements of the correlation matrix to 1, which makes the learnt representations of the two distorted versions of a sample as close as possible. The redundancy reduction term compresses the correlations between the off-diagonal elements of the correlation matrix. This reduction of the redundancy between output elements in a representation vector results in representations of sufficient disentanglement.

Clustering, such as K-means clustering [56], [74], provides a way to yield pseudo-labels for SSL. Considering that different objects are naturally associated with distinct categories, each category should occupy a separate manifold in the

¹<https://generallyintelligent.ai/blog/2020-08-24-understanding-self-supervised-contrastive-learning/>

representation space. Deep Cluster [56], as shown in Figure 1c, performs two steps iteratively. First, it exploits the K-means clustering method to group the encoded representations and produce pseudo-labels for each sample. Then, with the created pseudo-labels assigned to each sample, the encoder network can be optimised by minimising the classification loss, such as by cross-entropy loss. Instead of the global clustering method of K-means clustering, Local Aggregation (LA) [57] allows to model more flexible statistical structures by separately identifying neighbours for each example. Moreover, LA proposes an objective function that directly optimises a local soft-clustering metric, leading to better training efficiency. Another clustering method used in SSL is SwAV [58], which introduces online clustering ideas into a Siamese architecture, by this avoiding the time consumption due to the two-steps training paradigm. The online clustering assignment provides pseudo-labels within mini-batches. In addition, the authors introduce multi-crop augmentation, aiming at solving a swapped prediction problem.

2) Contrastive Models: Considering negative samples when training an SSL model, such as a Siamese model, assists in achieving representations of better distinctiveness. The key idea behind is to pull the representations of two similar inputs (defined as a positive pair) close in the latent space, and to push those of dissimilar inputs (defined as a negative pair) far. Hence, extensive efforts are put on the designing of contrastive losses. Hereby, we summarise the most popular types of contrastive loss before going through the contrastive SSL methods in the literature.

Contrastive loss In early versions of contrastive loss, an anchor data is paired with only one positive and one negative sample, leading to a positive pair and a negative pair. Recent training objectives take also into consideration multiple positive and negative pairs in one batch. Max margin contrastive loss, designed for deep metric learning [75], takes a pair of inputs and minimises the embedding distance when they are from the same class and maximises it, otherwise. More formally, given a data point x in a batch of samples \mathcal{X} , CL learns an encoder f that minimises

$$\begin{aligned} L(x, x^+) &= \sum_{x \in \mathcal{X}} \|f(x) - f(x^+)\|_2^2 \\ L(x, x^-) &= \sum_{x \in \mathcal{X}} \max(0, \epsilon - \|f(x) - f(x^-)\|_2)^2, \end{aligned} \quad (5)$$

where x^+ is a data point similar or congruent to x , and therefore is referred to as a positive sample. On the other hand, x^- denotes a data point dissimilar to x , and hence is referred to as a negative sample. In addition, ϵ is a hyperparameter that defines the lower bound distance between representations of different samples.

Triplet loss, proposed for face recognition [76], configures the offset distance between representations of positive and negative pairs:

$$\begin{aligned} L(x, x^+, x^-) &= \sum_{x \in \mathcal{X}} \max(0, \|f(x) - f(x^+)\|_2^2 - \|f(x) - f(x^-)\|_2^2 + \epsilon). \end{aligned} \quad (6)$$

Multi-Class N-pair loss [77] generalises the triplet loss allowing joint comparisons among multiple negative samples. It is formulated similarly to the softmax loss:

$$\begin{aligned} L(x, x^+, x^-_{n \in [1, 2N-1]}) &= \log\left(1 + \sum_{n=1}^{2N-1} e^{f(x)^T f(x_n^-) - f(x)^T f(x^+)}\right) \\ &= -\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{n=1}^{2N-1} e^{f(x)^T f(x_n^-)}}. \end{aligned} \quad (7)$$

NT-Xent, short for normalised temperature-scaled cross-entropy loss, introduces an additional temperature parameter for controlling the penalty on the effect of negative samples, similarly as ϵ in Equation (5) and Equation (6):

$$L(x, x^+, x^-_{n \in [1, 2N-1]}) = -\log \frac{e^{f(x)^T f(x^+)/\tau}}{\sum_{n=1}^{2N-1} e^{f(x)^T f(x_n^-)/\tau}}. \quad (8)$$

The loss function, also known as InfoNCE [40] objective, is inspired by Noise Contrastive Estimation (NCE) [78], which is proposed in order to estimate the parameters of a statistical model. InfoNCE generalises by contrasting the distance of representations from a positive pair to their distance to $N - 1$ negative pairs:

$$\begin{aligned} L(x, x^+, x^-_{n \in [1, N-1]}) &= \mathbb{E}\left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{n=1}^{N-1} e^{f(x)^T f(x_n^-)}}\right]. \end{aligned} \quad (9)$$

Its denominator terms contain one positive and $N - 1$ negative samples. Hence, we can construct a softmax classifier which is optimised using cross-entropy loss for N classes. The classifier assigns large and small values to the positive and negative examples, respectively. In this regard, InfoNCE can be seen as using categorical cross-entropy loss for identifying a positive sample within a set of (unrelated) noise samples.

Contrastive SSL for Siamese Models A typical architecture that is trained by using contrastive loss is SimCLR [20]. SimCLR exploits several different data augmentation techniques for transforming an input image, including random cropping, resizing, colour distortions, and Gaussian blur. The transformed images are then coded into representations using ResNet. After going through projection heads built on Dense-ReLU-Dense structure, NT-Xnet is used as objective function for self-supervised learning. The authors of SimCLR emphasised the importance of ‘scaling up’, i.e., using a larger batch size, a deeper and wider network, as well as training for longer epochs, in order to guarantee the success of the method.

Unlike SimCLR, which uses only one encoder f , Momentum Contrast (MoCo) [29] exploits an additional momentum encoder f_m . The encoder and momentum encoder, sharing the same architecture and being identically initialised, process two views of an image. The method also applies contrastive loss, where the negative samples are provided by previous batches. For this, representations of previous samples are stored into a queue during training. Representations from a new batch are pushed into the queue after training and old representations are excluded. The encoder is updated by applying back-propagation as in SimCLR, while the momentum encoder

is updated by linear interpolation of the two encoders as introduced in Equation (1). The momentum parameter is set to $\xi = 0.999$ by default, meaning the update of the momentum encoder is much slower. However, the update mode of the momentum encoder avoids back-propagation, which hence can increase the amount of negative samples for training. The synchronise update of the encoder and momentum encoder also solves the problem of inconsistency of encoded representations happening in works using memory bank [29]. Posterior architectures such as MoCo v2 [44] integrate the effective components presented in SimCLR. MoCo v2 incorporates stronger data augmentation techniques, i.e., using an additional Gaussian Deblur method and a larger batch size. Moreover, its projection head layer is increased as a 2-layers MLP for both the encoder and momentum encoder. Similarly, SimCLR v2 [43] upgrades the system proposed in SimCLR by scaling up the model size from ResNet-50 to ResNet-152 and improving the depth of the projection head. In addition, the authors leave one projection layer for fine-tuning on semi-supervised tasks, by this aiming at learning from few labelled examples while making best use of a large amount of unlabelled data. Furthermore, in order to efficiently provide a large size of negative samples for training, the idea of memory mechanism used in MoCo v2 is employed in SimCLR v2 too. Differently, the latest MoCo v3 [45], removes the memory queue with the cost of requiring a bigger batch size. In addition, it applies a prediction layer after the projection head, similarly as proposed in BYOL, which further improves the representation capability.

In contrastive SSL, given one data point, the learnt representations of its positive and negative samples are seen as able to provide opposite pseudo-labels. A positive sample provides an additional view of the same data. Theoretical analysis has proven that when two views provide redundant information of the label, applying linear projections to the learnt representations can guarantee the performance on downstream prediction tasks [19]. This proof indicates that SSL can produce high-quality representations from the multi-views of a data point and guarantees prediction performance with simple downstream models.

Contrasting the distance of a data point to its positive samples with respect to the one to its negative samples avoids the model from falling into representational collapse. To analyse the effectiveness of contrastive loss, we can split it into two parts:

$$\begin{aligned} L &= \mathbb{E}[-\log \frac{e^{f(x)^T f(x^+)/\tau}}{e^{f(x)^T f(x^+)/\tau} + \sum_{n=1}^{N-1} e^{f(x)^T f(x_n^-)/\tau}}] \\ &= \underbrace{\mathbb{E}[-f(x)^T f(x^+)/\tau]}_{\text{alignment}} \\ &\quad + \underbrace{\mathbb{E}[\log(e^{f(x)^T f(x^+)/\tau} + \sum_{n=1}^{N-1} e^{f(x)^T f(x_n^-)/\tau})]}_{\text{uniformity}}, \end{aligned} \quad (10)$$

where the ‘alignment’ term targets at maximising the similarity between the learnt embeddings of the positive pairs. Then, the ‘uniformity’ term helps the contrastive learning to learn

separable features by maximal-uniformly distributing the embeddings on a unit sphere given the normalisation condition. Both terms are crucial to the downstream tasks according to [79]. Different from an instance discrimination objective, which pushes all different instances apart and considers no underlying relations between samples, the design of contrastive loss requires a proper temperature coefficient τ that finds a balance between learning separable features and at the same time provides some degree of tolerance to the closeness of semantically similar samples. A too small τ loses the tolerance to group the similar input samples and hence may break the underlying semantic structure, by this harming the learnt features for its use in downstream tasks. The effect of the temperature parameter is similar as the margin value set in Equation (5), which has been investigated in detail by [80].

In [15], Wang suggests to adjust the alignment and uniformity loss to

$$\begin{aligned} L_{\text{align}} &= \mathbb{E}[||f(x) - f(x^+)||_2^\alpha] \\ L_{\text{uniform}} &= \log \mathbb{E}[e^{-t||f(x) - f(x^*)||_2^2}], \end{aligned} \quad (11)$$

indicating that both terms should be minimised simultaneously. Maintaining a good balance between these two losses has been found more effective than standard contrastive loss.

3) Contrastive Predictive Coding: Contrastive Predictive Coding (CPC) [40], exploits an auto-regressive predictive model, and is optimised to predict the correct future information based on the aggregated global context from the past frames. In addition, CPC gets use of negative samples to improve the representation discrimination in training objectives such as InfoNCE loss Equation (9), replacing the query data point x by a context vector c_t :

$$L(c_t, z_{t+\tau}, z_{n \in [1, N-1]}^-) = \mathbb{E}[-\log \frac{e^{c_t^T z_{t+\tau}}}{e^{c_t^T z_{t+\tau}} + \sum_{n=1}^{N-1} e^{c_t^T z_n^-}}], \quad (12)$$

where z_n^- denotes a negative data point sampled from the proposal distribution of $z_{t+\tau}$, i.e., randomly sampled from the sequence z . In CPC v2 [59], the model is scaled up to achieve larger model capacity, and the batch normalisation is replaced by layer normalisation, as batch normalisation is found to harm downstream tasks for CPC frameworks [59]. Moreover, patch-based augmentation is introduced in order to add more diversity to the model’s input.

Similarly, the contrastive loss has been used in masked predictive models, leading to contrastive MPC. The training objective can be formalised as

$$L(c_t, z_t, z_{n \in [1, N-1]}^-) = \mathbb{E}[-\log \frac{e^{c_t^T z_t}}{e^{c_t^T z_t} + \sum_{n=1}^{N-1} e^{c_t^T z_n^-}}]. \quad (13)$$

This loss has been modified and used in wav2vec 2.0 [81], which will be introduced in details Section III. The negative samples are generated in the same way as for CPC.

C. Training with or without negative samples

Recent works have experimentally tested the importance of components for achieving effective SSL models [82]. The training objectives have been shown to be more important

than the network architecture, and the quality of the learnt representations can be improved by scaling up the model size and the representation size. Furthermore, the quantity and quality of the negative samples has also shown to be important for the performance of SSL using contrastive learning [83]. The SSL models optimised without contrastive loss, such as BYOL [69], Barlow Twins [54], SimSiam [55], and the other previously introduced auto-encoder-like predictive methods, aim at reducing the distance between the latent representations of views from the same data point. Contrastive SSL aims at contrasting the distance between positive samples against the distance to negative ones. The advantages and disadvantages of these two training strategies can be traced back to the difference between generative and discriminative models in the wide field of machine learning methods. Training SSL without negative samples may be less effective in learning discriminative features between samples, but has more potential to code more complete information into representations. Differently, contrastive SSL approaches are expected to learn more discriminative features by comparing to negative samples at expenses of dropping the common attributes, which are salient to represent the sample itself but are not very informative for distinguishing between samples. Based on this, we infer that applying SSL without negative samples is more useful, as it would do a feature extractor without losing much salient information. Contrastive SSL can perform better if a pretext task and the downstream task fit closer, for example, if the data for the pretext task contains reading speech and the downstream task aims at speech recognition. However, as the parameters of a pre-trained model can be further fine-tuned for a downstream task in a specific domain, an SSL model optimised with negative samples can be improved to complement its generalisation ability to capture more complete representations. Furthermore, a contrastive SSL model can also be further improved to reduce the redundant information in the representations for the downstream tasks. Still, reducing redundancy in representations for downstream tasks seems to be harder than restoring the lost representation completeness, which is confirmed by the fact that non-contrastive SSL outperforms contrastive SSL, as shown in recent works [55], [69].

A summary table of the typical SSL methods is shown in Table I.

III. AUDIO SSL

Most of the above introduced SSL methods have been transferred from those aiming to solve audio tasks, especially speech applications such as Automatic Speech Recognition (ASR) [40], [81], [84]–[87]. Other methods, which differ in the used audio input formats, such as LIM [36], COLA [37], CLAR [33], and the work by Fonseca *et al.* [38], expand the SimCLR approach for learning auditory representations. The LIM model [36] processes directly speech samples expecting to maximise local mutual information between the encoded representations of chunks of speech sampled from the same utterance. In COLA [37] and the work by Fonseca *et al.* [38], the presented models take segments randomly extracted from

time-frequency features along the temporal direction. Several stochastic data augmentations are adopted for the patches before feeding to the model in [38], such as random size cropping and Gaussian noise addition. In addition, the authors proposed *mix-back* for additional augmentation, which mixes the incoming patch with a background patch, by this ensuring that the incoming patch is dominant in their mixture. In CLAR [33], the paired views of the model’s input are generated by applying data augmentations on raw audio signals and time-frequency audio features, for which effective compositions of data augmentation are explored. In addition, the authors found that combining a contrastive loss, such as cross-entropy loss, in training objectives for supervised learning, while using substantially less labelled data, can provide significant improvements in terms of convergence speed and representation effectiveness, with respect to using SSL only. Similarly, Wang [88] also suggests to train audio SSL models with different formats of an audio sample. More precisely, the training objective is to maximise the agreement between the raw waveform and its spectral representation. The approach is shown effective for downstream classification tasks on both AudioSet, and ESC-50 datasets. BYOL has also been adopted in the audio domain, named as BYOL-A [89], which learns representations from a single audio without using negative samples.

Other typical works include Audio2Vec [49] and Speech2Vec [48], inspired by Word2Vec [47], as introduced in Section II-B1. Both works learn audio representations using CBoW and skip-gram formulations. In the CBoW formulation, the task is to reconstruct a temporal spectrogram slice of pre-determined duration from a number of past and future slices. The method has also been shown effective for acoustic scene classification in [90]. Differently, the roles of the target and surrounding slices are reversed in the Skip-gram formulation. Audio2Vec and Speech2Vec mainly differ in the following aspects: (i) Speech2Vec applies audio segmentation, by using an explicit forced alignment technique, in order to isolate audio slices corresponding to each word. The forced alignment segmentation may introduce supervision to some extent; (ii) Audio2Vec requires no explicit assistance and hence completely removes the need of supervision; (iii) unlike neural network architectures, Speech2Vec is built based on an RNN encoder-decoder, and Audio2Vec is built of stacks of CNN blocks; (iv) as model input, Speech2Vec processes the Mel-spectrogram of an audio, while Audio2Vec operates on Mel-Frequency Cepstral Coefficients (MFCCs); and (v) in Audio2Vec, the TemporalGap formulation is additionally introduced, which requests that the model estimates the absolute time distance between two (randomly sampled) slices, taken from the same audio clip.

Although TemporalGap fails to surpass the CBoW and Skip-gram approaches, it presents the idea of measuring the relative positions of audio components as a pretext task. Carr *et al.* [67] proposed a training strategy based on permutations, i. e., training a model that can reorder shuffled patches of an audio spectrogram, as analogous to solving a jigsaw puzzle [65]. The method draws inspiration from ‘Shuffle and Learn’ [100] and has also been considered in another work for industrial

TABLE II: An overview of the recent audio self-supervised learning methods. The “speech” column distinguishes whether a method addresses speech tasks or for general purpose audio representations. The “framework” type refers to Figure 1.

Model	Speech	Input format	Framework	Encoder	Loss	Inspired by
LIM [36]	✓	raw waveform	(d)	SincNet	BCE, MINE or NCE loss	SimCLR
COLA [36]	✗	log mel-filterbanks	(d)	EfficientNet	InfoNCE loss	SimCLR
CLAR [33] (semi)	✗	raw waveform log mel-spectrogram	(d)	1D ResNet-18 ResNet-18	NT-Xent + cross-entropy	SimCLR
Fonseca et al. [36]	✗	log mel-spectrogram	(d)	ResNet, VGG, CRNN	NT-Xent loss	SimCLR
Wang et al. [88]	✗	raw waveform + log mel-filterbanks	(d)	CNN ResNet	NT-Xent loss + cross-entropy	SimCLR
BYOL-A [89]	✗	log mel-filterbanks	(b)	CNN	MSE loss	BYOL
Speech2Vec [48]	✓	mel-spectrogram	(a)	RNN	MSE loss	Word2Vec
Audio2Vec [91]	✓✗	MFCCs	(a)	CNN	MSE loss	Word2Vec
Carr [67]	✓	MFCCs	(a)	Context-free network	Fenchel-Young loss	-
Ryan [68]	✗	constant-Q transform spectrogram	(a)	AlexNet	Triplet loss	-
Mockingjay [92]	✓	mel-spectrogram	(a)	Transformer	L1 loss	BERT
TERA [93]	✓	log mel-spectrogram	(a)	Transformer	L1 loss	BERT
Audio ALBERT [94]	✓	log mel-spectrogram	(a)	Transformer	L1 loss	BERT
DAPC [95]	✓	spectrogram	(a)	Transformer	Modified MSE loss + orthogonality penalty	BERT
PASE [96]	✓	raw waveform	(a)	SincNet + CNN	L1, BCE loss	BERT
PASE+ [97]	✓	raw waveform	(a)	SincNet + CNN + QRNN	MSE, BCE loss	BERT
CPC [40]	✓	raw waveform	(a)	ResNet + GRU	InfoNCE loss	-
CPC v2 [59]	✓	raw waveform	(a)	ResNet + Masked CNN	InfoNCE loss	-
CPC2 [98]	✓	raw waveform	(a)	ResNet + LSTM	InfoNCE loss	-
Wav2Vec [84]	✓	raw waveform	(a)	1D CNN	Contrastive loss	-
VQ-Wav2Vec [85]	✓	raw waveform	(a)	1D CNN + BERT	Contrastive loss	BERT
Wav2Vec 2.0 [81]	✓	raw waveform	(a)	1D CNN + Transformer	Contrastive loss	BERT
HuBERT [99]	✓	raw waveform	(c)	1D CNN + Transformer	Contrastive loss	BERT

audio classification [68]. In [67], Carr et al. also leverage differentiable ranking to integrate permutation inversions into an end-to-end training, which enables solving the permutation inversion for the whole set of permutations, i.e., reducing the space of permutations that might be exploited and performing the reordering as a classification task.

Another predictive model using an auto-encoder [92]–[94], as shown in Figure 3b, exploits a masked acoustic model (MAM) that masks some parts of an audio input and reconstructs the entire original input, essentially to fill the masked parts that are not known by MAM during training. The model is optimised by minimising the reconstruction error for learning general audio representations. Mockingjay [92] (Figure 5a) takes the Mel-spectrogram as input acoustic features and exploits transformers to code randomly masked frames into audio representations. The encoded representations are mapped to predict the complete frames using a projection head built of 2-layers MLP with layer normalisation. The transformer encoder and projection head are jointly optimised by minimising the L1 reconstruction loss. The effectiveness of self-attention in transformer encoders has been further explored in [101]; the authors also provide a visualisation tool for understanding the attention, based on which several attention refinement techniques are proposed to improve model performance. Audio ALBERT [94] has the same network architecture as Mockingjay, but the parameters are shared across all its transformer encoder layers, by this achieving a

faster inference and increasing training speed without harming the performance of two evaluation downstream tasks, i.e., speaker classification and phoneme classification. In TERA [93], –short for Transformer Encoder Representations from Alteration – the authors extend the used masking procedures, including replacing contiguous segments with randomness, masking along the channel axis, and applying Gaussian noise for pre-training the transformers. This resulted in a better representation performance than the one shown by Mockingjay and audio ALBERT, for the downstream tasks, phoneme classification, keyword spotting, and speaker classification [93]. In addition, it shows also promising results for ASR tasks based on the LibriSpeech and TIMIT data sets.

Unlike the works that predict the entire audio frames from their masked version, DAPC [95] (Figure 5b) proposes a method to only predict the missing components along the time- and frequency axes of an audio spectrogram by minimising a masked reconstruction loss. The method is also regarded as an extension of CBoW, for which the input masked spectrogram can be easily generated using SpecAugment [34], and hence, the missing parts to be predicted are not only temporal frames, but also frequency bins. The problem agnostic speech encoder (PASE, Figure 5c) [96] is another approach that combines a CNN encoder with multiple neural decoders, defined as workers in the literature. The workers, fed with learnt representations from the encoder, aim at solving regression or binary discrimination tasks. The regression tasks include, for

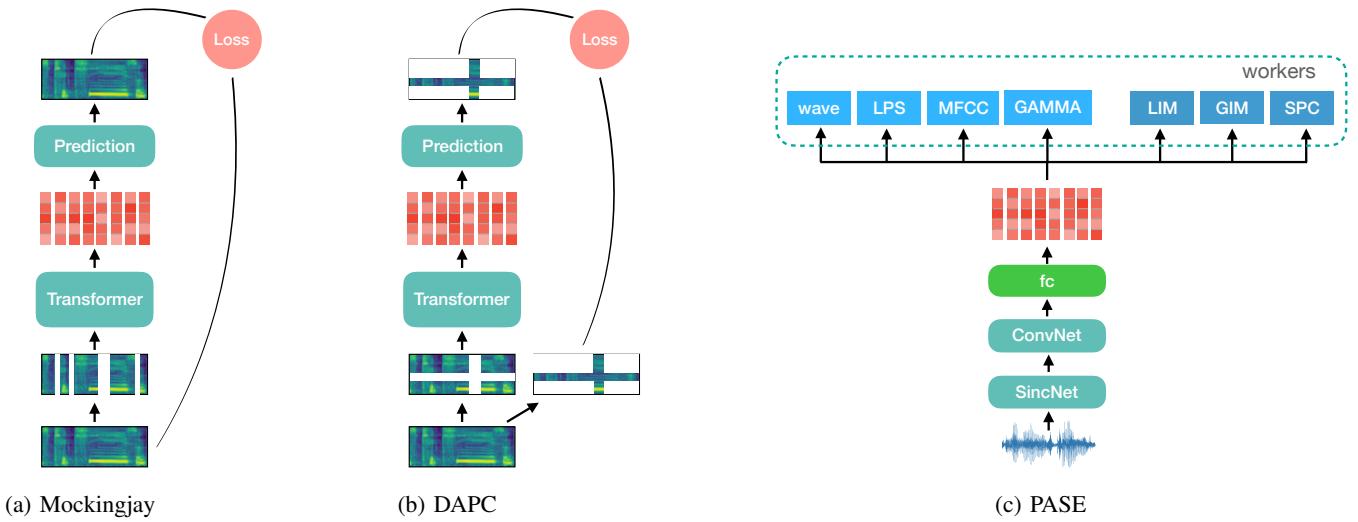


Fig. 5: Predictive models for audio SSL.

instance, recovering the raw audio waveform, the log power spectrogram, MFCCs, and prosody. The binary discrimination tasks applies contrastive learning, by this maximising local and global mutual information similar as in [36] and [39], and optimises sequence predicting coding similar to [102]. Each self-supervised task is expected to provide a different view of the speech signal; jointly solving self-supervised problems pushes the views into a unique representation that contains meaningful speech information such as speaker voice-print, phonemes, or emotions. In addition, to process the raw waveform as the encoder input, the SincNet [103] model is used as the first stage of PASE, which performs a convolution with a set of parametrised Sinc functions that implement rectangular band-pass filters. PASE+ [97] incorporates additional data augmentation techniques and more effective workers. The CNN encoder is combined with a Quasi-Recurrent Neural Network (QRNN) [104] for capturing long-term dependencies in sequential data in a more efficient way.

Van den Oord proposed CPC [40], which can effectively learn representations by predicting the future in a latent space using an auto-regressive model, showing very promising results for audio, images, text processing, and reinforcement learning. For audio, a strided convolutional network is used to encode raw audio to its latent representation. Then, a Gated Recurrent Unit (GRU) - RNN aggregates the information from all the past timesteps to form a context vector. More importantly, contrastive loss is applied to learn more discriminative representations by contrasting the true future to noise representations, given an aggregated context vector. Speech signals can be pre-processed by using a time-domain data augmentation library, such as WavAugment [98], in order to achieve more powerful representations by CPC. The library contains several DA techniques, including pitch modification, additive noise, reverberation, band reject filtering, or time masking, to name a few. In [98], the authors define a CPC2 model, which replaces the GRU-RNN of CPC by a two-layers LSTM-RNN and replaces the linear prediction network by a single multi-head transformer layer, leading to better training

efficiency without harming representation performance.

Wav2vec [84], as shown in Figure 6a, adjusts the CPC structure to a fully convolutional architecture, enabling easy parallelisation over time on hardware. One CNN is used to produce a representation from audio, and the other captures global context information into a context vector for each time step. Besides moving beyond phoneme-based ASR, explored in [40], it substantially improves a character-based ASR system. Specifically, the wav2vec approach is optimised by minimising contrastive loss for each step $k = 1, \dots, K$:

$$L_k = - \sum_{i=1}^{T-k} (\log\sigma(z_{i+k}^T h_k(c_i)) + \lambda \mathbb{E}[\log\sigma(-\tilde{z}^T h_k(c_i))]), \quad (14)$$

where $\sigma(x) = 1/(1 + \exp(-x))$, and $\sigma(z_{i+k}^T h_k(c_i))$ stands for the probability of z_{i+k} being the true future sample, applying an affine transformation, $h_k(c_i) = W_k c_i + b_k$. The total loss sums up considering K steps; $L = \sum_{k=1}^K L_k$ is minimised for training. After pre-training, the affine projection layer is removed for creating the learnt representations from the raw audio.

The works by Baevski et al. [85], [105] (Figure 6b) exploit a vector quantisation module after the wav2vec encoder in order to discretise the audio representations. This aims to find, for each representation, the closest embedding from a fixed size codebook $e \in \mathbb{R}^{V \times d}$ containing V representations of size d . The discrete representations are fed into the context network and then are optimised in the same way as for wav2vec. To solve the discontinuity caused by the argmax operation of the vector quantisation, Gumbel-Softmax [106] (refer to Figure 7a) as a differentiable approximation of the argmax for computing one-hot representations as well as online k-means clustering (refer to Figure 7b) are alternative solutions. This is similar to a vector-quantised variational auto-encoder (VQ-VAE) [102] and to vector-quantised autoregressive predictive coding [107].

Using a single codebook for coding representations tends to mode collapse in the cases in which only some of the codewords are actually used. To solve this issue, multiple

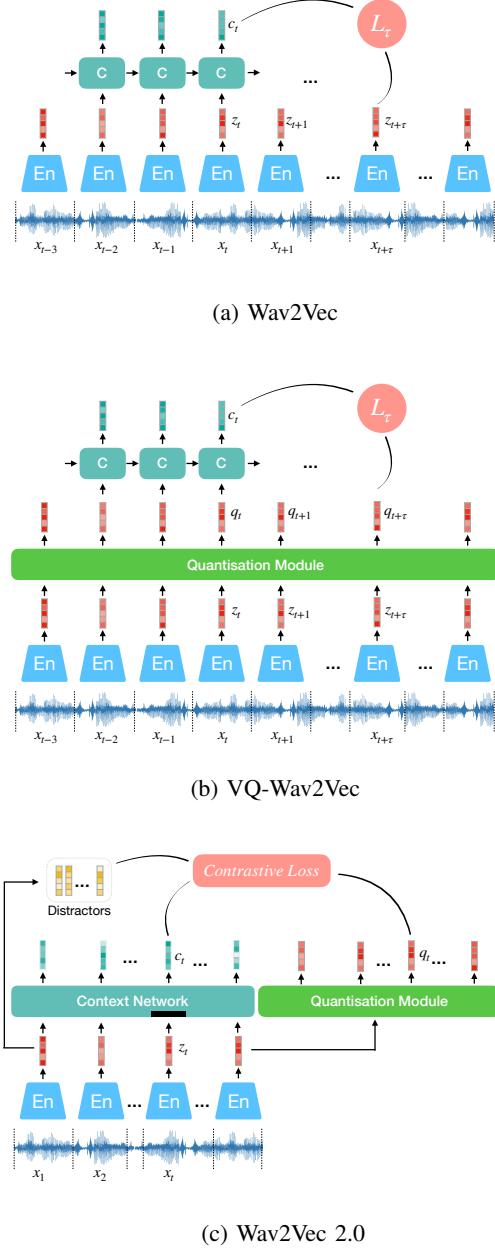


Fig. 6: Predictive models for audio SSL.

codebooks are used as in product quantisation [108]. Specifically, product quantisation is equivalent to choosing quantised representations from multiple codebooks and concatenating them. Given G codebooks with V entries $e \in \mathbb{R}^{V \times d/G}$, one entry from each codebook is selected. A linear transformation is applied after concatenating the selected codewords. The probabilities for choosing the v -th codebook entry for group g are

$$p_{g,v} = \frac{e^{(l_{g,v} + n_v)/\tau}}{\sum_{k=1}^V e^{(l_{g,k} + n_k)/\tau}}, \quad (15)$$

where $l \in \mathbb{R}^{G \times V}$ represent the logits from projecting the encoded dense representation, $n = -\log(-\log(u))$ and u are uniform samples from $U(0, 1)$, and τ is a non-negative

temperature parameter. The codeword i in group g is chosen by $\text{argmax}_i p_{g,i}$.

K-means clustering can also be used for differentiable vector quantisation. A codeword is selected as long as it has the closest distance to the dense representations z . For training the model, in this case, additional terms are added in the wav2vec objective function, leading to

$$L = \sum_k L_k + (\|\text{sg}(z) - q\|^2 + \gamma \|z - \text{sg}(q)\|^2), \quad (16)$$

where sg is the stop gradient operator and γ is a hyper-parameter. By minimising the loss, the term $\|\text{sg}(z) - q\|^2$ freezes the encoder output z and forces the codewords Q to be closer to the encoder output. The other additional term $\|z - \text{sg}(q)\|^2$ drives each encoder output to be close to one codeword, which is one centroid of the K-means clustering.

The discretised representations are then used for training a BERT model in [85], aiming to predict masked input tokens based on encoding the surrounding context. The resulting representations are then fed into an acoustic model for producing transcriptions.

The optimisation of Wav2Vec and VQ-Wav2Vec are motivated by CPC, processing audio input for only one forward direction. Instead, Wav2vec 2.0 exploits a bidirectional MPC model, optimised by using contrastive loss, such as InfoNCE [40]. The raw audio is encoded using multiple 1D-CNN layers, and the resulting representations are partly masked before sending to a transformer network to learn contextualised representations. The networks are jointly trained to contrast the true representations from distractors, given the contextualised representations. Similarly to VQ-Wav2Vec, Wav2vec 2.0 applies product quantisation too; however, the quantised vector q_t for each time step is not fed into a context network, but only used in the objective function:

$$L = \mathbb{E}[-\log \frac{e^{c_t^T q_t / \tau}}{\sum_{\tilde{q} \sim Q_t} e^{c_t^T \tilde{q} / \tau}}], \quad (17)$$

where $\tilde{q} \sim Q_t$ includes q_t and K distractors. In addition to InfoNCE, the training loss is regularised by a diversity loss L_d to encourage the model to use V codebook entries equally often. The diversity loss is formulated as

$$L_d = \frac{1}{GV} \sum_{g=1}^G -H(\bar{p}_g) = \frac{1}{GV} \sum_{g=1}^G \sum_{v=1}^V \bar{p}_{g,v} \log \bar{p}_{g,v}. \quad (18)$$

Wav2Vec 2.0 shows the state-of-the-art results for ASR tasks evaluating on both LibriSpeech [109] and TIMIT [110] data sets.

The method has been explored from the perspective of domain shift in [111], where the data for pre-training, fine-tuning, and evaluation originates from different domains. The authors conclude that the matching conditions between data of pre-training and testing are very important in order to achieve satisfying speech recognition results. Moreover, pre-training on multiple domains can improve the generalisation ability of the learnt representations.

Based on Wav2vec 2.0, models for cross-lingual speech representation learning have been explored separately in [112]

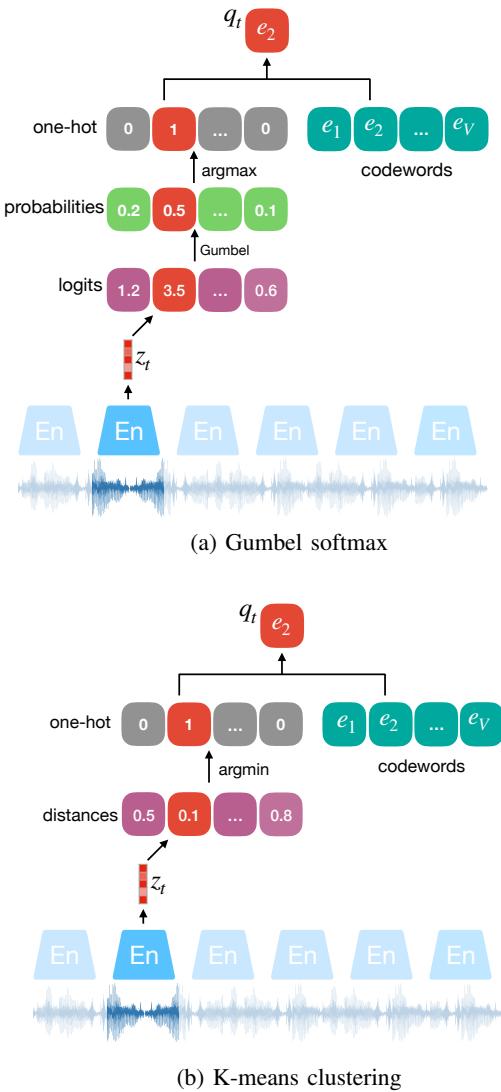


Fig. 7: Diagrams illustrating the mechanisms to select codewords from a codebook.

and [113]. The task of learning multilingual speech representations has also been undertaken in [114] but using a bidirectional CPC model. Sadhu *et al.* [115] proposed a way to integrate Wav2vec 2.0 and VQ-VAE [102] in a single model, defined as Wav2vec-C. The use of VQ-VAE, reconstructing the discrete codes to the original input features, can be seen as providing an additional regularisation when creating discrete speech representations by Wav2vec 2.0. The optimisation target combines the reconstruction error from VQ-VAE and the loss function of Wav2vec 2.0. This enforces the learnt latent representations, which explicitly carry essential information for recovering the input features. This is expected to alleviate the codebook utilisation difficulties observed for Wav2vec 2.0. The method achieved some additional improvement in ASR on real-world far-field noisy data, compared to the original Wav2vec 2.0 [81].

In general, all the introduced Wav2vec audio SSL models learn latent representations without considering specific tasks for pre-training. After pre-training, they are fine-tuned for

downstream tasks in an additional step. Baevski *et al.* present Wav2vec-U [116], short for Wav2vec Unsupervised, which learns a map from audio representations to phonemes directly without supervision. The method is a generative adversarial network, where the generator uses Wav2vec 2.0 to extract speech representations and generate phoneme sequence based on it using a clustering method; the generated phoneme tries to cheat a discriminator that is conditioned on a real phoneme sequence from unlabelled text.

The idea of grouping quantised audio representations into phoneme sequences is named as phonetic clustering in SeqRA-AE [87]. In this work, the discrete representation is learnt in an auto-encoder architecture with vector quantisation. Moreover, the consecutive repeated quantised representations are further grouped to form phonetic units. Each phoneme can therefore correspond to several repeated codewords, which is similar to the format of Connectionist Temporal Classification (CTC) [117]. Differently, Hidden unit BERT (HuBERT) [99] does not apply contrastive learning for training the same MPC model and avoids vector quantisation. Instead, each of the learnt audio representation is paired with a pseudo-label provided by applying K-means to MFCCs of the input audio. The method benefits from cluster ensembles, as the K-means clustering can be of different numbers of clustering centres, creating targets of different granularity.

We noticed that the classic formulation of several front-end audio processing tasks that have been explored for a long history are essentially using the framework audio SSL. For example, methods to solve the task of speech enhancement (SE) [118], [119] share similar structure with the auto-encoding predictive model, as shown in Figure 1a. An SE model processes a noisy audio input and outputs clean speech. For generating the noisy input, clean speech is typically mixed with a noise recording. This is exactly the same as the processing of input to an auto-encoding predictive model, while the noise addition is seen as a step for data augmentation. Hence, the latent features in the middle layers of an SE model are seen as a kind of audio representation of the clean speech. The formulation is not limited to speech enhancement, but is applicable to all the pre-processing tasks that aim at predicting audio of interest from additive and multiplicative noise or interference, such as, source separation, de-reverberation, and echo cancellation.

In some very recent works, audio SSL approaches have been specifically chosen to solve typically challenging tasks, such as speech enhancement [120]–[122] and source separation [123]. In [120], a pair of variational auto-encoders, named clean auto-encoder (CAE) and mixture autoencoder (MAE), were exploited. A CAE is trained to learn representations of clean speech by minimising the reconstruction error of its input spectrogram. An MAE encodes a noisy utterance and forces the encoded representation into the same latent space of the CAE by using a cycle-consistency loss terms. This paradigm leans a mapping from the domain of mixtures to the domain of clean sounds without using paired training examples. MixIT, which is short for Mixture Invariant Training, is proposed in [124] for solving unsupervised sound separation. In MixIT, a separation network takes a mixture of multiple single-channel acoustic mixtures (MOM) as model

input, where each of the acoustic mixtures is comprised of several speech sources. The separation network decomposes the MOM into separate audio sources, which are then selected to be re-mixed up to approximate each acoustic mixture of the MOM. Similarly as for the Permutation Invariant Training (PIT) [125], the remix matrix is optimised by choosing the best match between the separated sources and the acoustic mixtures. The method shows improvements for reverberant speech separation, universal sound separation, and is effective for speech enhancement, too. Finally, using denoising pre-training is an alternative solution to solve the permutation switching problem of source separation [123]. In this work, the authors use speech denoising as a self-supervised pre-training task to learn the structure information of speech from large-scale data. The model is subsequently fine-tuned with the normal training objective of source separation. As knowledge about the speech structure has been captured in the pre-trained model, it relaxes the permutation problem.

To develop an SE system specialised in a particular person (PSE), [121] presents two SSL algorithms, pseudo speech enhancement (PseudoSE) and contrastive mixtures (CM), for extracting speaker-specific discriminative features. A PseudoSE model is trained to recover a premixture signal (i.e., a clean speech contaminated by noise) from a pseudo-source (i.e., a mixup of the premixture signal and additional noise). The CM method generalises the training via contrastive learning, for which a positive pair shares the same premixture signal (but deformed with different additional noises), while a negative pair stems from two different premixture sources mixed with the same additional noise. The trained model, using either contrastive or non-contrastive SSL, is trained to recover premixture sources rather than clean speech, and hence, it requires fine-tuning for the downstream task. Data purification (DP) [126] is later introduced in the pseudo speech enhancement training. Specifically, a separate model is trained to estimate the segmental SNR of the premixture signals, measuring the different importance of the audio frames. Injecting the importance measurement in pseudo SE training enables the model to benefit from segments of higher quality, and hence, enables to derive more meaningful speaker-specific features.

A summary table of the typical audio SSL methods is shown in Table II.

IV. MULTI-MODAL AUDIO REPRESENTATION

The successful adoption of SSL has spread over many academic and industrial fields, including, but not limited to, CV, audio processing, and NLP, to cite a few. Moreover, exploiting multi-modal SSL has also been explored for a variety of applications whereby the representation learning of different modalities can be performed simultaneously. The mutual complementarity between different modalities, treated as different views, representing one unique object, is beneficial for the representation learning of each considered modality. In this section, we discuss multi-modal SSL approaches that use audio as one modality. Most of these works are based on audio-visual processing which aim, for example, at determining the correspondence between video frames and its audio

sequence. Other visual-audio methods, similarly as the SSL works previously described, make use of the synchronisation of the visual and audio streams of a video, taking the two views as input of a Siamese network. In this case, each modality of the two can be seen as the supervisory signal for the other. Audio representation can also be learnt during a task of video generation, where the representation of each segment of an utterance is expected to carry adequate speech information in order to transfer the knowledge into video frames. Some more interesting approaches are motivated by classic tasks in CV, i.e., object segmentation and localisation, and audio processing, i.e., source separation. On the other hand, text is also considered as one modality that assists in learning speech representations, regarding the similarity between speech and text in a linguistic structure.

A. Audio & Visual

1) Visual-Audio Correspondence Decision: By splitting a video into visual and audio streams, both L³-Net [127] (Figure 8a) and AVE-Net [128] (cf. Figure 8b) exploit two convolutional networks, named as vision sub-network and audio sub-network, to separately encode the two streams into a common space for cross-modal retrieval. Specifically, based on the alignment between both streams, one second of audio segment and the corresponding centre video frame are fed into these two networks. The model is required to decide whether the two inputs are in correspondence or not. For a video clip, the audio segment and video frame at the same time step are considered as a positive pair for model input, while a negative input pair is the audio segment paired with a video frame from another different video clip. In L³-Net [127], the audio and visual representations are concatenated before sending into fully-connected layers to predict the correspondence score. In contrast, AVE-Net [128] measures the correspondence degree by computing the Euclidean distance between audio and visual representations that are designed to be of the same dimensionality. Moreover, both L³-Net and AVE-Net are especially designed for recognising where the sound is generated in the video frame, for example, the location of specific instruments in a band. The vision sub-network of L³-Net has the intrinsic ability to recognise semantic entities that make sound, while AVE-Net needs additional modifications on its model architecture, incorporating a comparison mechanism to the audio representation with each spatial grid of the 3D visual representations (Figure 8c). The method encourages at least one region to respond highly for a corresponding audio and video frame, and hence enabling the localisation of the object that sounds. As these two AVC works formulate the task as a binary classification problem, the models can be optimised by minimising a logistic loss.

In [130], a model to predict whether the visual and an audio stream are synchronised is trained, with contrastive objectives, using a sequence of video frames instead of a single frame. Nagrani et al. [131] suggest to apply curriculum learning, i.e., starting the training with relative easier negative and positive pairs for good model initialisation, and gradually increase the difficulty of input pairs for easier model convergence.

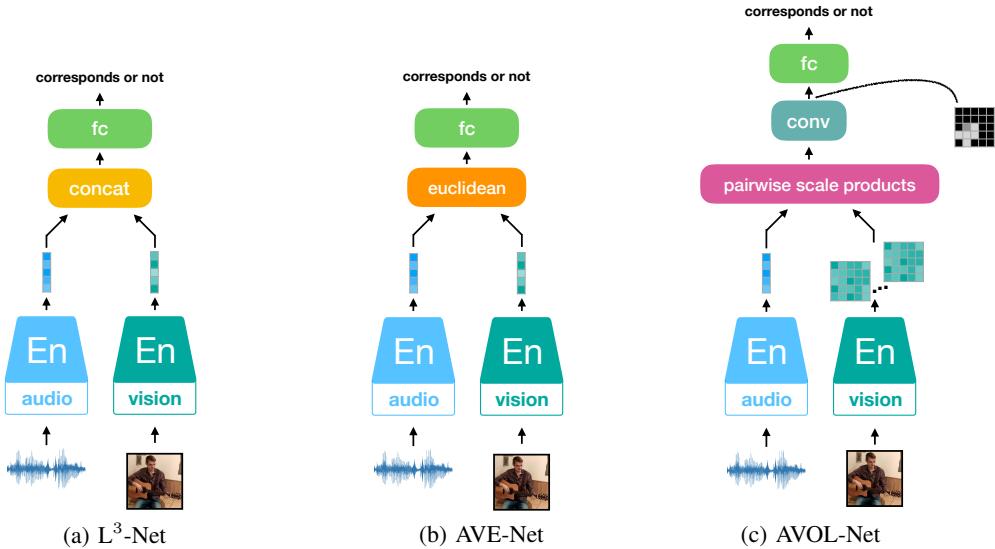


Fig. 8: Diagrams for L^3 , AVE-Net, and its modified version – AVOL-Net – for visual object localisation.

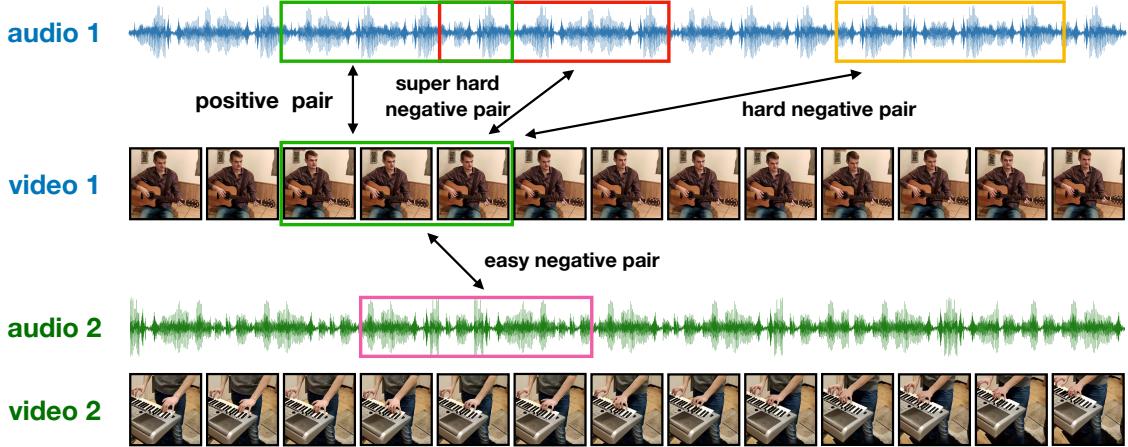


Fig. 9: Diagrams for demonstrating a negative sampling strategy introduced in [129].

The approach has shown promising results in learning cross-modal embeddings for the recognition of a person's identity. However, an incoming problem is that the model tends to rapidly learn to differentiate easy negative pairs from positive pairs while harder input pairs have very limited effect on learning discriminative representations. An opposite curriculum schedule has been shown to be effective for training a model that learns the cross-modal embeddings for ultrasound [132]. The visual and audio streams used in this work are medical ultrasound video and voice from a sonographer during the video recording. Due to the sparse correlation between the two inputs, hard positive and negative input pairs are first used in order to force the model to learn more strongly correlated representations. In [133], Zhang introduced a two-stage curriculum learning solution based on teacher-student training and identified it as self-supervised curriculum learning, for audio-visual representation learning. Before joint training of vision and audio sub-networks, one of the sub-networks is

updated using the representations from the other sub-network as teacher, which is frozen for updating. The two sub-networks exchange the role for training with the other sub-network. In addition, to enlarge the number of negative samples for training, a memory bank is applied, resulting in considerable improvements in a visual task of action recognition and an acoustic task of audio sound recognition.

Differently, in [129], the authors make use of margin loss in order to contrast positive and negative pairs that are of equal proportion. The negative examples of different hardness difficulty are considered, including easy negatives, hard negatives, and super-hard negatives (shown in Figure 9). The easy negatives are video frames and audio segments from different videos, while hard negatives are those pairs taken from the same video but that are at least half a second distant from each other. The super-hard negatives are the pairs that overlap for a certain temporal extent. The authors also confirmed the need of starting training the model with easy negatives and

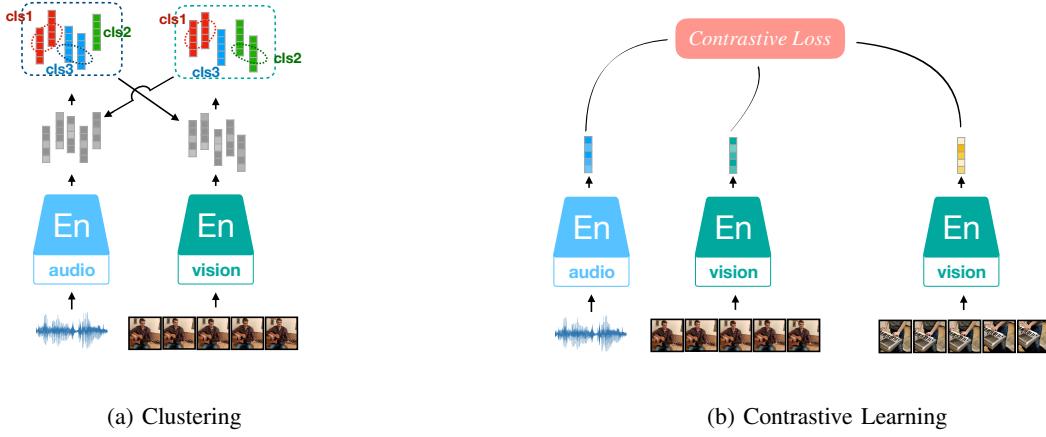


Fig. 10: Methods for audio-visual mutual supervision.

then gradually adding harder negatives, which has shown to be effective for learning high-quality representations. Similarly, treating negative pairs of different specialities, i.e., different difficulty levels, is investigated in [134] for audio-visual speaker diarisation. In this work, the margin value used in a triplet loss is controlled by the shifted range between audio and visual streams, by this representing a different difficulty degree of negatives. Nagrani et al. [135] optimised a model to learn audio-visual representations by formulating negative samples from the same video and different videos in content loss and identity loss. Additionally, in order to encourage explicit separation of representations, they used a disentanglement loss which is implemented as confusion loss in [136].

Harwath et al. [137]–[139] proposed another interesting pretext task by associating spoken audio captions with their corresponding image for learning audio-visual representations. Two networks are used to process the audio and image as input. In [137], the dot product of a pair of visual and audio representations is calculated as their similarity score. Similar to the AVOL-Net, similarity between audio representation and the visual embedding of each pixel can be computed to construct spatial activation maps, leading to a solution for object localisation [138]. In a different way, in [139], the generated audio and visual representations are pushed into a common latent space using triplet loss as well as by contrasting the positive pair to the negative pairs that contain either an unmatched caption or an unmatched image. Hsu et al. [140] solve the same task by building a system based on ResDAVEnet-VQ [139] and an Image-to-Unit Model [141]. Although each of these models is used to process an input stream, the two representations are pushed into the same latent space. Instead of using contrastive training objectives to reproduce the audio input, the learnt discrete linguistic units, learnt through ResDAVEnet-VQ from the input utterances, are fed to Tacotron2 [142], i.e., a text-to-speech (TTS) model for speech synthesis. The visual sub-network, i.e., ResDAVEnet-VQ, is then expected to learn representations that are close to the discrete linguistic units, by this enabling the representation learning to retrieve information from both modalities.

Based on their mutual correspondence, the audio and visual

streams of a video clip can be seen (to each other) as the supervisory signal for representation learning. An earlier study has shown the success of using synchronously recorded ambient sounds as supervision for visual learning [143]. Later on, Alwassel et al. [28] and Morgado et al. [144] empirically verified that exploiting the representation of one modality to create pseudo-labels for training the encoder network of the other modality outperforms not only SSL on a single modality, but also SSL based on pseudo-labels of both modalities. In [28], the pseudo-labels are generated using a deep clustering method (cf. Figure 10a). Differently, [144] aggregates ‘memory features’ by computing the slow exponential moving average (EMA), and subsequently applies contrastive learning (cf. Figure 10b), similarly as [69]. In [144], *Cross-Modal Agreement* (CMA) is additionally introduced to enhance the interactions between instances, specifically to calibrate within-modal similarities between positive pairs. Both methods, i.e., clustering- and contrastive learning-based modelling, learn effective audio-visual representations, evaluated on a downstream task of action recognition based on video.

2) Audio-Visual Source Separation: The PixelPlayer [27] effort proposed a Mix-and-Separate framework that solves visual object segmentation and audio source separation together. The framework consists of three networks: a video analysis network, an audio analysis network, and an audio synthesiser network; as shown in Figure 11a. The video analysis network extracts visual features from a sequence of video frames while the audio analysis network processes the mixture sound from two different video clips. The audio synthesiser network aims to separate the audio sources based on the learnt audio representations of the mixture, conditioned on the corresponding video frames. In this way, the model can learn a better semantic visual representation that is highly associated with its own audio, but less relevant to the audio of the other video clips. Although the learnt audio representations enable the model to retrieve information from the mixture sound, it cannot separate audio from each video. In a later work, the same authors also indicate that having synchronised audio and visual data is a requirement to disentangle the learnt audio and visual representations before feeding them into the

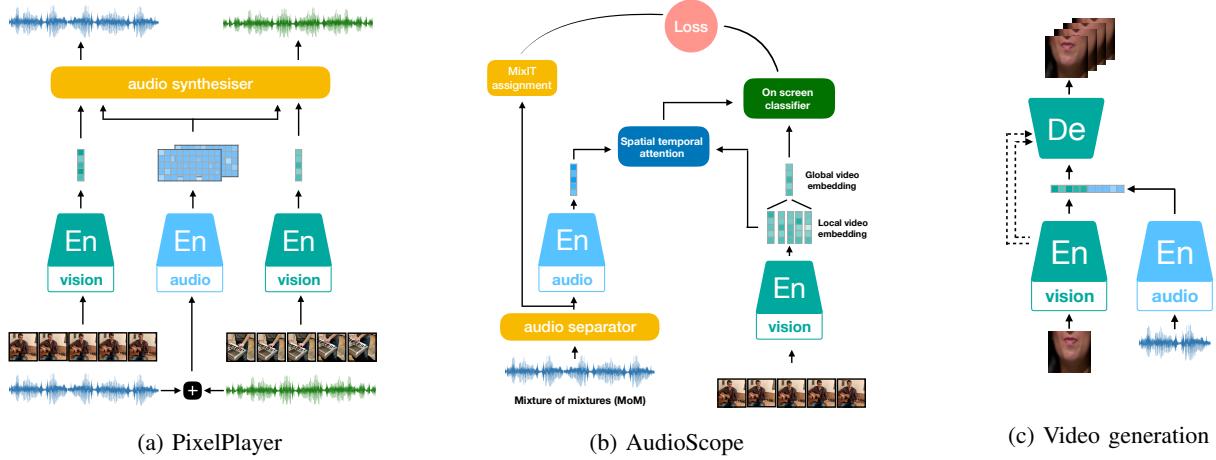


Fig. 11: Diagrams for PixelPlayer, AudioScope and the framework for video generation.

audio synthesiser network [145]. By doing this, the learnt audio and visual representations can be used independently.

AudioScope [146] expands the conditioning audio separation approach, and exploits an additional audio embedding network to process the separated audios. An audio representation then aggregates the global information from each resulting audio embedding using temporal pooling. Subsequently, attention is used to retrieve the mutual information between the local spatial-temporal video embedding (learnt with the video embedding network) and the global audio representations. This allows to generate an audio-visual representation that combines the audio and visual information. Finally, the audio-visual representations, i. e., the global video embedding and the global audio embedding, are concatenated together. By this, based on the separated audios, it is expected to create the MixIT assignment [124].

LWTNet [147] designs a model that can ingest a video and transform it into a set of discrete audio-visual objects using SSL. Similarly, an audio network and a video network encode the audio and video frames; then, a fine-grained audio visual attention map is computed by solving a synchronisation task, i. e., measuring the similarity between the audio and the visual features at every spatial location. The model can detect and track multiple audio-visual objects and extract an embedding for each of them. Given negative audio samples from shifted audio segments of the same video clip, contrastive loss is applied to maximise the similarity between a video frame and its true audio track. This, which is made in the form of an attention map, also minimises the similarity of the misaligned versions of the audio.

3) *Video Generation*: Given a starting video frame of a speaker, previous work has shown that a model can be trained to generate the subsequent video sequences based on the corresponding speech utterance [148]–[150]. In these works, a U-Net is used to code the starting video frame into a latent representation while an audio encoder is used to learn a representation of the speech utterance. In [148], the visual and audio representations are concatenated and then fed into the decoder of the U-Net to generate the full sequence of

video frames. Using neural decoders, the raw waveform, the log Mel-spectrogram, and the MFCC-spectrogram of the audio input are expected to be constructed from the learnt audio representation. L1 reconstruction errors between the output of the decoders (both audio and video) and the ground-truth, are minimised. Differently, in [149] the use of audio decoders is avoided and the length of the input audio is reduced to 0.2 s. In this work, a vector randomly created using Gaussian noise is appended to the audio-visual representation, by this injecting randomness in the procedure of face synthesis. In [148], [149], Shukla *et al.* show the efficacy of these two methods for spoken word classification and lip reading tasks. A later approach by the same authors is also developed for emotion recognition [150].

B. Audio & Text

As already introduced in Section III, Wav2vec-U uses self-supervised speech representations in order to fragment unlabelled samples and subsequently map them to phonemes through adversarial training. Similarly, Chung *et al.* [26] proposed to learn the individual speech and text embedding spaces, by aligning the two spaces via adversarial training and subsequently applying a refinement procedure. Under the assumption that embedding spaces from two modalities share a similar structure, in this work, the audio and text embeddings are first learnt using Speech2Vec and Word2Vec, respectively; then, adversarial training is used in order to learn a linear mapping between the speech embedding space and the text embedding space. The approach has shown promising results for the task of ASR and speech-to-text translation systems for low- or zero-resource languages such as German, which has little audio-text pairs for training.

The co-alignment of audio and text can also be done within an auto-encoder (AE) architecture. COALA, presented in [152], applies two AEs to process an audio spectrogram and the audio tag. Both AEs are optimised to reconstruct its input, resulting in semantic features of the audio and the text. The paired semantic features are pulled closer and the unpaired

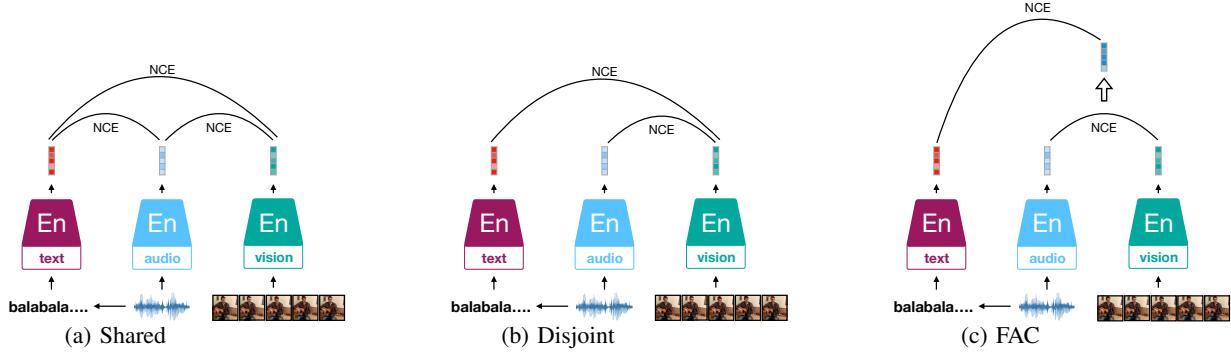


Fig. 12: Diagrams for three modes in a multi-modal versatile network (based on [151]).

semantic features are pushed far using a contrastive loss. The whole system is jointly optimised by minimising the two reconstruction errors and the contrastive loss as a multi-task learning problem. Affine transformations are applied to the two learnt representations, reducing the difficulty in maximising their agreement. An auto-encoder is also used in [153] for encoding audio spectrograms. As in COALA [152], the latent representations are also expected to be able to reconstruct the spectrogram and predict the linguistic features simultaneously. In these two works, one modality, either text or audio, is used to learn an embedding that is used to predict its paired input (in the other modality). Hence, it can be assumed that the learnt embedding contains the information from both streams. The reconstruction can be seen as a regularisation term that enables the embedding to reconstruct the input stream, by this ensuring that the learnt latent contains the salient features of the input stream. Similarly, CSTNet [154] is trained for speech-translation, but speech utterances are in English while the text translations are in any other language from French, German, Spanish, Portuguese, or Italian. The experimental results obtained from CSTNet indicate that the speech representation learnt using this framework can achieve comparable results for two downstream tasks, i.e., a Minimal-Pair ABX task and phone recognition.

C. Audio & Text & Video

To conclude this section, we will introduce some SSL works that learn audio representations through the use of three modalities: video, audio, and text. In these works, texts are commonly achieved by using off-the-shelf ASR systems from audio. For instance, in [155], the authors presented the use of keyword localisation as the pretext task. The authors also compare the performance obtained by separately using text or images as supervisory signal. They conclude that the visually supervised model performs worse than a text supervised model based on BoW. Indeed, although the visually trained model can sometimes locate semantically related words, this phenomenon is not consistently observed.

A multi-modal versatile network is presented in [151], a study that aims to find the best combination of the modalities. Learning a shared space of the three modalities, as well as two separate disjoint spaces for video-audio and video-text (considering that text originates from the audio), are

investigated. Fine and Coarse (FAC) spaces are additionally proposed due to the fact that the visual and audio domains differ (in terms of granularity) with respect to the language domain. In FAC, vision and audio are compared in a fine-grained space while text, audio, and vision are compared in a lower dimensional coarse-grained space. For this, the visual representation is first mapped into common latent spaces of audio and video, and sequentially projected into the common latent spaces of text and audio-visual common spaces. The authors consider no direct link between audio and text. Similar to the FAC approach, VATT [156] also presents a two-stage multi-modal projection. In VATT, audio and video are compared first using NCE loss. Subsequently, through the use of MIL-NEC loss [151] for optimisation purposes, the text is included in order to learn common latent spaces for the three modalities. Moreover, the authors suggest to use transformers for encoding all three modalities, which leads to a more uniform but efficient architecture.

V. DOWNSTREAM AUDIO TASKS & BENCHMARKS

After solving pretext tasks, an audio SSL model is expected to produce high-quality audio representations that are of sufficient generalisation and discrimination, by this, guaranteeing a good performance on downstream tasks. Several different downstream audio tasks have been considered for empirically measuring the audio representation quality. For example, ASR is used for evaluating all Wav2vec based methods [81], [84], [85]. Other tasks include speaker identification [36], [45], [103], speech emotion recognition [32], [157]–[159], speech machine translation [160], pitch detection [161], and acoustic scene classification [90], amongst others.

Hereby, in the following we will describe some publicly available benchmarks that enable a fair comparisons between different audio SSL algorithms.

The Zero resource Speech challenge (ZeroSpeech)² [162] started their first challenge in 2015 with the task of unsupervised discovery of linguistic units from raw speech in an unknown language. The tasks are split into two tracks, i.e., unsupervised sub-word modelling and spoken term discovery, each focusing on a different level of linguistic structure. The first track aims at constructing a representation of speech

²<https://zerospeech.com>

sounds that is robust to within- and between-speaker variation and supports word identification. The second targets at unsupervised discovery of ‘words’, taking raw speech as input. In 2017, the organisers extend the study for the variants in language and speaker, considering the topic of cross-language generalisation and speaker adaptation. The task of ZeroSpeech 2019 was to address the problem of a speech synthesiser without any text or phonetic labels. Participants were expected to discover sub-word units in an unsupervised way given raw audio. Subsequently, they were supposed to align them to the voice recording (as good as possible) for the purpose of synthesising utterances of target speakers. The latest challenge, launched in 2021, provided several tasks for spoken language modelling, based on speech only as well as visually-grounded. Speech-based language modelling consists in learning language models directly from raw audio in an unknown language. Visually-grounded language modelling aims at learning language models incorporating the visual information.

The Speech processing Universal PERformance Benchmark (SuperB)³ [163] aims to present a standard and comprehensive testbed for evaluation which can be generally applied to pre-trained models on various tasks. Taking into account the available datasets and conventional evaluation protocols, ten downstream tasks are provided: Phoneme Recognition, Automatic Speech Recognition, Keyword Spotting, Query by Example Spoken Term Detection, Speaker Identification, Automatic Speaker Verification, Speaker Diarisation, Intent Classification, Slot Filling, and Emotion Recognition.

LeBenchmark⁴ [164] is another reproducible and multi-faceted benchmark for evaluating speech SSL models for the French language. It is based on four tasks which aim to assess the speech representations: Speech Recognition (ASR), Spoken Language Understanding (SLU), Speech Translation (AST), and Emotion Recognition (AER). For reproducibility, the LeBenchmark organisers provided pre-trained SSL models learnt on different subsets of a large and heterogeneous collection of read, prepared, and spontaneous speech utterances in French.

Libri-Light [165] is a benchmark specifically designed for the task of ASR with limited or no supervision. Libri-Light is based on spoken English audio collected from open-source audio books of the LibriVox project.

HEAR⁵, short for Holistic Evaluation of Audio Representations, extends a benchmark suite for both speech and non-speech tasks. The challenge requests participants to create an audio representation that is as holistic as the human ear. In HEAR 2021, three main tasks are used including word classification, pitch detection, and sound event detection.

VI. DISCUSSION

In this section, we first clarify the differences and similarities between SSL methods and other confusing machine learning

mechanisms. Next, we discuss the the common problems and difficulties met during the development of SSL models. We further point out some additional concerns regarding audio SSL, considering the difference in data processing and augmentations, negative sample generation, and network construction, compared to the SSL approaches for other modalities.

A. Difference from Other Confusing Learning Mechanisms

Generally speaking, representation learning aims to capture the posterior distribution of the underlying explanatory factors from the observed input data. A good representation should be of sufficient generalisation and distinctiveness, so that it carries complete salient information of the data that is useful as input for supervised tasks, such as classification. Representation SSL is a representation learning approach that trains a model in order to produce representations. This is achieved by solving specially defined pretext tasks based on, usually very large-scale, data without human annotations. This is different to the classic learning mechanisms of transfer learning and domain adaptation, which learn to generate representations in supervised frameworks, i.e., using labelled data. SSL is commonly regarded as an unsupervised leaning method, as the ones using data without human annotations. However, it is also different from classic unsupervised learning, such as clustering, because these kinds of unsupervised learning concentrate on grouping inputs that have similar data patterns, whereas SSL learns representations with supervision of some automatically created training targets, such as pseudo-labels. Likewise, it is considered unsupervised in the sense of no labels from the target task are involved.

Contrastive SSL is highly related to distance metric learning (or simply, metric learning) [166]. Given an anchor paired with positive samples and negative samples, a weakly-supervised metric learning constructs a distance metric that puts positives close together and negatives far away in a latent space. Hence, contrastive SSL can be seen as a metric learning where the positive pairs are created from the same data source through procedures such as data augmentation. Contrastive SSL is also similar to instance discrimination [167]. Instead of processing positive and negative pairs, instance discrimination takes each data sample as from a separate class, and learns feature representation that discriminates among individual instances. According to our analysis of Equation (10) given in Section II-B2, when the temperature parameter is set too small, the InfoNCE loss tends to take the two inputs of a positive pair as the different instances, and therefore optimises the SSL model in a way of the same effect of instance discrimination.

Besides, Generative Adversarial Networks (GANs) are also seen as a kind of SSL framework in some works [8], [11], [16]. For instance, the generator creates data from a random vector by taking the real data as training targets. Then, the discriminator network aims to measure the similarity between generated and real data. It is worth noticing that the similarity measure changes, as the discriminator is updated. Such a kind of generative contrastive model has been successfully investigated for NLP tasks, such as in ELETRA [168], but rarely been explored for audio SSL. Hence, we did not

³<https://superbbenchmark.org/>

⁴<http://lebenchmark.com/>

⁵<https://neuralaudio.ai/hear2021-holistic-evaluation-of-audio-representations.html>

introduce it as an audio SSL form in the literature review, though it should be naturally considered for future works.

B. Difficulties and Problems for SSL Optimisation

The representation quality using SSL is determined by the efficacy of pretext tasks, of which the key component is the design of training targets or objectives. The training objectives for both predictive and contrastive SSL concentrate on the correlations between representations of observed data. Both methods concentrate on maximising the similarity between the representations of the two views of one unique data sample. Additionally, contrastive methods contrast the similarity against the distance to other data samples.

Representational collapse often appears when training a predictive SSL model, such as using a Siamese network architecture. To tackle this issue, the pair of networks is usually designed to be of asymmetric architecture and is updated asynchronously. Data augmentation techniques, used to generate different views of an input data, are used to additionally force the Siamese network to process asymmetric input. Contrastive SSL alleviates the problem of mode collapse by driving the representations of samples, including positives and negatives, to maximal-uniformly distributed appearance on a unit sphere. Minimising a contrastive loss, such as InfoNCE, is found to be approximately equivalent to maximise the mutual information between representations. With the rise of the number of negative samples, a lower bound on mutual information is raised up [16], [40]. Therefore, better representations that carry more correlation information between representations can be obtained by enlarging the amount of negative samples, for instance, as shown in [20] and [29]. Similar considerations have led to the success of contrastive audio SSL [40], [81], [84]. For this to happen, however, the requirement on memory dramatically boosts. Therefore, negative sampling of better efficiency needs to be further explored. On the other hand, according to the theoretical analysis in [17], a too large number of negative samples may not be profitable for training contrastive SSL models. So far, no research has been done to suggest a golden standard rule for setting a proper number of negatives. Moreover, the setting should potentially be considered differently for different tasks and applications. Another issue that can hamper contrastive SSL is early degeneration, which means that the SSL model over-fits to the discriminative pretext task in very early training steps, and therefore, the representations do not present a sufficient generalisation ability. Solutions that can relax this early degeneration issue should also be addressed in future work.

C. Additional adjustments on SSL for Audio

As introduced above, SSL approaches that have been well explored for CV and NLP tasks, are being transferred to the audio domain. For this, some works process 1D audio data into a 2D format, in order to match the formulations of these SSL frameworks. For example, time-frequency representations of audio and the advanced transformations based on it, such as a spectrogram, Mel-spectrogram, and MFCC can be used

as ‘image’ to some SSL models designed for CV tasks [37], [89], [93]. For this case, data augmentation techniques widely used in the CV domain have also been considered, which are essential for achieving high-quality representations. Taking the features as sequential frames, we can process them in similar ways as we would do for NLP tasks [48], [49].

An alternative way is to directly process the 1D waveform using deep learning encoders, such as 1D convolution, able to convert the 1D signal into higher-dimensional features for further processing. This solution has been successfully used in [40], [81], [96]. The focus of this paper was not assessing network architectures, but rather concentrating on the framework and formulations of SSL approaches. As the network architecture is found to be less important than formulations in achieving good representations, ResNet is typically used in most visual SSL works. However, the importance of neural network architectures are not that clear for audio SSL. Researchers tend to use the network architectures that were designed to respect the speech or audio structure, which can achieve more promising results in the context of audio SSL. Still, more research evaluating the effect of network components, such as assessing the effect of the attention mechanism used in transformers [101], should be carried out.

D. Fitness and mismatch between pretext and downstream tasks

In general, we expect that by training a model with SSL, it is possible to learn general representations that are effective for downstream tasks. Although this is slightly different from classic transfer learning which performs pretext tasks in a supervised framework, the gap between the source data in pretext tasks and target data for downstream tasks is expected to be matched.

Comparing speech and other audio signals, such as acoustic scene recordings, the speech signal is more variable from a temporal and frequency perspective, while the acoustic scene recordings are usually more stationary along the temporal axis. Hence, an SSL model that is trained on a speech signal may still cover the ability to learn discriminative representations to be used in acoustic scene classification. However, in the reverse way, an SSL model trained on scene recordings may still work for simple speech-related classification tasks, such as speech command recognition, but its performance would be weak in more advanced speech tasks, such as ASR.

In the standard framework of SSL, labelled data is used in downstream tasks for fine-tuning. It has been shown that a small quantity of labelled data can already guide a pre-trained model to achieve very satisfying performance on downstream tasks. This inspired semi-supervised learning using very little human-labelled data from the target data domain, for closing the gap between source and target data. Specifically, the training objectives of SSL and supervised learning are combined and optimised simultaneously. For many audio applications, SSL approaches have shown promising performance and reached (or even surpassed) state-of-the-art results achieved through supervised learning. Still, when labels are available or partly available, like in CLAR [33] and UniSpeech [169],

combining SSL and SL together into a multi-task learning setting enables to learn better speech representations for some audio tasks.

VII. CONCLUSION

This survey has provided an overview of the existing approaches and methods for uni-modal and multi-modal self-supervised learning approaches using audio. The success of these methods has been analysed in several classic audio tasks, including speech recognition, speaker identification, speech emotion recognition, and acoustic scene classification. Audio SSL methods, such as Wav2Vec 2.0, have shown to even surpass the performance of supervised learning methods on the same task. Moreover, the generalisation ability of representations learnt using audio SSL can decrease the urgency of searching for hand-crafted, engineered features. The superior performances obtained using SSL-based approaches support the generalisation capabilities of this representation learning method, and encourage the use of this technique to shape the future and advance the state-of-the-art in the field of audio processing.

ACKNOWLEDGEMENTS

This project has been supported in part by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 826506 (sustAGE), and by the BIT Teli Young Fellow Program from the Beijing Institute of Technology, China.

REFERENCES

- [1] J. Piaget, "Part I: Cognitive development in children: Piaget development and learning," *Journal of research in science teaching*, vol. 2, no. 3, pp. 176–186, 1964.
- [2] W. Huitt and J. Hummel, "Piaget's theory of cognitive development," *Educational psychology interactive*, vol. 3, no. 2, pp. 1–5, 2003.
- [3] R. Baillargeon and J. DeVos, "Object permanence in young infants: Further evidence," *Child development*, vol. 62, no. 6, pp. 1227–1246, 1991.
- [4] G. W. Oesterdiekhoff, "Child and ancient man: How to define their commonalities and differences," *American Journal of Psychology*, vol. 129, no. 3, pp. 295–312, 2016.
- [5] W. F. Brewer and G. V. Nakamura, "The nature and functions of schemas," *Center for the Study of Reading Technical Report*, no. 325, p. 52 pages, 1984.
- [6] B. J. Wadsworth, *Piaget's theory of cognitive and affective development: Foundations of constructivism*, 1996.
- [7] D. N. Perkins, G. Salomon *et al.*, "Transfer of learning," *International encyclopedia of education*, vol. 2, pp. 6452–6457, 1992.
- [8] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.
- [9] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [10] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. ICML*, Sydney, Australia, 2007, pp. 759–766.
- [11] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, p. 20 pages, 2021.
- [12] Y. Bansal, G. Kaplun, and B. Barak, "For self-supervised learning, rationality implies generalization, provably," in *Proc. ICLR*, Vienna, Austria, 2021, p. 25 pages.
- [13] J. Teng and W. Huang, "Can pretext-based self-supervised learning be boosted by downstream data? a theoretical analysis," *CoRR*, 2021.
- [14] J. D. Lee, Q. Lei, N. Saunshi, and J. Zhuo, "Predicting what you already know helps: Provable self-supervised learning," in *Proc. ICLR*, Vienna, Austria, 2021, p. 30 pages.
- [15] F. Wang and H. Liu, "Understanding the behaviour of contrastive loss," in *Proc. CVPR*, Nashville, TN, USA, 2021, pp. 2495–2504.
- [16] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive representation learning: A framework and review," *IEEE Access*, vol. 8, pp. 193 907–193 934, 2020.
- [17] N. Saunshi, O. Plevrakis, S. Arora, M. Khodak, and H. Khandeparkar, "A theoretical analysis of contrastive unsupervised representation learning," in *Proc. ICML*, Long Beach, CA, USA, 2019, pp. 5628–5637.
- [18] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 2, p. 22 pages, 2021.
- [19] C. Tosh, A. Krishnamurthy, and D. Hsu, "Contrastive learning, multi-view redundancy, and linear models," in *Proc. ALT*, 2021, pp. 1179–1206.
- [20] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. ICML*, 2020, p. 11 pages.
- [21] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, vol. 63, pp. 1–26, 2020.
- [22] L. Wu, H. Lin, C. Tan, Z. Gao, and S. Z. Li, "Self-supervised learning on graphs: Contrastive, generative, or predictive," *IEEE Transactions on Knowledge and Data Engineering*, p. 20 pages, 2021.
- [23] E. Shelhamer, P. Mahmoudieh, M. Argus, and T. Darrell, "Loss is its own reward: Self-supervision for reinforcement learning," p. 4 pages, 2017.
- [24] S. Liu, G. Keren, E. Parada-Cabaleiro, and B. Schuller, "N-HANS: A neural network-based toolkit for in-the-wild audio enhancement," *Multimedia Tools and Applications*, vol. 80, pp. 28 365–28 389, 2021.
- [25] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *Proc. ICRA*, Brisbane, Australia, 2018, pp. 1134–1141.
- [26] Y.-A. Chung, W.-H. Weng, S. Tong, and J. Glass, "Unsupervised cross-modal alignment of speech and text embedding spaces," *Proc. NeurIPS*, vol. 31, pp. 7354–7364, 2018.
- [27] H. Zhao, C. Gan, A. Rouditchenko, C. Vondrick, J. McDermott, and A. Torralba, "The sound of pixels," in *Proc. ECCV*, Munich, Germany, 2018, pp. 570–586.
- [28] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, and D. Tran, "Self-supervised learning by cross-modal audio-video clustering," in *Proc. NeurIPS*, 2020, p. 13 pages.
- [29] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. ICCV*, Venice, Italy, 2020, pp. 9729–9738.
- [30] N. Komodakis and S. Gidaris, "Unsupervised representation learning by predicting image rotations," in *Proc. ICLR*, Salt Lake City, Utah, USA, 2018, p. 16 pages.
- [31] G. Larsson, M. Maire, and G. Shakhnarovich, "Colorization as a proxy task for visual understanding," *Proc. CVPR*, pp. 840–849, 2017.
- [32] A. Nandan and J. Vepa, "Language agnostic speech embeddings for emotion classification," in *Proc. ICML*, 2020, p. 6 pages.
- [33] H. Al-Tahan and Y. Mohsenzadeh, "CLAR: Contrastive learning of auditory representations," in *Proc. AISTATS*, 2021, pp. 2530–2538.
- [34] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple augmentation method for automatic speech recognition," in *Proc. INTERSPEECH*, Graz, Austria, 2019, pp. 2613–2617.
- [35] D. Dwibedi, J. Tompson, C. Lynch, and P. Sermanet, "Learning actionable representations from visual observations," in *Proc. IROS*, Madrid, Spain, 2018, pp. 1577–1584.
- [36] M. Ravanelli and Y. Bengio, "Learning speaker representations with mutual information," in *Proc. INTERSPEECH*, Graz, Austria, 2019, pp. 1153–1157.
- [37] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," in *Proc. ICASSP*, Toronto, Canada, 2021, pp. 3875–3879.
- [38] E. Fonseca, D. Ortego, K. McGuinness, N. E. O'Connor, and X. Serra, "Unsupervised contrastive learning of sound event representations," in *Proc. ICASSP*, Toronto, Canada, 2021, pp. 371–375.
- [39] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *Proc. ICLR*, Vancouver, Canada, 2019, p. 24 pages.

- [40] A. v. d. Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.
- [41] B. Poole, C. Sun, C. Schmid, D. Krishnan, P. Isola, and Y. Tian, “What makes for good views for contrastive representation learning?” in *Proc. NeurIPS*, 2020, p. 13 pages.
- [42] M. Tschannen, J. Djolonga, P. K. Rubenstein, S. Gelly, and M. Lucic, “On mutual information maximization for representation learning,” in *Proc. ICLR*, New Orleans, LA, USA, 2019.
- [43] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, “Big self-supervised models are strong semi-supervised learners,” in *Proc. NeurIPS*, online, 2020, p. 13 pages.
- [44] X. Chen, H. Fan, R. Girshick, and K. He, “Improved baselines with momentum contrastive learning,” *arXiv preprint arXiv:2003.04297*, 2020.
- [45] X. Chen, S. Xie, and K. He, “An empirical study of training self-supervised vision transformers,” in *Proc. ICCV*, 2021, pp. 9640–9649.
- [46] S. Gidaris, P. Singh, and N. Komodakis, “Unsupervised representation learning by predicting image rotations,” in *Proc. ICLR*, Vancouver, Canada, 2018, p. 16 pages.
- [47] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in *Proc. ICLR*, Scottsdale, AZ, USA, 2013, p. 12 pages.
- [48] Y.-A. Chung and J. Glass, “Speech2Vec: A sequence-to-sequence framework for learning word embeddings from speech,” in *Proc. INTERSPEECH*, Hyderabad, India, 2018, pp. 811–815.
- [49] B. Gfeller, D. Roblek, F. de Chaumont Quirky, and M. Tagliasacchi, “Pre-training audio representations with self-supervision,” *IEEE Signal Processing Letters*, vol. 27, no. 2020, pp. 600–604, 2020.
- [50] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL*, Minneapolis, MN, USA, 2019, pp. 4171–4186.
- [51] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “Albert: A lite bert for self-supervised learning of language representations,” in *ICLR*, 2020, pp. 6707–6717.
- [52] A. H. Liu, Y.-A. Chung, and J. Glass, “Non-autoregressive predictive coding for learning speech representations from local dependencies,” in *Proc. INTERSPEECH*, Brno, Czechia, 2021, pp. 3730–3734.
- [53] P. H. Richemond, J.-B. Grill, F. Altché, C. Tallec, F. Strub, A. Brock, S. L. Smith, S. De, R. Pascanu, B. Piot, and M. Valko, “BYOL works even without batch statistics,” in *Proc. NeurIPS Workshop: Self-Supervised Learning - Theory and Practice*, 2020.
- [54] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, “Barlow twins: Self-supervised learning via redundancy reduction,” p. 11 pages, 2021.
- [55] X. Chen and K. He, “Exploring simple siamese representation learning,” in *Proc. ICCV*, Montreal, Canada, 2021, pp. 15750–15758.
- [56] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, “Deep clustering for unsupervised learning of visual features,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 132–149.
- [57] C. Zhuang, A. L. Zhai, and D. Yamins, “Local aggregation for unsupervised learning of visual embeddings,” in *Proc. ICCV*, Seoul, South Korea, 2019, pp. 6002–6012.
- [58] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, “Unsupervised learning of visual features by contrasting cluster assignments,” in *Proc. NeurIPS*, 2020, pp. 9912–9924.
- [59] O. Henaff, “Data-efficient image recognition with contrastive predictive coding,” in *Proc. ICML*, 2020, pp. 4182–4192.
- [60] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proceedings of ICML workshop on unsupervised and transfer learning*, Bellevue, Washington, USA, 2012, pp. 37–49.
- [61] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [62] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Proc. NeurIPS*, Sierra Nevada, USA, 2013, pp. 3111–3119.
- [63] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, “An Unsupervised Autoregressive Model for Speech Representation Learning,” in *Proc. INTERSPEECH*, Graz, Austria, 2019, pp. 146–150.
- [64] C. Doersch, A. Gupta, and A. A. Efros, “Unsupervised visual representation learning by context prediction,” in *Proc. ICCV*, Santiago, Chile, 2015, pp. 1422–1430.
- [65] M. Noroozi and P. Favaro, “Unsupervised learning of visual representations by solving jigsaw puzzles,” in *Proc. ECCV*, Amsterdam, Netherlands, 2016, pp. 69–84.
- [66] I. Misra and L. v. d. Maaten, “Self-supervised learning of pre-text-invariant representations,” in *Proc. ICCV*, 2020, pp. 6707–6717.
- [67] A. N. Carr, Q. Berthet, M. Blondel, O. Teboul, and N. Zeghidour, “Self-supervised learning of audio representations from permutations with differentiable ranking,” *IEEE Signal Processing Letters*, vol. 28, pp. 708–712, 2021.
- [68] P. Ryan, S. Takafuji, C. Yang, N. Wilson, and C. McBride, “Using self-supervised learning of birdsong for downstream industrial audio classification,” in *Proc. ICML*, 2020, p. 5 pages.
- [69] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Pires, Z. Guo, M. Azar *et al.*, “Bootstrap your own latent: A new approach to self-supervised learning,” in *Proc. NeurIPS*, 2020.
- [70] Y. You, I. Gitman, and B. Ginsburg, “Large batch training of convolutional networks,” *arXiv preprint arXiv:1708.03888*, 2017.
- [71] Y. Tian, X. Chen, and S. Ganguli, “Understanding self-supervised learning dynamics without contrastive pairs,” in *Proc. ICML*, 2021, pp. 10268–10278.
- [72] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proc. ICML*, Lille, France, 2015, pp. 448–456.
- [73] H. B. Barlow *et al.*, “Possible principles underlying the transformation of sensory messages,” *Sensory communication*, vol. 1, no. 01, 1961.
- [74] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, “Boosting self-supervised learning via knowledge transfer,” in *Proc. CVPR*, Salt Lake City, Utah, USA, 2018, pp. 9359–9367.
- [75] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” in *Proc. CVPR*, vol. 1, San Diego, CA, USA, 2005, pp. 539–546.
- [76] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *Proc. CVPR*, Boston, MA, USA, 2015, pp. 815–823.
- [77] K. Sohn, “Improved deep metric learning with multi-class n-pair loss objective,” in *Proc. NeurIPS*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., Barcelona, Spain, 2016, p. 9 pages.
- [78] M. Gutmann and A. Hyvärinen, “Noise-contrastive estimation: A new estimation principle for unnormalized statistical models,” in *Proc. AISTATS*, Sardinia, Italy, 2010, pp. 297–304.
- [79] T. Wang and P. Isola, “Understanding contrastive representation learning through alignment and uniformity on the hypersphere,” in *Proc. ICML*, 2020, pp. 9929–9939.
- [80] S. Liu, J. Han, E. Puyal, S. Kontaxis, S. Sun, P. Locatelli, J. Dineley, F. Pokorny, G. Costa, L. Leocani, A. Guerrero, C. Nos, A. Zabalza, P. Soerensen, M. Buron, M. Magyari, Y. Ranjan, Z. Rashid, P. Conde, and R.-C. Consortium, “Fitbeat: COVID-19 estimation based on wristband heart rate using a contrastive convolutional auto-encoder,” *Pattern Recognition*, vol. 123, p. 108403, 2021.
- [81] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “Wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Proc. NeurIPS*, p. 12 pages, 2020.
- [82] A. Kolesnikov, X. Zhai, and L. Beyer, “Revisiting self-supervised visual representation learning,” in *Proc. ICCV*, Long Beach, CA, USA, 2019, pp. 1920–1929.
- [83] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl, “Sampling matters in deep embedding learning,” in *Proc. ICCV*, Venice, Italy, 2017, pp. 2840–2848.
- [84] S. Schneider, A. Baevski, R. Collobert, and M. Auli, “wav2vec: Unsupervised pre-training for speech recognition,” in *Proc. INTERSPEECH*, Graz, Austria, 2019, pp. 3465–3469.
- [85] A. Baevski, S. Schneider, and M. Auli, “vq-wav2vec: Self-supervised learning of discrete speech representations,” in *Proc. ICLR*, New Orleans, LA, USA, 2019, p. 12 pages.
- [86] Y.-A. Chung and J. Glass, “Generative pre-training for speech with autoregressive predictive coding,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 3497–3501.
- [87] A. H. Liu, T. Tu, H.-y. Lee, and L.-s. Lee, “Towards unsupervised speech recognition and synthesis with quantized speech representation learning,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 7259–7263.
- [88] L. Wang and A. van den Oord, “Multi-format contrastive learning of audio representations,” in *Proc. NeurIPS*, 2020, p. 8 pages.
- [89] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, “BYOL for audio: Self-supervised learning for general-purpose audio representation,” pp. 1–8, 2021.
- [90] F. Gontier, V. Lostanlen, M. Lagrange, N. Fortin, C. Lavandier, and J.-F. Petiot, “Polyphonic training set synthesis improves self-supervised urban sound classification,” *The Journal of the Acoustical Society of America*, vol. 149, no. 6, pp. 4309–4326, 2021.

- [91] M. Tagliasacchi, B. Gfeller, F. d. C. Quiry, and D. Roblek, “Pre-training audio representations with self-supervision,” *IEEE Signal Processing Letters*, vol. 27, pp. 600–604, 2020.
- [92] 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,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 6419–6423.
- [93] A. T. Liu, S.-W. Li, and H.-y. Lee, “TERA: Self-supervised learning of transformer encoder representation for speech,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2351–2366, 2021.
- [94] P.-H. Chi, P.-H. Chung, T.-H. Wu, C.-C. Hsieh, Y.-H. Chen, S.-W. Li, and H.-y. Lee, “Audio albert: A lite bert for self-supervised learning of audio representation,” in *Proc. SLT*, 2021, pp. 344–350.
- [95] J. Bai, W. Wang, Y. Zhou, and C. Xiong, “Representation learning for sequence data with deep autoencoding predictive components,” in *Proc. ICLR*, 2021. [Online]. Available: <https://openreview.net/forum?id=Naqw7EHIfrv>
- [96] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, “Learning problem-agnostic speech representations from multiple self-supervised tasks,” in *Proc. INTERSPEECH*, Graz, Austria, 2019, pp. 161–165.
- [97] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, “Multi-task self-supervised learning for robust speech recognition,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 6989–6993.
- [98] 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 *Proc. SLT*, 2021, pp. 215–222.
- [99] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, “Hubert: Self-Supervised speech representation learning by masked prediction of hidden units,” *arXiv preprint arXiv:2106.07447*, 2021.
- [100] I. Misra, C. L. Zitnick, and M. Hebert, “Shuffle and learn: Unsupervised learning using temporal order verification,” in *Proc. ECCV*, Amsterdam, Netherlands, 2016, pp. 527–544.
- [101] S. Yang, A. T. Liu, and H. yi Lee, “Understanding self-attention of self-supervised audio transformers,” in *Proc. ICML*, 2020, p. 5 pages.
- [102] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, “Neural discrete representation learning,” in *Proc. NeurIPS*, Long Beach, CA, USA, 2017, pp. 6309–6318.
- [103] M. Ravanelli and Y. Bengio, “Speaker recognition from raw waveform with SincNet,” *Proc. SLT*, pp. 1021–1028, 2018.
- [104] J. Bradbury, S. Merity, C. Xiong, and R. Socher, “Quasi-recurrent neural networks,” *Proc. ICLR*, p. 12 pages, 2017.
- [105] A. Baevski and A. Mohamed, “Effectiveness of self-supervised pre-training for ASR,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 7694–7698.
- [106] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with gumbel-softmax,” in *Proc. ICLR*, New Orleans, LA, USA, 2019, p. 12 pages.
- [107] Y.-A. Chung, H. Tang, and J. Glass, “Vector-Quantized autoregressive predictive coding,” pp. 3760–3764, 2020.
- [108] H. Jegou, M. Douze, and C. Schmid, “Product quantization for nearest neighbor search,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 1, pp. 117–128, 2010.
- [109] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an ASR corpus based on public domain audio books,” in *Proc. ICASSP*, Brisbane, Australia, 2015, pp. 5206–5210.
- [110] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, “DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1,” *NASA STI/Recon Technical Report*, vol. 93, p. 27403, 1993.
- [111] W.-N. Hsu, A. Sriram, A. Baevski, T. Likhomannenko, Q. Xu, V. Pratap, J. Kahn, A. Lee, R. Collobert, G. Synnaeve, and M. Auli, “Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training,” in *Proc. INTERSPEECH*, Brno, Czech Republic, 2021, pp. 721–725.
- [112] M. Rivière, A. Joulin, P.-E. Mazaré, and E. Dupoux, “Unsupervised pretraining transfers well across languages,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 7414–7418.
- [113] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, and M. Auli, “XLS-R: Self-supervised cross-lingual speech representation learning at scale,” 2021.
- [114] K. Kawakami, L. Wang, C. Dyer, P. Blunsom, and A. van den Oord, “Learning robust and multilingual speech representations,” in *Proc. EMNLP*, 2020, pp. 1182–1192.
- [115] S. Sadhu, D. He, C.-W. Huang, S. H. Mallidi, M. Wu, A. Rastrow, A. Stolcke, J. Droppo, and R. Maas, “Wav2vec-C: A self-supervised model for speech representation learning,” *arXiv preprint arXiv:2103.08393*, 2021.
- [116] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, “Unsupervised speech recognition,” in *Proc. NeurIPS*, 2021, p. 15 pages.
- [117] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in *Proc. ICML*, Pittsburgh, PA, USA, 2006, pp. 369–376.
- [118] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.
- [119] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, “Phase-aware speech enhancement with deep complex u-net,” in *Proc. ICLR*, Vancouver, Canada, 2018, p. 20 pages.
- [120] Y.-C. Wang, S. Venkataramani, and P. Smaragdis, “Self-supervised learning for speech enhancement,” *arXiv preprint arXiv:2006.10388*, 2020.
- [121] A. Sivaraman and M. Kim, “Self-supervised learning from contrastive mixtures for personalized speech enhancement,” in *Proc. NeurIPS*, 2020, p. 4 pages.
- [122] Y. Qiu, R. Wang, S. Singh, Z. Ma, and F. Hou, “Self-supervised learning based phone-fortified speech enhancement,” in *Proc. INTERSPEECH*, Brno, Czech Republic, 2021, pp. 211–215.
- [123] S.-F. Huang, S.-P. Chuang, D.-R. Liu, Y.-C. Chen, G.-P. Yang, and H.-y. Lee, “Stabilizing label assignment for speech separation by self-supervised pre-training,” in *Proc. INTERSPEECH*, 08 2021, pp. 3056–3060.
- [124] S. Wisdom, E. Tzinis, H. Erdogan, R. J. Weiss, K. Wilson, and J. R. Hershey, “Unsupervised sound separation using mixture invariant training,” in *Proc. NeurIPS*, 2020, p. 17 pages.
- [125] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, “Permutation invariant training of deep models for speaker-independent multi-talker speech separation,” in *Proc. ICASSP*, New Orleans, LA, USA, 2017, pp. 241–245.
- [126] A. Sivaraman, S. Kim, and M. Kim, “Personalized speech enhancement through self-supervised data augmentation and purification,” in *Proc. INTERSPEECH*, 08 2021, pp. 2676–2680.
- [127] R. Arandjelovic and A. Zisserman, “Look, listen and learn,” in *Proc. ICCV*, Venice, Italy, 2017, pp. 609–617.
- [128] ———, “Objects that sound,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 435–451.
- [129] B. Korbar, D. Tran, and L. Torresani, “Cooperative learning of audio and video models from self-supervised synchronization,” in *Proc. NeurIPS*, 2018, p. 7774–7785.
- [130] A. Owens and A. A. Efros, “Audio-visual scene analysis with self-supervised multisensory features,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 631–648.
- [131] A. Nagrani, S. Albanie, and A. Zisserman, “Learnable PINs: Cross-modal embeddings for person identity,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 71–88.
- [132] J. Jiao, Y. Cai, M. Alsharid, L. Drukker, A. T. Papageorgiou, and J. A. Noble, “Self-supervised contrastive video-speech representation learning for ultrasound,” in *Proc. MICCAI*, Lima, Peru, 2020, pp. 534–543.
- [133] J. Zhang, X. Xu, F. Shen, H. Lu, X. Liu, and H. T. Shen, “Enhancing audio-visual association with self-supervised curriculum learning,” in *Proc. AAAI Conference on Artificial Intelligence*, 2021, pp. 3351–3359.
- [134] Y. Ding, Y. Xu, S.-X. Zhang, Y. Cong, and L. Wang, “Self-supervised learning for audio-visual speaker diarization,” in *Proc. ICASSP*, Brighton, UK, 2020, pp. 4367–4371.
- [135] A. Nagrani, J. S. Chung, S. Albanie, and A. Zisserman, “Disentangled speech embeddings using cross-modal self-supervision,” in *Proc. ICASSP*, 2020, pp. 6829–6833.
- [136] M. Alvi, A. Zisserman, and C. Nellåker, “Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 556–572.
- [137] D. Harwath, A. Torralba, and J. R. Glass, “Unsupervised learning of spoken language with visual context,” p. 1866–1874, 2016.
- [138] D. Harwath, A. Recasens, D. Surís, G. Chuang, A. Torralba, and J. Glass, “Jointly discovering visual objects and spoken words from raw sensory input,” in *Proc. ECCV*, Munich, Germany, 2018, pp. 649–665.

- [139] D. Harwath, W.-N. Hsu, and J. Glass, “Learning hierarchical discrete linguistic units from visually-grounded speech,” in *Proc. ICLR*, 2019.
- [140] W.-N. Hsu, D. F. Harwath, C. Song, and J. R. Glass, “Text-free image-to-speech synthesis using learned segmental units,” in *Proc. ACL/IJCNLP*, 2021, p. 25 pages.
- [141] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *Proc. ICML*, Lille, France, 2015, pp. 2048–2057.
- [142] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions,” in *Proc. ICASSP*, 2018, pp. 4779–4783.
- [143] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, “Learning sight from sound: Ambient sound provides supervision for visual learning,” *Int. J. Comput. Vision*, vol. 126, no. 10, p. 1120–1137, 2018.
- [144] P. Morgado, N. Vasconcelos, and I. Misra, “Audio-visual instance discrimination with cross-modal agreement,” in *Proc. CVPR*, Nashville, TN, USA, 2021, pp. 12475–12486.
- [145] A. Rouditchenko, H. Zhao, C. Gan, J. McDermott, and A. Torralba, “Self-supervised audio-visual co-segmentation,” in *Proc. ICASSP*, Brighton, UK, 2019, pp. 2357–2361.
- [146] E. Tzinis, S. Wisdom, A. Jansen, S. Hershey, T. Remez, D. Ellis, and J. R. Hershey, “Into the wild with audioscope: Unsupervised audio-visual separation of on-screen sounds,” in *Proc. ICLR*, 2021, p. 9 pages.
- [147] T. Afouras, A. Owens, J. S. Chung, and A. Zisserman, “Self-supervised learning of audio-visual objects from video,” in *Proc. ECCV*, 2020, pp. 208–224.
- [148] A. Shukla, S. Petridis, and M. Pantic, “Learning speech representations from raw audio by joint audiovisual self-supervision,” in *Proc. ICML*, 2020, p. 8 pages.
- [149] A. Shukla, K. Vougioukas, P. Ma, S. Petridis, and M. Pantic, “Visually guided self supervised learning of speech representations,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 6299–6303.
- [150] A. Shukla, S. Petridis, and M. Pantic, “Does visual self-supervision improve learning of speech representations for emotion recognition,” *IEEE Transactions on Affective Computing*, 2021.
- [151] J.-B. Alayrac, A. Recasens, R. Schneider, R. Arandjelović, J. Ramapuram, J. De Fauw, L. Smaira, S. Dieleman, and A. Zisserman, “Self-supervised multi modal versatile networks,” in *Proc. NeurIPS*, 2020, p. 13 pages.
- [152] X. Favory, K. Drossos, T. Virtanen, and X. Serra, “COALA: Co-aligned autoencoders for learning semantically enriched audio representations,” *Proc. ICML*, p. 8 pages, 2020.
- [153] A. Haque, M. Guo, P. Verma, and L. Fei-Fei, “Audio-linguistic embeddings for spoken sentences,” in *Proc. ICASSP*, Brighton, UK, 2019, pp. 7355–7359.
- [154] S. Khurana, A. Laurent, and J. Glass, “Cstnet: Contrastive speech translation network for self-supervised speech representation learning,” 2020.
- [155] C. Sun, F. Baradel, K. Murphy, and C. Schmid, “Learning video representations using contrastive bidirectional transformer,” *arXiv preprint arXiv:1906.05743*, 2019.
- [156] H. Akbari, L. Yuan, R. Qian, W.-H. Chuang, S.-F. Chang, Y. Cui, and B. Gong, “VATT: Transformers for multimodal self-supervised learning from raw video, audio and text,” in *Proc. NeurIPS*, 2021, p. 20 pages.
- [157] D. Jiang, X. Lei, W. Li, N. Luo, Y. Hu, W. Zou, and X. Li, “Improving transformer-based speech recognition using unsupervised pre-training,” *arXiv preprint arXiv:1910.09932*, 2019.
- [158] M. Neumann and N. T. Vu, “Improving speech emotion recognition with unsupervised representation learning on unlabeled speech,” in *Proc. ICASSP*, Brighton, UK, 2019, pp. 7390–7394.
- [159] S. Siriwardhana, A. Reis, R. Weerasekera, and S. Nanayakkara, “Jointly fine-tuning “bert-like” self supervised models to improve multimodal speech emotion recognition,” in *Proc. INTERSPEECH*, Shanghai, China, 10 2020, pp. 3755–3759.
- [160] H. Nguyen, F. Bougares, N. Tomashenko, Y. Estève, and laurent besacier, “Investigating self-supervised pre-training for end-to-end speech translation,” in *Proc. ICML*, 2020, p. 7 pages.
- [161] J. Engel, R. Swavely, L. H. Hantrakul, A. Roberts, and C. Hawthorne, “Self-supervised pitch detection by inverse audio synthesis,” in *Proc. ICML*, 2020, p. 9 pages.
- [162] “The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling,” in *Proc. NeurIPS*, 2020.
- [163] “SUPERB: Speech processing universal performance benchmark,” in *Proc. INTERSPEECH*, 2021.
- [164] “LeBenchmark: A reproducible framework for assessing self-supervised representation learning from speech,” in *Proc. INTERSPEECH*, 2020.
- [165] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayı, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, “Libri-light: A benchmark for asr with limited or no supervision,” in *Proc. ICASSP*, Barcelona, Spain, 2020, pp. 7669–7673.
- [166] J. L. Suárez, S. García, and F. Herrera, “A tutorial on distance metric learning: Mathematical foundations, algorithms, experimental analysis, prospects and challenges,” *Neurocomputing*, vol. 425, pp. 300–322, 2021.
- [167] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, “Unsupervised feature learning via non-parametric instance discrimination,” in *Proc. CVPR*, Salt Lake City, UT, USA, 2018, pp. 3733–3742.
- [168] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, “ELECTRA: Pre-training text encoders as discriminators rather than generators,” in *Proc. ICLR*, 2020, p. 18 pages.
- [169] C. Wang, Y. Wu, Y. Qian, K. Kumatori, S. Liu, F. Wei, M. Zeng, and X. Huang, “Unispeech: Unified speech representation learning with labeled and unlabeled data,” in *Proceedings of the 38th Proc. ICML, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 10937–10947.