data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language

Alexei Baevski [1] **Wei-Ning Hsu** [1] **Qiantong Xu** [2] **Arun Babu** [1] **Jiatao Gu** [1] **Michael Auli** [1]

Abstract

While the general idea of self-supervised learning is identical across modalities, the actual algorithms and objectives differ widely because they were developed with a single modality in mind. To get us closer to general self-supervised learning, we present data2vec, a framework that uses the same learning method for either speech, NLP or computer vision. The core idea is to predict latent representations of the full input data based on a masked view of the input in a self-distillation setup using a standard Transformer architecture. Instead of predicting modality-specific targets such as words, visual tokens or units of human speech which are local in nature, data2vec predicts contextualized latent representations that contain information from the entire input. Experiments on the major benchmarks of speech recognition, image classification, and natural language understanding demonstrate a new state of the art or competitive performance to predominant approaches. Models and code are available at www.github.com/pytorch/fairseq/tree/master/examples/data2vec.

1. Introduction

Self-supervised learning builds representations of data without human annotated labels which led to significant advances in natural language processing (NLP; Peters et al. 2018; Radford et al. 2018; Devlin et al. 2019; Brown et al. 2020), speech processing (van den Oord et al., 2018; Schneider et al., 2019; Baevski et al., 2020b) as well as computer vision (Chen et al., 2020; 2021b; Caron et al., 2021; Bao et al., 2021; He et al., 2021). Self-supervised representations have even enabled completely unsupervised learning in tasks such as machine translation (Lample et al., 2018) and speech recognition (Baevski et al., 2021).

Research in self-supervised algorithms has focused on individual modalities which results in specific designs and learning biases. For example, in speech processing, there is no vocabulary of speech units over which we can define a self-supervised learning task such as words in NLP[1] and therefore several prominent models are equipped with mechanisms to learn an inventory of speech units (Baevski et al., 2020b; Hsu et al., 2021). A similar problem exists for computer vision, where researchers either learn discrete visual tokens (Radford et al., 2021a; Bao et al., 2021), regress the input (He et al., 2021) or learn representations invariant to data augmentation (Chen et al., 2020; Grill et al., 2020; Caron et al., 2021).

While learning biases are certainly helpful, it is often unclear whether they will generalize to other modalities. Moreover, leading theories on the biology of learning (Friston & Kiebel, 2009; Friston, 2010) imply that humans likely use similar learning processes to understand the visual world as they do for language. Relatedly, general neural network architectures have been shown to perform very well compared to modality-specific counterparts (Jaegle et al., 2021b).

In an effort to get closer to machines that learn in general ways about the environment, we designed data2vec, a framework for general self-supervised learning that works for images, speech and text where the learning objective is identical in each modality. The present work unifies the learning algorithm but still learns representations individually for each modality. We hope that a single algorithm will make future multi-modal learning simpler, more effective and lead to models that understand the world better through multiple modalities.

Our method combines masked prediction (Devlin et al., 2019; Baevski et al., 2020b; Bao et al., 2021) with the learning of latent target representations (Grill et al., 2020; Caron et al., 2021) but generalizes the latter by using multiple network layers as targets and shows that this approach works across several modalities. Specifically, we train an off-the-

[1]Meta AI [2]SambaNova, work done while at Meta AI. Correspondence to: Alexei Baevski <abaevski@fb.com>, Michael Auli <michaelauli@fb.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

[1]This is true for many languages but not for certain Asian languages.

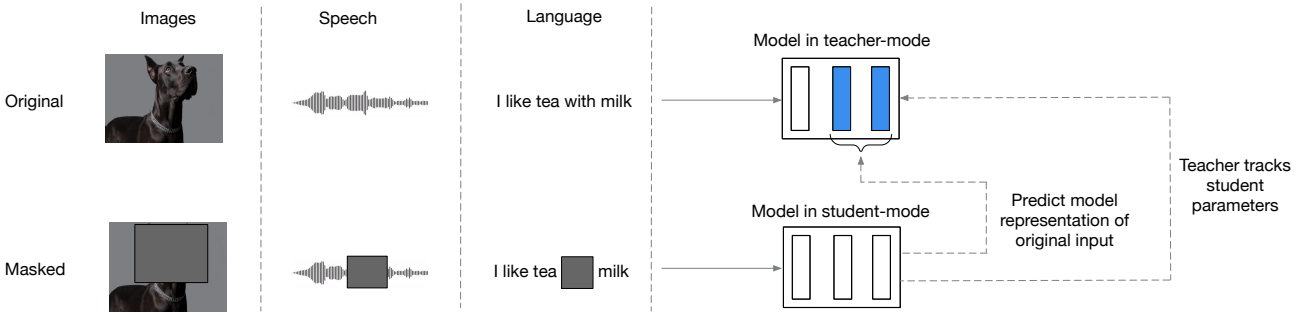


Figure 1. Illustration of how data2vec follows the same learning process for different modalities. The model first produces representations of the original input example (teacher mode) which are then regressed by the same model based on a masked version of the input. The teacher parameters are an exponentially moving average of the student weights. The student predicts the average of K network layers of the teacher (shaded in blue).

shelf Transformer network (Vaswani et al., 2017) which we use either in teacher or student mode (Illustration in Figure 1): we first build representations of the full input data whose purpose is to serve as targets in the learning task (teacher mode). Next, we encode a masked version of the input sample with which we predict the full data representations (student mode). The weights of the teacher are an exponentially decaying average of the student (He et al., 2019; Grill et al., 2020; Caron et al., 2021). Since different modalities have vastly different inputs, e.g., pixels vs. words, we use modality-specific feature encoders and masking strategies from the literature.

Since our method works with the latent network representations of the learner itself, it can be seen as a simplification of many modality-specific designs such as learning a fixed set of visual tokens (Radford et al., 2021a; van den Oord et al., 2017), or normalization of the input to create suitable targets (He et al., 2021), or the learning of a vocabulary of discrete speech units (Baevski et al., 2020b; Hsu et al., 2021). Moreover, our target representations are *continuous* and *contextualized*, through the use of self-attention, which makes them richer than a fixed set of targets and/or targets based on local context such as used in most prior work.

Experimental results show data2vec to be effective in all three modalities, setting a new state of the art for ViT-B with single models and ViT-L on ImageNet-1K, improving over the best prior work in speech processing on speech recognition (Baevski et al., 2020b; Hsu et al., 2021) and outperforming a like for like RoBERTa baseline on the GLUE natural language understanding benchmark (Liu et al., 2019).

2. Related Work

Self-supervised Learning in Computer Vision. Unsupervised pre-training for computer vision has been a very active area of research with methods contrasting representations of augmentations of the same image, entirely different images (Chen et al., 2020; Grill et al., 2020; Caron et al., 2021; Chen et al., 2021b) as well as online clustering (Caron et al., 2020). Similar to our work, both BYOL (Grill et al., 2020) and DINO (Caron et al., 2021) regress neural network representations of a momentum encoder, but our work differs in that it uses a masked prediction task and we regress multiple neural network layer representations instead of just the top layer which we find to be more effective. Moreover, data2vec works for multiple modalities.

The most recent work focuses on training vision Transformers (Dosovitskiy et al., 2020) with masked prediction objectives (Bao et al., 2021; He et al., 2021; Xie et al., 2021) whose performance surpasses supervised-only training on ImageNet-1K. Several of these methods predict visual tokens (Bao et al., 2021; He et al., 2021; Dong et al., 2022) learned in a separate step before pre-training (van den Oord et al., 2017; Ramesh et al., 2021), during pretraining (Zhou et al., 2021), and others directly predict the input pixels (He et al., 2021; Xie et al., 2021).

Instead, data2vec predicts the latent representations of the input data. Another difference to this body of work is that the latent target representations are *contextualized*, incorporating relevant features from the entire image instead of targets which contain information isolated to the current patch, such as visual tokens or pixels.

Self-supervised Learning in NLP. Pre-training has been very successful in advancing natural language understanding (McCann et al., 2017; Peters et al., 2018; Radford et al., 2018; Baevski et al., 2019; Devlin et al., 2019; Yang et al., 2019; Brown et al., 2020). The most prominent model is BERT (Devlin et al., 2019) which solves a masked prediction task where some of the input tokens are blanked out in order to be predicted given the remaining input. For many languages it is easy to determine word boundaries and most

methods therefore predict word or sub-word units for pre-training. There is also work on knowledge distillation to obtain smaller BERT-style models, both for pre-training and fine-tuning (Jiao et al., 2020).

Compared to prior NLP algorithms, data2vec does not predict discrete linguistic tokens such as words, sub-words or bytes but rather a continuous and contextualized representation. This has two advantages: first, the targets themselves are not predefined, nor is their number limited. This enables the model to adapt to a particular input example. Second, targets are *contextualized*, taking context information into account. This is unlike BERT-style models which learn a single embedding for each target which needs to fit all instances of a particular target in the data.

Self-supervised Learning in Speech. Work in self-supervised learning for speech includes autoregressive models (van den Oord et al., 2018; Schneider et al., 2019; Baevski et al., 2020a; Chung et al., 2019) as well as bi-directional models (Baevski et al., 2020b; Hsu et al., 2021; Ao et al., 2021; Chen et al., 2021a). Two prominent models, wav2vec 2.0 and HuBERT are based on predicting discrete units of speech, either learned jointly during pre-training (Baevski et al., 2020b), or in an iterative pipeline approach (Hsu et al., 2021) where pre-training and clustering alternate.[2] Another line of work directly reconstructs the input features (Eloff et al., 2019; Liu et al., 2021).

In comparison to wav2vec 2.0, data2vec directly predicts contextualized latent representations without quantization. HuBERT discretizes representations from different layers across iterations and predicts these discretized units whereas data2vec predicts the average over multiple layers. Similar to other modalities, there is work on distilling larger self-supervised models into smaller models but primarily for the purpose of efficiency (Chang et al., 2021).

Multimodal Pre-training. There has been a considerable body of research on learning representations of multiple modalities simultaneously often using paired data (Aytar et al., 2017; Radford et al., 2021b; Wang et al., 2021; Singh et al., 2021) with the aim to produce cross-modal representations which can perform well on multi-modal tasks and with modalities benefiting from each other through joint training (Alayrac et al., 2020; Akbari et al., 2021) with recent methods exploring few-shot learning (Tsimpoukelli et al., 2021). Our work does not perform multimodal training but aims to unifiy the learning objective for self-supervised learning in different modalities. We hope that this will enable better multimodal representations in the future.

[2]Quantization is optional for wav2vec 2.0 (Baevski et al., 2020b; Zhang et al., 2020) but helpful for noisy speech (Chung et al., 2021).

3. Method

data2vec is trained by predicting the model representations of the full input data given a partial view of the input (Figure 1). We first encode a masked version of the training sample (model in *student mode*) and then construct training targets by encoding the unmasked version of the input with the same model but when parameterized as an exponentially moving average of the model weights (model in *teacher mode*; Grill et al. 2020; Caron et al. 2021). The target representations encode all of the information in the training sample and the learning task is for the student to predict these representations given a partial view of the input.

3.1. Model Architecture

We use the standard Transformer architecture (Vaswani et al., 2017) with a modality-specific encoding of the input data borrowed from prior work:[3] for computer vision, we use the ViT-strategy of encoding an image as a sequence of patches, each spanning 16x16 pixels, input to a linear transformation (Dosovitskiy et al., 2020; Bao et al., 2021). Speech data is encoded using a multi-layer 1-D convolutional neural network that maps 16 kHz waveform to 50 Hz representations (Baevski et al., 2020b). Text is pre-processed to obtain sub-word units (Sennrich et al., 2016; Devlin et al., 2019), which are then embedded in distributional space via learned embedding vectors. We detail these methods below (§4).

3.2. Masking

After the input sample has been embedded as a sequence of tokens, we mask part of these units by replacing them with a learned MASK embedding token and feed the sequence to the Transformer network. For computer vision, we follow the block-wise masking strategy of Bao et al. (2021), for speech we mask spans of latent speech representations (Baevski et al., 2020b) and for language we mask tokens (Devlin et al., 2019); §4 details each strategy.

3.3. Training Targets

The model is trained to predict the model representations of the original unmasked training sample based on an encoding of the masked sample. We predict model representations only for time-steps which are masked. The representations we predict are *contextualized representations*, encoding the particular time-step but also other information from the sample due to the use of self-attention in the Transformer network.[4] This is an important difference to BERT (De-

[3]While we used Transformer networks, alternative architectures may be equally applicable.

[4]In preliminary experiments, we found that additional context information for the targets was helpful since masking some of the time-steps when in teacher mode resulted in lower accuracy.

vlin et al., 2019), wav2vec 2.0 (Baevski et al., 2020b) or BEiT, MAE, SimMIM, and MaskFeat (Bao et al., 2021; He et al., 2021; Xie et al., 2021; Wei et al., 2021) which predict targets lacking contextual information. Below, we detail how we parameterize the teacher which predicts the network representations that will serve as targets as well as how we construct the final target vectors to be predicted by the model in student-mode.

Teacher Parameterization. The encoding of the unmasked training sample is parameterized by an exponentially moving average (EMA) of the model parameters (θ; Tarvainen & Valpola 2018; Grill et al. 2020; Caron et al. 2021) where the weights of the model in target-mode Δ are:

$$\Delta \leftarrow \tau\Delta + (1-\tau)\,\theta$$

We use a schedule for τ that linearly increases this parameter from τ_0 to the target value τ_e over the first τ_n updates after which the value is kept constant for the remainder of training. This strategy results in the teacher being updated more frequently at the beginning of training, when the model is random, and less frequently later in training, when good parameters have already been learned. We found it more efficient and slightly more accurate to share the parameters of the feature encoder and the positional encoder between the teacher and student networks.

Targets. Training targets are constructed based on the output of the top K blocks of the teacher network for time-steps which are masked in student-mode.[5] The output of block l at time-step t is denoted as a_t^l. We apply a normalization to each block to obtain $\hat{a}_t^l$ before averaging the top K blocks $y_t = \frac{1}{K}\sum_{l=L-K+1}^{L}\hat{a}_t^l$ for a network with L blocks in total to obtain the training target y_t for time-step t. This creates training targets that are to be regressed by the model when in student mode. In preliminary experiments we found that averaging performed as well as predicting each block separately with a dedicated projection while enjoying the advantage of being more efficient.

Normalizing the targets helps prevent the model from collapsing into a constant representation for all time-steps and it also prevents layers with high norm to dominate the target features. For speech representations, we use instance normalization (Ulyanov et al., 2016) without any learned parameters over the current input sample since neighboring representations are highly correlated due to the small stride over the input data, while for NLP and vision we found parameter-less layer normalization (Ba et al., 2016) to be sufficient. Variance-Invariance-Covariance regularization (Bardes et al., 2021) also addresses this problem but we found the above strategy to perform well and it does not introduce additional hyper-parameters.

[5] We generally use the output of the FFN prior to the last residual connection in each block as target. See the ablation in §5.4.

3.4. Objective

Given contextualized training targets y_t, we use a Smooth L1 loss to regress these targets:

$$\mathcal{L}(y_t, f_t(x)) = \begin{cases} \frac{1}{2}(y_t - f_t(x))^2/\beta & |y_t - f_t(x)| \le \beta \\ (|y_t - f_t(x)| - \frac{1}{2}\beta) & \text{otherwise} \end{cases}$$

where β controls the transition from a squared loss to an L_1 loss, depending on the size of the gap between the target y_t and the model prediction $f_t(x)$ at time-step t. The advantage of this loss is that it is less sensitive to outliers, however, we need to tune the setting of β.

4. Experimental Setup

We experiment with two model sizes: data2vec Base and data2vec Large, containing either $L = 12$ or $L = 24$ Transformer blocks with $H = 768$ or $H = 1024$ hidden dimension (with $4 \times H$ feed-forward inner-dimension). EMA updates are performed in fp32 for numerical stability (Manohar et al., 2021).

4.1. Computer Vision

We embed images of 224x224 pixels as patches of 16x16 pixels (Dosovitskiy et al., 2020). Each patch is linearly transformed and a sequence of 196 representations is input to a standard Transformer. We follow BEiT (Bao et al., 2021) by masking blocks of multiple adjacent patches where each block contains at least 16 patches with a random aspect ratio. Different to their work, we found it more accurate to mask 60% of the patches instead of 40%. We use randomly applied resized image crops, horizontal flipping, and color jittering (Bao et al., 2021). We use the same modified image both in teacher mode and student mode.

ViT-B models are pre-trained for 800 epochs. As batch size we use 2,048 for ViT-B and 8,192 for ViT-L. We use Adam (Kingma & Ba, 2015) and a cosine schedule (Loshchilov & Hutter, 2016) with a single cycle where we warm up the learning rate for 40 epochs to 0.002 for ViT-B and for 80 epochs to 0.001 for ViT-L after which the learning rate is annealed following the cosine schedule. For ViT-B and ViT-L, we use $\beta = 2$, $K = 6$ and $\tau = 0.9998$ as a constant value with no schedule which worked well. We use stochastic depth with rate 0.2 (Huang et al., 2016). For ViT-L, we train for 1,600 epochs in total, the first 800 epochs use $\tau = 0.9998$, we then reset the learning rate schedule and the teacher weights to the student and continue for another 800 epochs with $\tau = 0.9999$.

For image classification we mean-pool the output of the

last Transformer block and input it to a softmax-normalized classifier. We fine-tune ViT-B for 100 epochs and ViT-L for 50 epochs using Adam and a cosine schedule where we warmup up the learning rate for 20 epochs to 0.004 for ViT-B and for 5 epochs to 0.004 for ViT-L after which the learning rate follows the cosine schedule. We build on the open source implementation of BEiT (Bao et al., 2021).

4.2. Speech Processing

Models are implemented in fairseq (Ott et al., 2019) and take as input 16 kHz waveform which is processed by a feature encoder (Baevski et al., 2020b) containing seven temporal convolutions with 512 channels, strides (5,2,2,2,2,2,2) and kernel widths (10,3,3,3,3,2,2). This results in an encoder output frequency of 50 Hz with a stride of about 20ms between each sample, and a receptive field of 400 input samples or 25ms of audio. The raw waveform input to the encoder is normalized to zero mean and unit variance.

The masking strategy for the Base model is also identical to Baevski et al. (2020b): we sample $p = 0.065$ of all time-steps to be starting indices and mask the subsequent ten time-steps. This results in approximately 49% of all time-steps to be masked for a typical training sequence. During pre-training we linearly anneal τ using $\tau_0 = 0.999$, $\tau_e = 0.9999$ and $\tau_n = 30,000$, average the top $K = 8$ blocks as targets and found a simple L2 loss to work well.

We optimize with Adam (Kingma & Ba, 2015), with a peak learning rate of 5×10^{-4} for data2vec Base. The Base model uses a tri-stage scheduler which linearly warms up the learning rate over the first 3% of updates, holds it for 90% of updates and then linearly decays it over the remaining 7%. We train data2vec Base for 400K updates with a batch size of 63 minutes of audio (61M frames). We follow the fine-tuning regime of wav2vec 2.0 (Baevski et al., 2020b) whose hyper-parameters depend on the labeled data setup.

4.3. Natural Language Processing

We build on the BERT re-implementation RoBERTa (Liu et al., 2019) available in fairseq (Ott et al., 2019). The input data is tokenized using a byte-pair encoding (Sennrich et al., 2016) of 50K types and the model learns an embedding for each type (Devlin et al., 2019; Liu et al., 2019) Once the data is embedded, we apply the BERT masking strategy to 15% of uniformly selected tokens: 80% are replaced by a learned mask token, 10% are left unchanged and 10% are replaced by randomly selected vocabulary token; we do not use the next-sentence prediction task. We also consider the wav2vec 2.0 strategy of masking spans of four tokens.

For pre-training we use $\tau_0 = 0.999$, $\tau_e = 0.9999$ and $\tau_n = 100,000$, $K = 10$ and set $\beta = 4$. The model is optimized with Adam over 1M updates using a tri-stage learning rate schedule (5%, 80% and 15% of updates for warm-up, holding and linearly decaying, respectively). The peak learning rate is 2×10^{-4}. We train on 16 GPUs with a total batch size of 256 sequences and each sequence is up to 512 tokens. For downstream tasks, we fine-tune the pre-trained model with four different learning rates (1×10^{-5}, 2×10^{-5}, 3×10^{-5}, 4×10^{-5}) and choose the one which performs best across all considered NLP downstream tasks.

Table 1. Computer vision: top-1 validation accuracy on ImageNet-1K with ViT-B and ViT-L models. data2vec ViT-B was trained for 800 epochs and ViT-L for 1,600 epochs. We distinguish between individual models and setups composed of multiple models (BEiT/PeCo train separate visual tokenizers and PeCo also distills two MoCo-v3 models).

	ViT-B	ViT-L
Multiple models		
BEiT (Bao et al., 2021)	83.2	85.2
PeCo (Dong et al., 2022)	84.5	86.5
Single models		
MoCo v3 (Chen et al., 2021b)	83.2	84.1
DINO (Caron et al., 2021)	82.8	-
MAE (He et al., 2021)	83.6	85.9
SimMIM (Xie et al., 2021)	83.8	-
iBOT (Zhou et al., 2021)	83.8	-
MaskFeat (Wei et al., 2021)	84.0	85.7
data2vec	84.2	86.6

5. Results

5.1. Computer Vision

To evaluate our approach for computer vision, we pre-train data2vec on the images of the ImageNet-1K training set (Deng et al., 2009) and fine-tune the resulting model for image classification using the labeled data of the same benchmark (§4.1). Following standard practice, models are evaluated in terms of top-1 accuracy on the validation set. We distinguish between results based on a single self-supervised model, and results which train a separate visual tokenizer on additional data (Bao et al., 2021) or distill other self-supervised models (Dong et al., 2022).

Table 1 shows that data2vec outperforms prior work with ViT-B and ViT-L in the single model setting and all prior work for ViT-L. Predicting contextualized latent representations in a masked prediction setup can perform very well compared to approaches which predict local targets such as the original input pixels (He et al., 2021; Xie et al., 2021), engineered image features (Wei et al., 2021) or visual tokens (Bao et al., 2021). It also outperforms prior self-distillation methods (Caron et al., 2021) which regressed

Table 2. Speech processing: word error rate on the Librispeech test-other test set when fine-tuning pre-trained models on the Libri-light low-resource labeled data setups (Kahn et al., 2020) of 10 min, 1 hour, 10 hours, the clean 100h subset of Librispeech and the full 960h of Librispeech. Models use the 960 hours of audio from Librispeech (LS-960) as unlabeled data. We indicate the language model used during decoding (LM). Results for all dev/test sets and other LMs can be found in the supplementary material (Table 6).

	Unlabeled data	LM	Amount of labeled data				
			10m	1h	10h	100h	960h
Base models							
wav2vec 2.0 (Baevski et al., 2020b)	LS-960	4-gram	15.6	11.3	9.5	8.0	6.1
HuBERT (Hsu et al., 2021)	LS-960	4-gram	15.3	11.3	9.4	8.1	-
WavLM (Chen et al., 2021a)	LS-960	4-gram	-	10.8	9.2	7.7	-
data2vec	LS-960	4-gram	12.3	9.1	8.1	6.8	5.5
Large models							
wav2vec 2.0 (Baevski et al., 2020b)	LS-960	4-gram	10.3	7.1	5.8	4.6	3.6
HuBERT (Hsu et al., 2021)	LS-960	4-gram	10.1	6.8	5.5	4.5	3.7
WavLM (Chen et al., 2021a)	LS-960	4-gram	-	6.6	5.5	4.6	-
data2vec	LS-960	4-gram	8.4	6.3	5.3	4.6	3.7

the final layer of the student network while inputting two different augmented versions of an image to the student and teacher networks.

5.2. Speech and Audio Processing

For speech processing, we pre-train data2vec on the 960 hours of speech audio data from Librispeech (LS-960). This dataset contains relatively clean speech audio from read audiobooks in English and is a standard benchmark in the speech community. To get a sense of performance in different resource settings, we fine-tune models for automatic speech recognition using different amounts of labeled data, ranging from just 10 minutes to 960 hours. We also compare to other work from the literature, including wav2vec 2.0 (Baevski et al., 2020b) and HuBERT (Hsu et al., 2021), two popular algorithms for speech representation learning relying on discrete units of speech.

Table 2 shows improvements for most labeled data setups with the largest gains for 10 minutes of labeled data (20% relative WER improvement) for the Base models. For Large models, there are strong improvments for the smallest labeled data setups, and comparable performance for the resource-rich settings of 100 hours and 960 hours of labeled data where performance is generally saturating for many models (Zhang et al., 2020; Chung et al., 2021). Our results suggest that learning discrete units is not required when rich contextualized targets are used and that learning contextualized targets during pre-training improves performance.

To further validate our approach for speech, we also trained a model on the AudioSet benchmark (Gemmeke et al., 2017). The pre-training setup is the same as Librispeech, but we set $K = 12$ and train for 200K updates with a batch size of 94.5 minutes of audio. We apply DeepNorm (Wang et al., 2022) and layer normalization of the targets to stabilize training. Finetuning is done on the balanced subset over 13k updates with a batch size of 21.3 minutes. Similar to Srivastava et al. (2021), we use linear softmax pooling (Wang et al., 2018b) and mixup (Tokozume et al., 2017) with probability 0.7. We add a single linear projection layer into 527 audioset classes and set the projection learning rate to $2e-4$. The pre-trained parameters are trained with a learning rate of $3e-5$ and we use masking during fine-tuning similar to Baevski et al. (2020b) with $p = 0.45$ and mask length of 4. The results (Table 3) show that data2vec can outperform a comparable setup that uses the same pre-training and fine-tuning data.

Table 3. Audio event classification: mean average precision (mAP) on the eval set of AudioSet when pre-training on all of AudioSet (AS) and/or Librispeech (LS) and fine-tuning on the 20K subset.

	Pretrain data	mAP
SSAST (Gong et al., 2021)	AS, LS	31.0
MaskSpec (Chong et al., 2022)	AS	32.3
MAE-AST (Baade et al., 2022)	AS, LS	30.6
data2vec	AS	34.5

5.3. Natural Language Processing

To get a sense of how data2vec performs for language, we adopt the same training setup as BERT (Devlin et al., 2019) by pre-training on the Books Corpus (Zhu et al., 2015) and English Wikipedia data over 1M updates and a batch size of 256 sequences. We evaluate on the General Language

Table 4. Natural language processing: GLUE results on the development set for single-task fine-tuning of individual models. For MNLI we report accuracy on both the matched and unmatched dev sets, for MRPC and QQP, we report the unweighted average of accuracy and F1, for STS-B the unweighted average of Pearson and Spearman correlation, for CoLA we report Matthews correlation and for all other tasks we report accuracy. BERT Base results are from Wu et al. (2020) and our baseline is RoBERTa re-trained in a similar setup as BERT. We also report results with wav2vec 2.0 style masking of spans of four BPE tokens with no unmasked tokens or random targets.

	MNLI	QNLI	RTE	MRPC	QQP	STS-B	CoLA	SST	Avg.
BERT (Devlin et al., 2019)	84.0/84.4	89.0	61.0	86.3	89.1	89.5	57.3	93.0	80.7
Baseline (Liu et al., 2019)	84.1/83.9	90.4	69.3	89.0	89.3	88.9	56.8	92.3	82.5
data2vec	83.2/83.0	90.9	67.0	90.2	89.1	87.2	62.2	91.8	82.7
+ wav2vec 2.0 masking	82.8/83.4	91.1	69.9	90.0	89.0	87.7	60.3	92.4	82.9

Understanding Evaluation (GLUE) benchmark (Wang et al., 2018a) which includes tasks for natural language inference (MNLI, QNLI, RTE), sentence similarity (MRPC, QQP and STS-B), grammaticality (CoLA), and sentiment analysis (SST-2).[6] We fine-tune data2vec separately on the labeled data provided by each task and report the average accuracy on the development sets over five fine-tuning runs. We compare to the published BERT results as well as to the results we obtain by retraining RoBERTa in the current setup (Baseline; Liu et al. 2019) which provides a more suitable baseline to data2vec since we build on their open source code.

The results (Table 4) show that data2vec outperforms the RoBERTa baseline. When we mask spans of four BPE tokens with masking probability 0.35 (Baevski et al., 2020b), then results improve further.[7] This strategy does not leave tokens unmasked or uses random targets as for BERT (§4.3).

To our knowledge this is the first successful pre-trained NLP model which does not use discrete units (words, subwords, characters or bytes) as the training target. Instead, the model predicts a *contextualized latent representation* emerging from self-attention over the entire unmasked text sequence. This enables a learning task where the model needs to predict targets with specific properties of the current text sequence rather than representations which are generic to every text sequence in which the particular discrete unit occurs. Moreover, the set of training targets is not fixed, i.e., not a closed vocabulary, and the model can choose to define new targets as it sees fit, akin to an open vocabulary setting.

[6]MNLI (Multi Genre Natural Language Inference; Williams et al. 2018), Stanford Question Answering Dataset (QNLI; Rajpurkar et al. 2016), Recognizing Textual Entailment (RTE;Dagan et al. 2006; Haim et al. 2006; Giampiccolo et al. 2007; Bentivogli et al. 2009), and we exclude Winograd NLI task from our results similar to Devlin et al. (2019), Microsoft Research Paragraph Corpus (MRPC; Dolan & Brockett 2005), Quora Question Pairs benchmark (QQP), and the Semantic Textual Similarity Benchmark (STS-B; Cer et al. 2018), Corpus of Linguistic Acceptability (CoLA; Warstadt et al. 2018), Stanford Sentiment Treebank (SST-2; Socher et al. 2013)

[7]We used a cosine learning rate schedule for this result.

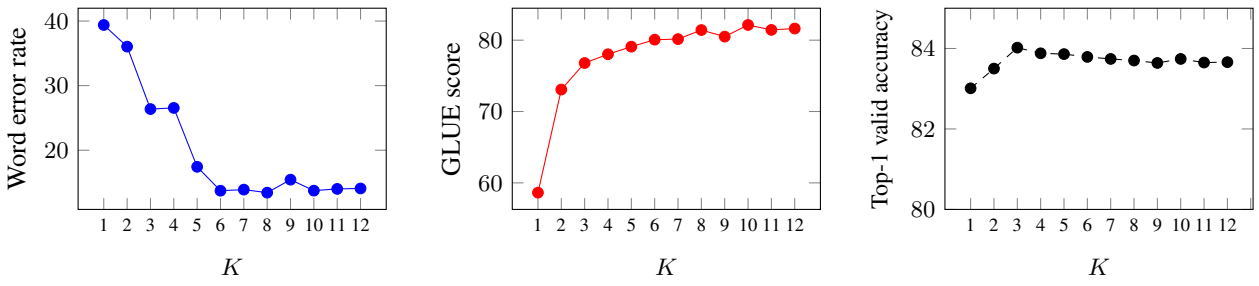
5.4. Ablations

Layer-averaged Targets. One of the main differences of our method compared to BYOL is the use of targets which are based on averaging multiple layers from the teacher network (§3.3). This idea was partly inspired by the fact that the top layers of wav2vec 2.0 do not perform as well for downstream tasks as layers in the middle of the network (Baevski et al., 2021; Pasad et al., 2021).

In the next experiment, we measure performance for all three modalities when averaging $K = 1, \ldots, 12$ layers where $K = 1$ corresponds to predicting only the top layer similar to BYOL. For faster experimental turn-around, we train Base models with $L = 12$ layers in total. For speech, we pre-train for 200K updates on Librispeech, fine-tune on the 10 hour labeled split of Libri-light (Kahn et al., 2019) and report word error rate without a language model on dev-other. For NLP, we report the average GLUE score on the validation set (§5.3) and for computer vision we pre-train models for 300 epochs and report the top-1 accuracy on ImageNet (§5.1).

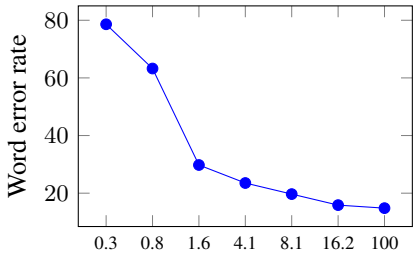
Figure 2 shows that targets based on multiple layers improves over using only the top layer ($K = 1$) for all modalities. Using all layers is generally a good choice and only slightly worse than a carefully tuned value of K. Neural networks build features over multiple layers and different types of features are extracted at different layers. Using features from multiple layers enriches the self-supervised task and improves accuracy.

Target Contextualization. Teacher representations are based on self-attention over the entire input which results in contextualized targets. This distinguishes data2vec from other self-supervised approaches which construct a learning task by predicting or reconstructing local parts of the input (§2). This poses the natural question of whether contextualized targets are required for data2vec to work well.

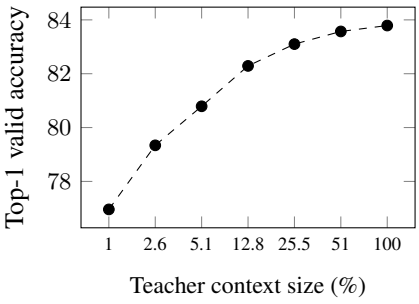


(a) Speech (b) NLP (c) Vision

Figure 2. Predicting targets which are the average of multiple layers is more robust than predicting only the top most layer ($K = 1$) for most modalities. We show the performance of predicting the average of K teacher layer representations (§3.3). The effect is very pronounced for speech and NLP while for vision there is still a slight advantage of predicting more than a single layer.



(a) Speech



(b) Vision

Figure 3. Contextualized target representations improve accuracy. We mask all but a fraction of the data when constructing the target representations in the teacher for pre-training and report downstream performance for speech and vision. Time-steps for speech are more fine-grained (20ms patches) compared to vision (16x16 pixels per patch).

Table 5. Effect of using different features from the teacher model as targets: we compare using the output of the self-attention module, the feed-forward module (FFN) as well as after the final residual connection (FFN + residual) and layer normalization (End of block). Results are not directly comparable to the main results since we use a reduced setup (§5.4).

Layer	WER
self-attention	100.0
FFN	13.1
FFN + residual	14.8
End of block	14.5

In order to get a better sense of this, we construct target representations which do not have access to the entire input sample but rather only a pre-defined fraction of it. Concretely, we restrict the self-attention mechansim of the teacher to only be able to access a portion of the input surrounding the current time-step. Once the model is trained, we fine-tune it so that it can access the full context size. Figure 3 shows that larger context sizes lead to better downstream performance. The best accuracy is achieved when the entire input sample is visible. This shows that richer target representations can indeed lead to better performance.

Target Feature Type. Transformers blocks contain several layers which can each serve as targets. To get a sense of how different layers impact performance, we pre-train speech models on Librispeech using different layers as target features. Table 5 shows that the output of the feed-forward network (FFN) block works best while the output of the self-attention block does not yield a usable model. We believe this is because the self-attention is before the residual connection and features are heavily biased towards other time-steps. This issue is alleviated by the use of FFN features since these also include the features before the self-attention.

6. Discussion

Modality-specific Feature Extractors and Masking. Our primary goal is to design a single learning mechanism for different modalities. Despite the unified learning regime, we still use modality-specific feature extractors and masking strategies. This makes sense given the vastly different nature of the input data: for example, in speech we learn from a very high resolution input (16 kHz waveform) which contains hundreds of thousands of samples for typical utterances. To process this, we apply a multilayer convolutional neural network to obtain a 50 Hz feature sequence. For NLP, word input sequences have vastly lower resolution which can be directly embedded in distributional space via a lookup table. The type of data also impacts how we should mask the input to create a challenging learning task: removing individual words provides a sufficiently challenging task but for speech it is necessary to mask spans since adjacent audio samples are highly correlated with each other. Relatedly, there has been recent work on a Transformer architecture that can directly operate on the raw data of different modalities without modality-specific feature encoders (Jaegle et al., 2021b;a). Their work is focused on supervised learning for classification tasks and we believe that our work is complementary.

Structured and Contextualized Targets. One of the main differences of data2vec to most other masked prediction work (Devlin et al., 2019; Baevski et al., 2020b; Ling et al., 2020; Bao et al., 2021; He et al., 2021; Wei et al., 2021) is that the features of the training targets are contextualized since the features are built with self-attention over the entire unmasked input in teacher mode. And while BYOL (Grill et al., 2020) and DINO (Caron et al., 2021) also use latent target representations based on the entire input, their focus is on learning transformation-invariant representations instead of structural information within a sample.

One exception is HuBERT (Hsu et al., 2021) which builds a fixed set of discrete target units by clustering Transformer layer representations. In comparison, data2vec has no limitation on the number of target units. Instead of representing each instance of particular discrete target unit with the same set of features, data2vec can build target features that are specific to the current sequence.

For NLP, we believe data2vec is the first work that does not rely on pre-defined target units. Most other work uses either words, sub-words (Radford et al., 2018; Devlin et al., 2019), characters (Tay et al., 2021) or even bytes (Xue et al., 2021). Aside, defining word boundaries is not straightforward for some Asian languages. Contextualized targets enable integrating features from the entire sequence into the training target which provides a richer self-supervised task. Furthermore, the representation of each instance of a particular unit (word/sub-word/character/byte) can differ for the masked prediction task. This enables to associate a different meaning to a particular depending on the context it occurs in. It also relieves the model from the need to learn a single set of features for a target unit that fits all instances of this unit.

Representation Collapse. A common failure mode when learning targets is for similar representations for targets to be produced resulting in a trivial task (Jing et al., 2021). To deal with this, contrastive models such as wav2vec 2.0 (Baevski et al., 2020b) use the same target representation both as a positive and a negative example. BYOL (Grill et al., 2020) do not optimize the teacher parameters to minimize the loss and VicReg (Bardes et al., 2021) adds an explicit loss encouraging variance among different representations.

We found that collapse is most likely to happen in the following scenarios: First, the learning rate is too large or the learning rate warmup is too short which can often be solved by tuning the respective hyperparameters. Second, τ is too low which leads to student model collapse and is then propagated to the teacher. This can be addressed by tuning τ_0, τ_e and τ_n. Third, we found collapse to be more likely for modalities where adjacent targets are very correlated and where longer spans need to be masked, e.g., speech. We address this by promoting variance through normalizing target representations over the sequence or batch (Grill et al., 2020). For models where targets are less correlated, such as vision and NLP, momentum tracking is sufficient.

7. Conclusion

Recent work showed that uniform model architectures can be effective for multiple modalities (Jaegle et al., 2021b). Whereas we show that a single self-supervised learning regime can be effective for vision, speech and language. The key idea is to regress contextualized representations based on a partial input view. data2vec outperforms prior self-supervised work on ImageNet-1K for ViT-B and ViT-L single models, it improves over prior work on speech recognition for the low-resource Libri-light setups, and it outperforms RoBERTa on GLUE in the original BERT setup.

A single learning method for multiple modalities will make it easier to learn across modalities and future work may investigate tasks such as audio-visual speech recognition or cross-modal retrieval. Our approach still uses modality-specific input encoders and we adopt modality-specific masking strategies which future work may unify.

Acknowledgements

We thank Brenden Lake, Dhruv Batra, Marco Baroni and Laurens van der Maaten for helpful discussions.

References

Akbari, H., Yuan, L., Qian, R., Chuang, W.-H., Chang, S.-F., Cui, Y., and Gong, B. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text, 2021.

Alayrac, J.-B., Recasens, A., Schneider, R., Arandjelović, R., Ramapuram, J., Fauw, J. D., Smaira, L., Dieleman, S., and Zisserman, A. Self-supervised multimodal versatile networks, 2020.

Ao, J., Wang, R., Zhou, L., Liu, S., Ren, S., Wu, Y., Ko, T., Li, Q., Zhang, Y., Wei, Z., Qian, Y., Li, J., and Wei, F. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. *arXiv*, abs/2110.07205, 2021.

Aytar, Y., Vondrick, C., and Torralba, A. See, hear, and read: Deep aligned representations, 2017.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv*, abs/1607.06450, 2016.

Baade, A., Peng, P., and Harwath, D. Mae-ast: Masked autoencoding audio spectrogram transformer. *arXiv*, abs/2203.16691, 2022.

Baevski, A., Edunov, S., Liu, Y., Zettlemoyer, L., and Auli, M. Cloze-driven pretraining of self-attention networks. In *Proc. of EMNLP*, 2019.

Baevski, A., Schneider, S., and Auli, M. vq-wav2vec: Self-supervised learning of discrete speech representations. In *Proc. of ICLR*, 2020a.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. of NeurIPS*, 2020b.

Baevski, A., Hsu, W.-N., Conneau, A., and Auli, M. Unsupervised speech recognition. In *Proc. of NeurIPS*, 2021.

Bao, H., Dong, L., and Wei, F. Beit: BERT pre-training of image transformers. *arXiv*, abs/2106.08254, 2021.

Bardes, A., Ponce, J., and LeCun, Y. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv*, abs/2105.04906, 2021.

Bentivogli, L., Clark, P., Dagan, I., and Giampiccolo, D. The fifth pascal recognizing textual entailment challenge. In *Proc. of TAC*, 2009.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In *Proc. of NeurIPS*, 2020.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv*, abs/2006.09882, 2020.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. *arXiv*, abs/2104.14294, 2021.

Cer, D. M., Diab, M. T., Agirre, E., Lopez-Gazpio, I., and Specia, L. Semeval-2017 task 1: Semantic textual similarity - multilingual and cross-lingual focused evaluation. In *Proc. of SemEval*, 2018.

Chang, H.-J., wen Yang, S., and yi Lee, H. Distilhubert: Speech representation learning by layer-wise distillation of hidden-unit bert. *arXiv*, abs/2110.01900, 2021.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Wu, J., Zeng, M., and Wei, F. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *arXiv*, abs/2110.13900, 2021a.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv*, abs/2002.05709, 2020.

Chen, X., Xie, S., and He, K. An empirical study of training self-supervised vision transformers. *arXiv*, abs/2104.02057, 2021b.

Chong, D., Wang, H., Zhou, P., and Zeng, Q. Masked spectrogram prediction for self-supervised audio pre-training. *arXiv*, abs/2204.12768, 2022.

Chung, Y., Hsu, W., Tang, H., and Glass, J. R. An unsupervised autoregressive model for speech representation learning. *Proc. of Interspeech*, 2019.

Chung, Y.-A., Zhang, Y., Han, W., Chiu, C.-C., Qin, J., Pang, R., and Wu, Y. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. *arXiv*, abs/2108.06209, 2021.

Dagan, I., Glickman, O., and Magnini, B. The pascal recognizing textual entailment challenge. *Machine learning challenges, evaluating predictive uncertainty, visual object classification, and recognizing textual entailment*, pp. 177–190, 2006.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proc. of CVPR*, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proc. of NAACL*, 2019.

Dolan, W. B. and Brockett, C. Automatically constructing a corpus of sentential paraphrases. In *Proc. of IWP*, 2005.

Dong, X., Bao, J., Zhang, T., Chen, D., Zhang, W., Yuan, L., Chen, D., Wen, F., and Yu, N. Peco: Perceptual codebook for bert pre-training of vision transformers, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv*, abs/2010.11929, 2020.

Eloff, R., Nortje, A., van Niekerk, B., Govender, A., Nortje, L., Pretorius, A., Van Biljon, E., van der Westhuizen, E., van Staden, L., and Kamper, H. Unsupervised acoustic unit discovery for speech synthesis using discrete latent-variable neural networks. *arXiv*, abs/1904.07556, 2019.

Friston, K. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 2010.

Friston, K. and Kiebel, S. Predictive coding under the free-energy principle. *Philosophical transactions of the Royal Society: Biological sciences*, 2009.

Gemmeke, J. F., Ellis, D. P. W., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., Plakal, M., and Ritter, M. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. of ICASSP*, 2017.

Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, B. The pascal recognizing textual entailment challenge. *Proc. of the ACL-PASCAL workshop on textual entailment and paraphrasing*, 2007.

Gong, Y., Lai, C.-I. J., Chung, Y.-A., and Glass, J. Ssast: Self-supervised audio spectrogram transformer. *arXiv*, abs/2110.09784, 2021.

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

Haim, R. B., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., and Szpektor, I. The pascal recognising textual entailment challenge. *Lecture Notes in Computer Science*, 2006.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. *arXiv*, abs/1911.05722, 2019.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. *arXiv*, abs/2111.06377, 2021.

Hsu, W.-N., Tsai, Y.-H. H., Bolte, B., Salakhutdinov, R., and Mohamed, A. Hubert: How much can a bad teacher benefit ASR pre-training? In *Proc. of ICASSP*, 2021.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Deep networks with stochastic depth. *arXiv*, abs/1603.09382, 2016.

Jaegle, A., Borgeaud, S., Alayrac, J.-B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., Hénaff, O., Botvinick, M. M., Zisserman, A., Vinyals, O., and Carreira, J. Perceiver io: A general architecture for structured inputs & outputs. *arXiv*, abs/2107.14795, 2021a.

Jaegle, A., Gimeno, F., Brock, A., Zisserman, A., Vinyals, O., and Carreira, J. Perceiver: General perception with iterative attention. *arXiv*, abs/2103.03206, 2021b.

Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., and Liu, Q. Tinybert: Distilling bert for natural language understanding. *arXiv*, abs/1909.10351, 2020.

Jing, L., Vincent, P., LeCun, Y., and Tian, Y. Understanding dimensional collapse in contrastive self-supervised learning, 2021.

Kahn, J., Rivière, M., Zheng, W., Kharitonov, E., Xu, Q., Mazaré, P., Karadayi, J., Liptchinsky, V., Collobert, R., Fuegen, C., Likhomanenko, T., Synnaeve, G., Joulin, A., Mohamed, A., and Dupoux, E. Libri-light: A benchmark for asr with limited or no supervision. *arXiv*, abs/1912.07875, 2019.

Kahn, J. et al. Libri-light: A benchmark for asr with limited or no supervision. In *Proc. of ICASSP*, 2020.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, 2015.

Lample, G., Denoyer, L., and Ranzato, M. Unsupervised machine translation using monolingual corpora only. In *Proc. of ICLR*, 2018.

Likhomanenko, T., Xu, Q., Kahn, J., Synnaeve, G., and Collobert, R. slimipl: Language-model-free iterative pseudo-labeling. *arXiv*, abs/2010.11524, 2021.

Ling, S., Liu, Y., Salazar, J., and Kirchhoff, K. Deep contextualized acoustic representations for semi-supervised speech recognition. In *Proc. of ICASSP*, 2020.

Liu, A. T., Li, S.-W., and Lee, H.-y. Tera: Self-supervised learning of transformer encoder representation for speech. *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 2021.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Loshchilov, I. and Hutter, F. SGDR: stochastic gradient descent with restarts. *arXiv*, abs/1608.03983, 2016.

Manohar, V., Likhomanenko, T., Xu, Q., Hsu, W.-N., Collobert, R., Saraf, Y., Zweig, G., and Mohamed, A. Kaizen: Continuously improving teacher using exponential moving average for semi-supervised speech recognition, 2021.

McCann, B., Bradbury, J., Xiong, C., and Socher, R. Learned in translation: Contextualized word vectors. *arXiv*, abs/1708.00107, 2017.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. fairseq: A fast, extensible toolkit for sequence modeling. In *Proc. of NAACL System Demonstrations*, 2019.

Park, D. S., Zhang, Y., Jia, Y., Han, W., Chiu, C.-C., Li, B., Wu, Y., and Le, Q. V. Improved noisy student training for automatic speech recognition. *Proc. of Interspeech*, 2020.

Pasad, A., Chou, J.-C., and Livescu, K. Layer-wise analysis of a self-supervised speech representation model. *arXiv*, abs/2107.04734, 2021.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. Deep contextualized word representations. In *Proc. of ACL*, 2018.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf, 2018.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. *arXiv*, abs/2103.00020, 2021a.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. *arXiv*, abs/2103.00020, 2021b.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. Squad: 100, 000+ questions for machine comprehension of text. *arXiv*, abs/1606.05250, 2016.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. *arXiv*, abs/2102.12092, 2021.

Schneider, S., Baevski, A., Collobert, R., and Auli, M. wav2vec: Unsupervised pre-training for speech recognition. In *Proc. of Interspeech*, 2019.

Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. In *Proc. of ACL*, 2016.

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. FLAVA: A foundational language and vision alignment model. *arXiv*, abs/2112.04482, 2021.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. of EMNLP*, 2013.

Srivastava, S., Wang, Y., Tjandra, A., Kumar, A., Liu, C., Singh, K., and Saraf, Y. Conformer-based self-supervised learning for non-speech audio tasks. *arXiv*, abs/2110.07313, 2021.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, 2018.

Tay, Y., Tran, V. Q., Ruder, S., Gupta, J., Chung, H. W., Bahri, D., Qin, Z., Baumgartner, S., Yu, C., and Metzler, D. Charformer: Fast character transformers via gradient-based subword tokenization. *arXiv*, abs/2106.12672, 2021.

Tokozume, Y., Ushiku, Y., and Harada, T. Learning from between-class examples for deep sound recognition. *arXiv*, abs/1711.10282, 2017.

Tsimpoukelli, M., Menick, J., Cabi, S., Eslami, S. M. A., Vinyals, O., and Hill, F. Multimodal few-shot learning with frozen language models, 2021.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. S. Instance normalization: The missing ingredient for fast stylization. *arXiv*, abs/1607.08022, 2016.

van den Oord, A., Vinyals, O., et al. Neural discrete representation learning. In *Proc. of NeurIPS*, 2017.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *Proc. of NIPS*, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Proc. of NIPS*, 2017.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv*, abs/1804.07461, 2018a.

Wang, H., Ma, S., Dong, L., Huang, S., Zhang, D., and Wei, F. Deepnet: Scaling transformers to 1,000 layers. *arXiv*, abs/2203.00555, 2022.

Wang, W., Bao, H., Dong, L., and Wei, F. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *arXiv*, abs/2111.02358, 2021.

Wang, Y., Li, J., and Metze, F. A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling. *arXiv*, abs/1810.09050, 2018b.

Warstadt, A., Singh, A., and Bowman, S. Corpus of linguistic acceptability. https://nyu-mll.github.io/CoLA, 2018.

Wei, C., Fan, H., Xie, S., Wu, C.-Y., Yuille, A., and Feichtenhofer, C. Masked feature prediction for self-supervised visual pre-training. *arXiv*, abs/2112.09133, 2021.

Williams, A., Nangia, N., and Bowman, S. R. A broad-coverage challenge corpus for sentence understanding through inference. In *Proc. of NAACL*, 2018.

Wu, Z., Wang, S., Gu, J., Khabsa, M., Sun, F., and Ma, H. CLEAR: contrastive learning for sentence representation. *arXiv*, abs/2012.15466, 2020.

Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., and Hu, H. Simmim: A simple framework for masked image modeling. *arXiv*, abs/2111.09886, 2021.

Xu, Q., Likhomanenko, T., Kahn, J., Hannun, A., Synnaeve, G., and Collobert, R. Iterative pseudo-labeling for speech recognition. *Proc. of Interspeech*, 2020.

Xue, L., Barua, A., Constant, N., Al-Rfou, R., Narang, S., Kale, M., Roberts, A., and Raffel, C. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *arXiv*, abs/2105.13626, 2021.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv*, abs/1906.08237, 2019.

Zhang, Y., Qin, J., Park, D. S., Han, W., Chiu, C.-C., Pang, R., Le, Q. V., and Wu, Y. Pushing the limits of semi-supervised learning for automatic speech recognition. *Proc. of NeurIPS SAS Workshop*, 2020.

Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., and Kong, T. ibot: Image bert pre-training with online tokenizer, 2021.

Zhu, Y., Kiros, R., Zemel, R. S., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *arXiv*, abs/1506.06724, 2015.

A. Extended speech processing results

Table 6. Speech processing: word error rate on the Librispeech dev/test sets when training on the Libri-light low-resource labeled data setups of 10 min, 1 hour, 10 hours and the clean 100h subset of Librispeech. Models use the audio of Librispeech (LS-960) as unlabeled data.

Model	Unlabeled data	LM	dev		test	
			clean	other	clean	other
10 min labeled						
wav2vec 2.0 Base (Baevski et al., 2020b)	LS-960	4-gram	8.9	15.7	9.1	15.6
Hubert Base (Hsu et al., 2021)	LS-960	4-gram	9.1	15.0	9.7	15.3
data2vec Base	LS-960	4-gram	7.3	11.6	7.9	12.3
1h labeled						
wav2vec 2.0 Base (Baevski et al., 2020b)	LS-960	4-gram	5.0	10.8	5.5	11.3
Hubert Base (Hsu et al., 2021)	LS-960	4-gram	5.6	10.9	6.1	11.3
data2vec Base	LS-960	4-gram	4.0	8.5	4.6	9.1
10h labeled						
wav2vec 2.0 Base (Baevski et al., 2020b)	LS-960	4-gram	3.8	9.1	4.3	9.5
Hubert Base (Hsu et al., 2021)	LS-960	4-gram	3.9	9.0	4.3	9.4
data2vec Base	LS-960	4-gram	3.3	7.5	3.9	8.1
100h labeled						
Noisy student (Park et al., 2020)	LS-860	LSTM	3.9	8.8	4.2	8.6
IPL (Xu et al., 2020)	LL-60K	4-gram+Transf.	3.2	6.1	3.7	7.1
SlimIPL (Likhomanenko et al., 2021)	LS-860	4-gram+Transf.	2.2	4.6	2.7	5.2
wav2vec 2.0 Base (Baevski et al., 2020b)	LS-960	4-gram	2.7	7.9	3.4	8.0
Hubert Base (Hsu et al., 2021)	LS-960	4-gram	2.7	7.8	3.4	8.1
data2vec Base	LS-960	4-gram	2.2	6.4	2.8	6.8

B. Comparison of loss functions

Table 7 shows that different choices of the loss function have a relatively small effect on final performance.

Table 7. Different pre-training losses on Librispeech dev-other (no language model).

	WER
L2	17.1
L1	17.2
Smooth L1 ($\beta = 0.08$)	17.2
Smooth L1 ($\beta = 0.25$)	16.8
Smooth L1 ($\beta = 0.5$)	16.8
Smooth L1 ($\beta = 1$)	17.3

C. Speech masking parameter ablation

Our method requires different masking parameters for each modality and this makes intuitive sense: masking 15% of inputs is effective for text but not for images since text tokens are highly semantic and it is sufficient to mask a smaller proportion of the input to construct a useful task. We rely largely on settings from the literature, except for images, where we found that a higher masking rate compared to BEiT works slightly better. When we tried tuning masking hyperparameters for other

modalities than vision, we did not see significant improvements (e.g., see Table 8) for speech. The small improvements we saw are likely to disappear after adding a language model.

Table 8. Ablation of speech masking parameters. Results are on Librispeech dev-other without a language model.

	WER
baseline (mask prob = 0.65, mask len = 10)	17.1
mask prob = 0.8	17.2
mask prob = 0.5	17.3
mask len = 5	22.1
mask len = 15	18.9