Towards a Weakly-Supervised Learning Paradigm for Speech Recognition

Aissatou Ndoye[†]
African Institute for Mathematical Sciences
Rwanda
andoye@aimsammi.org

Sewade Ogun[†]
African Institute for Mathematical Sciences
Ghana
sogun@aimsammi.org

Yossi Adi[‡]
Facebook AI Research
Tel-Aviv, Israel
yossiadidrum@gmail.com

Moustapha Cisse[‡]
Google AI Research
Accra, Ghana
mcisse@aimsammi.org

Abstract

Current methods for learning acoustic representations for Automatic Speech Recognition (ASR) involve either supervised learning, where the labels are given as text or semi-supervised learning where labels are given as some form of audio feature to learn representation and then finetuned on a small amount of labelled data. These methods either require enormous computational time for training or data labelling time. Therefore, we explore a simpler approach that uses weak labels, as a proxy for the actual labels, obtained from ASR model trained on another language. This form of representation learning, therefore requires few labelled audios for the final finetuning task, if a good representation is learnt initially. We explore the idea in this research project and show how our experiments compare to current methods of speech recognition. Also, we show how this can be improved further through empirical studies and serves as a basis for a new line of supervision in the speech domain. Code for our experiments are available at: https://github.com/ogunlao/low_res_speech_project

1 Introduction

There has been major advances in recent times in self-supervised learning for images, text, and particularly, speech. The introduction of the seminal paper on Contrastive Learning [1], ushered in a wave of different self-supervised methods, giving performances which are on par or even better than supervised tasks.

However, these methods usually require a large amount of unlabelled data, and a ton of computation time to achieve good performance. In speech recognition, the current methods of self-supervision requires predicting the input or features using a form of contrastive learning or autoregressive prediction of features [2, 3, 4, 5]. In this project, we explore another form of supervision, a weakly-supervised learning paradigm where phonetic representations of a high resource language is distilled into a student model to learn a low resource language. In particular, noisy pseudo-labels from a pretrained automatic speech recognition model, usually in a high resource language, is used to learn

[†] Equal contribution; Paper submitted as a research project for the African Masters in Machine Intelligence Program

[‡] Equal supervision of research project

Preprint. Under review.

acoustic representations for a low resource language, thereby eliminating or reducing the use of large amount of labelled training data.

This project want to answer the question; Can we learn a new language faster if we have previously learnt another language? This setup is probably more meaningful for very similar languages but we focus on African Languages in this research project, as many African languages are significantly low-resourced.

2 Related Work

Contrastive Learning Self-supervised models such as CPC [2], Wave2Vec [3], Wav2Vec2 [4] have similar setups where the models are trained to contrast between the speech features at different time steps, which they can then adapt their learned representations to perform downstream tasks like phoneme classification and speaker classification [6], and automatic speech recognition (ASR) [3, 4]. These acoustic features learnt have also been shown to transfer well to other languages, as in [6]. The commonly used features for supervision are mel-spectrogram, mel-frequency spectral coefficients (MFCCs) and raw audio.

Autoregressive Learning Other similar tasks like Time Contrastive Learning [7] and Autoregressive Predictive Coding (APC) [5] have been used to learn acoustic representations for ASR where future time-steps are predicted from current ones. It was also very successful in autoregressive generation of audio sounds [8, 9]

Multimodal Learning Representations can be learnt by combining actions or information from different modalities such as speech, text, image or video. This paradigm is common in the text/image domain. In speech literature for instance, Rahma et al.[10] combined audio and visual information of lips movements in a weakly supervised way using Siamese networks and lexical same-different side information to learn acoustic features.

3 Weak Supervision

Consider a set of N i.i.d observations $D_1 = \{(x_i, y_i)\}_{i=1}^{N}$ defined over $X \times Y$ domains, where in this case, X is a set of speech and Y is a set of weak-labels. Similarly, there is a set of M observations from $D_2 = \{(x_j, y_j)\}_{i=1}^{M}$ still defined over $X \times Z$ domains, where $x_j s$ are still samples from X, and $y_j s$ are labels, for the final task. In practice, samples from D_1 are more abundant in nature and helps to augment the few samples we have from D_2, hence, the setup is to transfer information as much as possible from the first task to the second.

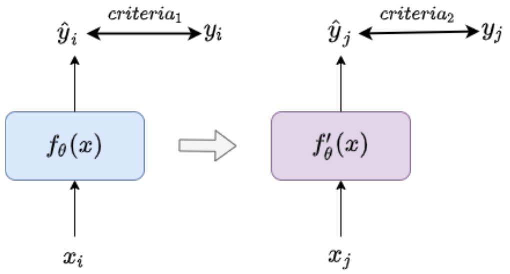

Figure 1: Diagram showing how weak supervision can be applied to learn good representations. The first task is to train a model on weak labels using a good criteria that puts into consideration the limitations of the labels and then finally finetune this model on the final task.

Here, a parameterized function $f_\theta(x)$ is first learnt by converting the weak-label classification problem to a supervised learning task as shown in Figure 1. In this task, predicted weak-labels $\hat{y}_i$ which are gotten through, $\hat{y}_i = f_\theta(x_i)$ are compared with the ground-truth weak labels y_i by computing a learning criteria, while the model continues to refine the parameterized function through an optimization process such as gradient descent. The criteria depends on diffent properties of the setup such as input-output type, alignment between input and output, nature of labels etc.

Afterwards, the parameterized function $f_\theta(x)$ is then refined using observed samples from $D_2 = \{(x_j, y_j)\}$ for a small number of iterations to arrive at $f_\theta^{'}(x)$ for the final task, assuming the representations learnt at the initial training stage is good. The output at this stage correspond to the true labels, and the criteria for model performance also depends on the the final task. The two tasks can also be jointly performed through a multi-task learning approach to ensure that this representation satisfy the two conditions. Common criteria used for learning include Cross entropy loss, Connectionist temporal classification criterion (CTC) [11] and Auto-segmentation criterion (ASG) [12].

Weak labels can be words, subwords, characters, phonemes, phones, images, speech or text which have a direct or indirect relation to the input signal. For our task, the weak labels are phonemes and labels are characters/text which is to be used for Automatic Speech Recognition. Both of our weak labels and labels in this task do not have direct alignment with the input speech, therefore we are limited in the kind of criteria to use in learning. The following subsection explains our setup for the weak supervision task.

3.1 Training Pipeline

We propose to explore the fact that phones in many languages sound similar [13]. This is even more factual with languages from the same family like English and French. For instance, more than 45% of words in English are borrowed words from the French language [14]. This implies that an acoustic model which has learnt french should transfer well to English and vice-versa. Our preliminary studies also showed this pattern.

Let's denote the high resource language as LangH and low-resourced language as LangL. Let us also represent the model pretrained on high resource language as ModelH and the model to be distilled with information from LangH as ModelL. Our setup is then as follows;

1. **Generate predictions:** Generate predictions for unlabelled speech in LangL using ModelH. These predictions, termed pseudo-labels clearly look like text in LangH but when read aloud will sound similar to the audio in LangL. We select the sequence of characters which maximize the posterior probability for the input feature as pseudo-labels. We call this max-decoding
2. **Phonemize:** Break down pseudo-labels to more informative units like phonemes or bigrams. In our experiments, we convert the pseudo-labels to phonemes in LangH, as we do not know of any direct phoneme conversion between the languages, and moreso, phonemizers are not currently available in our target African languages
3. **Distill ModelL:** A new model, ModelL, is trained using the unlabelled speech audio from LangL and their corresponding weak labels, phonemes in this case
4. **Finetune ModelL:** The model is finetuned on a small amount of labelled data in LangL. 5 hrs or 10 hrs of labelled data was used in our experiments. Also, the model encoder can be frozen while only the decoder is finetuned however we do not freeze the encoder in our setup.

Examples of pseudolabels generated using an English pretrained model for French and Kinyarwanda languages are shown in the following paragraphs. Figure 2 shows the training setup used in our experiments.

French
Label: *L'endroit est recouvert de goyaviers et d'acacias non endémiques*
Pseudolabel: *along the warld ihokuria deguya yi ite kasia nonamu*

Label: *Cette espèce est nommée en l'honneur d'Erich Titschack*
Pseudolabel: *setespice is ne mio muel derich techarch*

Label: *Il est le fils cadet de Teimouraz de Kakhétie et de Khorassan-Daredjan de Karthli*
Pseudolabel: *lil fhi cedied til wast catesi it kolasan daijanika*

Label: *Il a participé à la guerre de succession d'Espagne*
Pseudolabel: *ela participerior the other success of his panun*

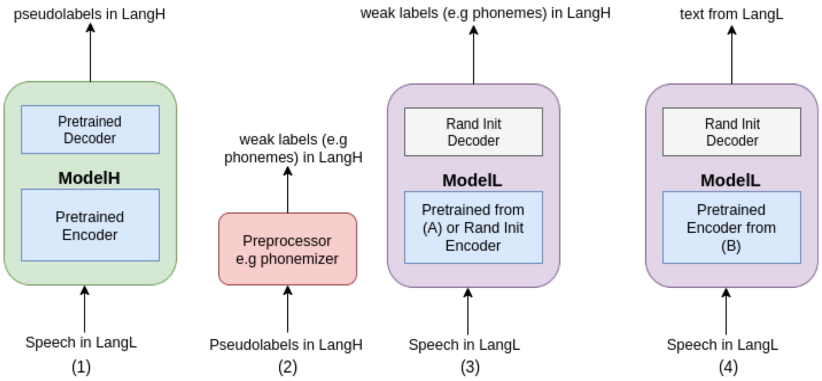

Figure 2: Training setup for weak supervised learning task. (1) Generate pseudolabels from ModelH using LangL speech audio. (2) Phonemize the pseudolabels. (3) Train a new model (initialized with ModelH encoder or randomly initialized) on phonemes. (4) Finally, finetune ModelL on small amount of LangL speech audio with text

Kinyarwanda
Label: *ati birashimishije nta muntu utatera inkunga iki gikorwa kandi kubufatanye bwacu twembi tuzabigeraho*
Pseudolabel: *airerashuishanamatatere o iuikorakanikufatee gauweuuzaera*

Label: *ibyo bikoresho ngo bikoresha ingufu nke ubikoresha arabyikanikira iyo byangiritse biraramba kandi byoroshye kubitwara*
Pseudolabel: *bikoreshoa ikoreshuguiyku kareshararaikanicayo anisriraaaaniie wa*

Label: *kwitabira ibikorwa bya leta ni inshingano za buri mutura rwanda*
Pseudolabel: *kwtaireko kaukwayare tanishn a zarima*

3.2 Model Architecture

The models comprise of an encoder layer and a decoder layer. The encoder encodes the speech into a latent vector, which is then converted to labels by the decoder. We adopt a Jasper10x5 model architecture [15], containing 10 blocks of convolutional layers and residual connections, with each block having 5 repeating sub-blocks. Each sub-block applies the following operations in sequence: 1D-Convolution, Batch Normalization, ReLU activation, and Dropout. The structure of the residual blocks and sub-blocks is shown in Figure 3.

Jasper is a computationally efficient end-to-end convolutional neural network acoustic model, which achieved competitive results on Librispeech dataset [16]. It takes spectrogram features as input and produces a sequence of character outputs. Jasper10x5 contains 333 million parameters. The same model architecture was used for ModelH and ModelL in our experiments, however a different model architectures can be used as ModelL. The experiments were setup using the NeMO conversational AI library[1] from Nvidia.

3.3 Loss/Criteria

The Connectionist Temporal Classification (CTC) criteria [11] was used in both the phoneme classification and fine-tuning experiments. This criteria maximises the joint probability between all possible paths to a particular sequence, and has been explored in many sequence-to-sequence tasks where the input and target require no explicit alignments [17, 18, 19].

In the pretraining stage, the model maximises the probability of getting a sequence of weak labels, given the input,

$$Y = \arg\max_{y} P(y_1, y_2, ..., y_n | x_1, x_2, ..., x_m; \theta)$$

[1] https://github.com/NVIDIA/NeMo

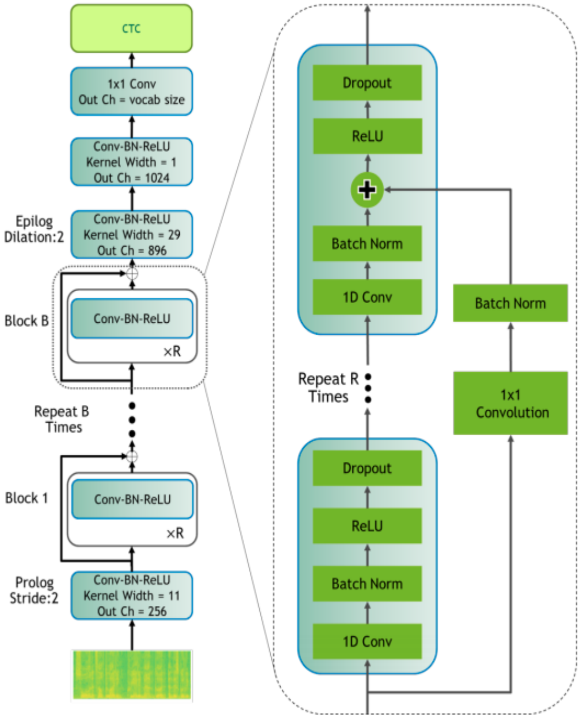

Figure 3: Jasper BxR model: B - number of blocks, R - number of sub-blocks. [15]

where $n \neq m$. Afterwards, we finetune the model, ModelL on labels. All of our experiments were performed using max-decoding without language model or beam search.

4 Training Setup

All audio files were converted to wav, mono-channel, 16bit format with a sampling rate of 16kHz. This format ensures compatibility with NeMo and used extensively in most ASR applications.

4.1 Pretraining

Data size 100 hours of unlabelled speech in two African languages, Kinyarwanda and Kabyle was used in our experiments. It should also be possible to very small amount of unlabelled data in this setup. More details about the datasets are given in section 5.

Generating Pseudo-labels Audio samples are processed and passed through ModelH, thereby getting predictions at the output as sequence of characters or pseudo-labels in LangH. Due to a lack of language model, the length of some pseudo-labels appear shorter than the speech sample text or even empty. So, these samples are filtered out and eliminated if their predictions are empty or has only one character. Another approach could be to estimate the length of the label given the audio, then use this information to filter out improbable pseudo-labels.

Converting Pseudo-labels to Phonemes We phonemize the sequence of characters using the bootphon phonemizer[2] with festival backend. Festival[20] uses a custom American English phoneme set and allows tokenization at the syllable level and word level. The custom phoneme set, comprising of 41 phonemes, can be found in the link[3].

[2] `https://github.com/bootphon/phonemizer`

[3] `http://www.festvox.org/bsv/c4711.html`

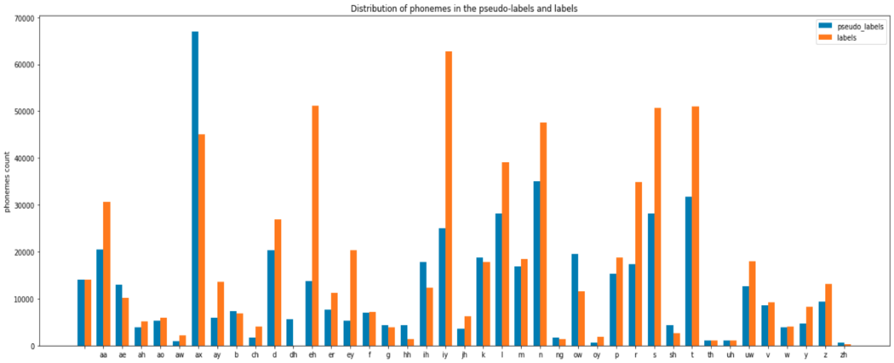

Figure 4: Distribution of phonemes in pseudolabels and labels in Kinyarwanda using the bootphon phonemizer

Phonemes with multiple characters are mapped to single ASCII characters for the training task and a silence token is also included between each word generated in the pseudo-label. A sequence of characters are taken as a word if there is an empty character token between the sequence and the next sequence. An approximate distribution of phonemes generated to phonemes of actual labels is shown in Figure 4.

The phoneme distribution follows the distribution of the original labels closely.

Evaluation The pretraining task was evaluated on a validation set using the Phone Error Rate (PER).

4.2 Finetuning

Data size Finetuning experiments were carried out with 5 and 10 hours of labelled speech in LangL. The smaller dataset is made to be a subset of the larger dataset.

Preprocessing All sentences are cleaned by removing all standard punctuation like full stop, commas, etc. except the apostrophe. For Kabyle, we also include the hyphen punctuation as it was a common character in the dataset and may contain extra information about word composition. Accents on characters were as well removed in order to have a substantial frequency count for the characters without impeding readability of sentences.

Evaluation The finetuned model, ModelL is evaluated on a validation and test set using Character Error Rate (CER).

5 Experimental Evaluation

Model A pretrained Jasper10x5 model[4] was the selected model for pseudolabel generation. This ASR model was trained on 7 different datasets of English speech, with a total of 7,057 hours of audio samples. For our experiments, we either train on the pseudolabels, initializing with weights of this pretrained model or train with random model initialization. We also randomly initialize the decoder during all finetuning experiments, irrespective of the encoder weights, since the length of vocabulary in the pretrained model will not match the length of vocabulary for the target language.

Datasets Experiments were performed with two African Languages, Kinyarwanda (KW) and Kabyle (KB). Kinyarwanda is a language of Niger-congo family predominantly spoken in Rwanda, Uganda, DR Congo, Tanzania. It uses Latin characters. Kabyle is a language of Afro-Asiatic family, spoken majorly in the northern part of Algeria. All datasets were sourced from the Common Voice Corpus 6.1 [21].

[4] `https://ngc.nvidia.com/catalog/models/nvidia:nemo:stt_en_jasper10x5dr`

To segment the audio samples, the samples were converted from their original mp3 format to wav format, then random samples of speech totalling 100 hours duration was selected for pretraining while other disjoint random samples of 10 hours and 5 hours were selected for finetuning, from the same training set. We use the predefined development set and test set for validation and testing respectively. KW dataset consist of 24 hours of development and test sets while KB dataset comprise of 14 hours of development and test sets respectively.

Hyperparameters A similar hyperparameter configuration was used for all experiments involving Jasper (except where stated otherwise). Stochastic Gradient Descent (SGD) with momentum and Cosine Annealing scheduler with linear warmup ratio of 1e-1 was used for learning. A base learning rate of 1e-4 for pseudolabel training and 1e-4 for finetuning ModelL was set for the experiments. The best model found during training based on the validation CER is saved and evaluated on the test set. Also, we setup our experiments on 4 or 8 Nvidia V100 GPUs depending on data size. The pseudolabel training took 3 days, while the finetuning took a day on average on 4 gpus.

For data augmentation, SpecAugment [22] and CutOut, [23] with time mask of 120, frequency mask of 50 and rectangular mask of 5, are applied on the spectrogram during finetuning. For pretraining, models without data augmentation were also considered, since the labels are noisy and further regularization via data augmentation may not be necessary. In our experiments, models with data augmentation gave better results. All other configurations are similar to the configurations used to train Jasper10x5 on the English ASR task.

5.1 Results

Firstly, we train some baselines to determine how our approach compares to other direct ASR approaches. The two baselines considered are in section 5.1.1

5.1.1 Baselines

a. Train a randomly initialized model on 5 hours and 10 hours of LangL

b. Finetune Jasper initialized with weights from ModelH on 5 hours and 10 hours of LangL

As seen in Table 2, finetuning a pretrained model (setup b) performs significantly better than training a randomly initialized model. This is a well-known phenomenon in deep learning, which is suitable for transfer learning. Afterwards, subsequent experiments explored phoneme classification pretraining to determine the influence of learned representations from phonemes on the final ASR task.

5.1.2 Our method

Pretrain without Data Augmentation These set of experiments do not use any form of augmentation such as time masking, frequency masking or cutout on the audio features. ModelL encoder is either initialized from a pretrained English model i.e ModelH encoder or train ModelL on phonemes directly. It is worth mentioning that these experiments follow those described in section 5.1.1 and the results of the phoneme classification are shown in Table 1. In the tables, the setup tags correspond to the experiment numbers with 1 indicating phoneme classification task and 2 indicating finetuning task.

c. Randomly initialize model, train on phonemes, then finetune on 10 hours and 5 hours of labelled data

d. Initialize with Jasper-Eng ASR, train on phonemes, then finetune on 10 hours and 5 hours of labelled data

   The best method between the two setups above are selected for further studies. Also, since the labels are noisy, the validation set may not be a true generalization criteria for our approach, therefore this take had to be explored in subsequent experiments.

e. Randomly initialize model, train on phonemes, then finetune on 10 hours and 5 hours of labelled data, returning the model with best train loss. stopping when the train loss stagnates.

f. Randomly initialize model, train a phoneme classifier with augmentation until train loss degrades, then finetune model on 10 hours and 5 hours of labelled data

Table 1: PER of train and validation sets for phoneme classification of 100 hours of Kinyarwanda and Kabyle. Lower PER is better.

	KW		KB	
setup	train PER	val PER	train PER	val PER
c-1	0.4276	0.5483	-	-
d-1	0.4924	0.5528	-	-
e-1	**0.2098**	-	-	-
f-1	0.2554	-	0.4629	0.5153

Table 2: CER of ModelL finetuned on 10 hours and 5 hours of Kinyarwanda labelled data. Lower CER is better.

	10 hrs KW		5hrs KW		10 hrs KB		5hrs KB	
setup	val CER	test CER	val CER	test CER	val CER	test CER	val CER	test CER
a	0.3489	0.3813	0.4398	0.4640	0.3177	0.3364	0.4115	0.4262
b	**0.2124**	**0.2507**	**0.2396**	**0.2747**	**0.2089**	**0.2112**	**0.2179**	**0.2284**
c-2	0.3306	0.3665	0.3828	0.4116	-	-	-	-
d-2	0.3409	0.3743	0.3924	0.4225	-	-	-	-
e-2	0.3273	0.3621	0.3811	0.4124	-	-	-	-
f-2	**0.3120**	**0.3484**	**0.3725**	**0.4055**	0.2936	0.3112	0.3497	0.3667

An empirically observation is that there is an improvement in our results when the model trains longer on the training set, instead of early stopping where the loss on the validation set was no longer improving. In reality, the model could not essentially overfit the training data, as it is very noisy, and this is indicated by the high train PER.

Pretrain with Data Augmentation We further include data augmentation in the pretraining loop to evaluate its effect on the task. The PER was slightly higher than the experiment without augmentation, but it still performed better on the final finetuning task than previous approaches. This may be because the finetuning task also used the same augmentation strategy as used during phoneme classification, without any augmentation mismatch. Table 1, setup (f-1) shows the result for this experiment.

5.2 Interpreting Results after Finetuning

For our task, we finetuned all pretrained models on 5 hours and 10 hours of labelled speech. Table 2 shows the results. Under our approach, we got the best finetuning results when we pretrained with augmentation and allow the training task to fit the training data, regardless of the validation CER. Note that for Kabyle, we only run experiments for the baseline and the best experiment results found during the Kinyarwanda task.

All our approach performed better than training a randomly initialized model from scratch, however we perform far worse than an English pretrained model, directly finetuned with labelled speech. This might be indicative of a breakdown in acoustic features during our phoneme classification, or phoneme-character mismatch for the final task. We would have expected to performed at least as good as a pretrained model, or even better. Also, we discovered that it is better to pretrain a randomly initialized model for our task than initializing with a pretrained model for the phoneme classification task.

Finally, finding the most appropriate learning rate for this task proved difficult as we initially got worse results for most of the experiments with high learning rates. Our approach even performed significantly better than the English pretrained model at high learning rates.

6 Discussions, Limitations and Future Work

6.1 Discussions and Limitations

In this project, a weakly supervised learning framework for learning representations was explored particularly for African languages, however the results showed that more work need to be done in this line of research to be fully competitive with current approaches. It was discovered through empirical evaluations that the pseudo-labels were very noisy, thereby degrading the performance of the acoustic model. We believe with further experiments, and probing, we can perform knowledge distillation using weak labels, significantly reducing the burden of labelling audio for future ASR tasks.

6.2 Future Work

For future work, we intend to explore other criteria for the pseudo-label training. In particular, we will explore loss functions that can handle noisy labels optimally. Another idea is to convert weak labels to other informative forms for training. We can also generate pseudo-labels conditioned on the probability distribution of predictions of a language model either in the source or target language, to help improve the pseudo-label prediction. We hope that these modifications can help improve on the results presented in this paper.

7 Acknowledgements

We acknowledge Fadel Thior and Ahmed Hassan for their help in setting up some of our experiments. Also, initial discussions with Sai Krishna provided useful insights into how to proceed in this line of research.

References

[1] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, 2017.

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

[3] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*, 2019.

[4] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*, 2020.

[5] Yu-An Chung, Hao Tang, and James Glass. Vector-quantized autoregressive predictive coding. *arXiv preprint arXiv:2005.08392*, 2020.

[6] Morgane Rivière, Armand Joulin, Pierre-Emmanuel Mazaré, and Emmanuel Dupoux. Unsupervised pretraining transfers well across languages. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7414–7418. IEEE, 2020.

[7] Achintya Kumar Sarkar, Zheng-Hua Tan, Hao Tang, Suwon Shon, and James Glass. Time-contrastive learning based deep bottleneck features for text-dependent speaker verification. *Ieee/acm Transactions on Audio, Speech, and Language Processing*, 27(8):1267–1279, 2019.

[8] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[9] Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron Oord, Sander Dieleman, and Koray Kavukcuoglu. Efficient neural audio synthesis. In *International Conference on Machine Learning*, pages 2410–2419. PMLR, 2018.

[10] Rahma Chaabouni, Ewan Dunbar, Neil Zeghidour, and Emmanuel Dupoux. Learning weakly supervised multimodal phoneme embeddings. *arXiv preprint arXiv:1704.06913*, 2017.

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

[12] Ronan Collobert, Christian Puhrsch, and Gabriel Synnaeve. Wav2letter: an end-to-end convnet-based speech recognition system. *arXiv preprint arXiv:1609.03193*, 2016.

[13] Damián E. Blasi, Søren Wichmann, Harald Hammarström, Peter F. Stadler, and Morten H. Christiansen. Sound–meaning association biases evidenced across thousands of languages. *Proceedings of the National Academy of Sciences*, 113(39):10818–10823, 2016.

[14] Wikipedia contributors. List of english words of french origin — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=List_of_English_words_of_French_origin&oldid=1019837089`, 2021. [Online; accessed 22-June-2021].

[15] Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan M Cohen, Huyen Nguyen, and Ravi Teja Gadde. Jasper: An end-to-end convolutional neural acoustic model. *arXiv preprint arXiv:1904.03288*, 2019.

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

[17] Alex Graves, Marcus Liwicki, Santiago Fernández, Roman Bertolami, Horst Bunke, and Jürgen Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):855–868, 2008.

[18] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pages 1764–1772. PMLR, 2014.

[19] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[20] Paul Taylor, Alan W Black, and Richard Caley. The architecture of the festival speech synthesis system. In *The Third ESCA/COCOSDA Workshop (ETRW) on Speech Synthesis*, 1998.

[21] Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*, 2019.

[22] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

[23] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.