INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

The MALACH Corpus: Results with End-to-End Architectures and Pretraining

Michael Picheny[1],*Qin Yang*[2],*Daiheng Zhang*[2],*Lining Zhang*[2]

[1]NYU Courant-Computer Science and Center for Data Science, USA
[2]NYU Center for Data Science, USA

map22@nyu.edu,qy692@nyu.edu,dz2266@nyu.edu,lz2332@nyu.edu

Abstract

The MALACH corpus contains approximately 375 hours of Holocaust survivor testimonies along with transcripts (for approximately half the data) and audio. It is an extremely difficult corpus for speech recognition, encompassing accented, emotional speech, in many cases from elderly survivors. Nevertheless, significant progress has been made on speech recognition on MALACH with WERs now typically hovering at a 20% level for hybrid speech recognition systems. The purpose of this paper is to examine if recent developments in end-to-end architectures and pretraining with self-supervision continue to drive down performance as they do on popular read corpora such as Librispeech. We also experiment with leveraging the large fraction of unlabeled corpus data by extracting pseudolabels produced from previously trained systems. It is found that the best system - a fine-tuned wav2vec2 system trained on labeled and pseudolabeled data - achieves a WER of 13.5%, a huge gain in performance.

Index Terms: speech recognition, human-computer interaction, computational paralinguistics

1. Introduction

The USC-SFI MALACH Interviews and Transcripts English corpus [1] is a 375-Hour subset of a very large archive of Holocaust testimonies collected by the Survivors of the Shoah Visual History Foundation [2]. It contains accented and emotional speech, in many cases from elderly survivors, and contains a large number of non-English named entities, such as names and places. Recognition of this data was originally thought to be an impossible task, but by 2019, the word error rate (WER) had been driven down to approximately 21.7% using LSTM hybrid models [3].

The importance of this corpus is that it is not an artificial task; it contains real, difficult audio data of deep historical interest that would not be accessible without expensive manual transcription. The goal is to obtain good enough WERs to enable downstream NLP tasks important for understanding, such as Entity Detection, Document Segmentation, and Segment Categorization [4]. It is unclear if a WER of 21.7% is adequate. Early work on MALACH [4] suggested that different NLP tasks require different thresholds of WERs to reach usable levels of performance, but the study, while comprehensive, was done on a much older technology base both for recognition and NLP, and nothing like it has been done since. Anecdotal evidence reported in papers on closed captioning [5, 6] suggest that WERs below 10% are required for readability. It is therefore important to continue to track if recent techniques in speech recognition can further drive down the WER on MALACH. This paper examines WER impact with such recent techniques as transformer-based end-to-end architectures (e.g., [7, 8]) and leveraging large pretrained models based on massive amounts of unlabelled data (e.g., [9, 10]), either by leveraging such models directly, or by fine-tuning them. Most published results have tended to be on artificial tasks such as Librispeech [11], or private internal product-targeted databases. It cannot be blindly assumed that equivalent performance improvements will accrue for data in the ”wild” such as MALACH without verification.

A secondary issue to be addressed is that only part of the original MALACH speech corpus was released in a form suitable for speech recognition. Some fully transcribed interviews were omitted, and a large fraction of speech was only partially transcribed. This paper also describes attempts to process and leverage the additional training and test data suitable for speech recognition purposes. A set of experiments demonstrating the value of incorporating such previously unlabelled data to achieve additional improvements on this still-difficult corpus is also described and confirmed on both old and new MALACH test data.

The rest of the paper is broken up as follows. Section 2 describes previous and related work. Section 3 explains work done to recover additional data from the original corpus release, Section 4 describes the various recognition systems tested and the results from each system, Section 5 describes experiments on pseudo-labels derived from the unlabeled data, and Section 6 discusses the results along with suggesting future work.

2. Previous and Related Work

In 2019 a version of the MALACH English corpus was released in a form suitable for experiments in speech recognition. [12]. The basic training data set includes 176 hours of manually transcribed speech, along with a 1.5 hour ”minitest” that went through multiple correction passes to correct transcription errors. Note that the first release of the training data contained many utterances with blank transcriptions; when eliminated, 153 hours of audio containing speech was recovered. The best result, a WER of 21.7% [3], was obtained using a hybrid LSTM acoustic model rescored with a LSTM-based LM.

Since then, there have been numerous improvements made to general speech recognition performance. No attempt will be made to provide a comprehensive review; this paper will focus on reasonably current architectures embodied in various accessible open-source systems.

ESPNET [13, 14] is a toolkit that provides implementations of current architectures such as transformer and conformer attention-based encoder-decoder systems, and a conformer-based RNN-T. Huggingface [15] provides a large set of finetunable pretrained models. One of the more popular methodologies is wav2vec2[16] in which a representation is pretrained

10.21437/Interspeech.2023-1014

on a large quantity of unlabelled data and then fine-tuned to produce very good results on a new corpus. There has been recent work applying wav2vec2 methodologies to a MALACH Czech corpus with promising results[17]. Whisper [10] is a large conformer-based system trained on an enormous quantity of semi-supervised data. All of these systems provide strong performance when trained on a large corpus; unfortunately it is not easy to compare results across systems because of differences in scoring methodologies, amount and types of training data, and the types of data augmentation applied. One interesting set of comparisons have been made across these architectures on a set of 8 public domain speech recognition tasks [18]. The systems with the best performance seemed to be Whisper and a Conformer-based RNN-T.

In terms of dealing with unlabeled data and significant numbers of named entities with many OOVs, early work on MALACH [19] demonstrated a roughly 20% Word Error Rate Reduction (WERR) using a word-fragment-based system along with a posterior-based confidence selection scheme involving 600 hours of unsupervised MALACH data (unfortunately not made available publicly). There have been many studies generally demonstrating the value of unlabelled data in speech recognition. A good recent article containing multiple references on the use of unlabelled data by creating artificially labeled data ("pseudo-labels", also known as "self-training", or "student-teacher training") is given by [20]. The technique will be employed in Section 5 to demonstrate additional speech recognition improvements on MALACH.

3. New Data Extraction

As mentioned above, the original speech recognition MALACH release comprised of approximately 155 hours (153 hours of transcribed speech and 1.5 hours of minitest data) of data created during the original NSF MALACH program. The original MALACH speech corpus release contains significantly more speech data, both transcribed and unlabelled. To increase the value of the resource to the community, some effort was made to extract and process the remaining data to increase the amount of training and testing data.

The original MALACH data was recorded on a set of digital audio tapes. Each tape held approximately 30 minutes of interview data. The original LDC data distribution consisted of a set of MP2 and XML files. Each MP2 file was paired with an XML file and corresponded to a digital tape (i.e., a segment of the interview). The interviewee and interviewer were captured on separate audio channels using two different microphones, but there was often considerable cross-talk across the channels because of proximity. In the original processing of the data for speech recognition, the interviewee and interviewer channels were manually chosen. In this new round of data extraction, we merely summed the channels for convenience.

The XML files contained the transcriptions. Speaker turns were manually annotated, as well as long pauses inside a speaker turn. For the additional data, the long pauses were used to segment each speaker turn into smaller segments to make it easier to process the data to build speech recognition systems. Speaker overlap was also annotated; any segment containing speaker overlap was discarded. The large number of unfamiliar foreign words and named entities presented challenges for the transcribers. All such entities were prefixed with an '@'. Sometimes they were correctly transcribed; sometimes not, but for ease of processing the '@' was just stripped off and whatever spelling was provided was accepted. The net effect was to produce another 18 hours of transcribed data for training, 3 hours of new transcribed test data, and 150 hours of unlabelled audio data. The unlabeled data was then processed by the Kaldi Voice Activity Detection (VAD) component[1] to segment the data into usable training length utterances for speech recognition. This component uses a simple energy based VAD algorithm and produces a set of speech segments. Note that the resultant segments tend to be much longer than the manually segmented data in the MALACH speech recognition release (average of 3 sec. vs. 9 sec.)

4. Recognition Systems

An End-to-End ASR model trained from scratch on the relatively small MALACH corpus may not be able to learn a representation that produces improved performance relative to existing results. Therefore, we tried multiple approaches. First, we built a number of standalone speech recognition systems based on transformer and transducer architectures using only MALACH data. Second, we examined recognition performance on the recently released Whisper system, trained on massive amounts of semi-supervised data. Last, we fine-tuned a wav2vec2 system which leverages a large pre-existing corpus of unlabelled data. We tried fine tuning on both the base MALACH data and also by augmenting this data with pseudo-labeled data from the base MALACH data. Again, to be clear, we opportunistically employed open-source systems to give us some sense about the relative merits and ease of use of the various systems.

4.1. Scoring

The NIST SCTK toolkit[2] was used for scoring the WER across all systems. To be consistent with prior MALACH work[3], a GLM (global mapping rule) file was utilized. This GLM file mapped numbers into spellings, expanded certain abbreviations into a spoken form, did not score filled pauses, and homogenized alternative spellings. To handle the outputs of E2E systems, the GLM file had to be expanded from its earlier release as part of the MALACH corpus[12]; the new version will be released along with the planned release of the additional metadata (it did not affect earlier results that were scored with the original GLM).

4.2. Standalone End-to-End Architectures

ESPNET [14] is a speech processing toolkit that implements a large number of architectures for various speech processing tasks. In particular, ESPNET provides implementations and training and evaluation recipes for three currently popular end-to-end speech recognition architectures: Attention-based Encoder-Decoder Transformer (T-AED), AED conformer (C-AED), and a conformer-based RNN-T (RNN-T). Each training recipe[3] was used as provided including the system architecture and hyperparameters. Each architecture was trained on the original MALACH training data; 4 hours of the training data were held out for validation, and the minitest data was used for evaluation. The training recipes all employed the Adam optimizer and the numel batch type. Table 1 contains the results along with the provided hyperparameters.

When compared to previous results (21.7%[3]) some relative improvements were obtained for all systems; the conformer-based RNN-T seems to have somewhat worse per-

[1] `https://github.com/kaldi-asr/kaldi/blob/`

Table 1: *WERs and Hyperparameters for different ESPNET Architectures.*

Model Arch	WER(%)	learning rate	epochs	batch bins	warm up
T-AED	19.9	.0030	75	14M	25K
C-AED	19.6	.0025	50	35M	40K
RNN-T	21.4	.0015	100	20M	25K

formance than the AED, at least for the configurations used.

4.3. Whisper

The Whisper system (Section 2) reports very strong performance across a variety of tasks; it therefore serves as another useful comparison point for MALACH. The Whisper system was used to transcribe the minitest data. The main problem that was encountered was that the output text from Whisper is normalized, whereas the reference text used to determine WER was not. Code had to be written to reverse the text normalization produced by Whisper. In the case of MALACH the main issues dealt with numbers and abbreviations. The model used was the Whisper "medium" model. After the de-normalization, the resultant WER from Whisper was 18.0%. Note that no fine tuning was attempted for the Whisper model, so this is a very impressive result.

4.4. Wav2vec2 Systems

All systems were fine tuned using as a base model the "wav2vec2-large-960h-lv60-self" model, which is the wav2vec2 model trained on 960 hours of Librispeech and 60K hours of Librilight data.

4.4.1. Hyperparameter Adjustment

As mentioned in Section 2, wav2vec2 produces representations that can be easily fine-tuned on new data. Wav2vec2 is trained by predicting speech units for masked parts of the audio. In the huggingface implementation that supports fine-tuning, the wav2vec representations can be fine-tuned using Connectionist Temporal Classification (CTC).

Preliminary experiments revealed that the fine-tuning process required some adjustment of the original training hyperparameters to ensure that training consistently converged to an optimum associated with a good word error rate, especially as the amount of fine-tuning data was increased. After a number of fruitless attempts attempts to tune the hyperparameters via grid search, an entry in a support forum[21] was found with a much better set of hyperparameters. It was also found that running considerably fewer epochs was both faster and did not seem to compromise results. Default and best values can be found in Table 2.

4.4.2. Fine-Tuning Results

We varied the amount of training data from 15.3 hours to 153.1 hours. It was also found that with the standard per user computational setups in our cluster relatively easily available (up to 400 GB of memory; 4 RTX6000 or VT100 GPUS; 48 core Lenovo 670 CPU), we encountered various memory issues for the larger configurations, sometimes due to GPU limitations and sometimes due to the size of the training data. These were resolved by increasing the amount of memory and also by running on 3 GPUs while reducing the batch size from 32 to 10 for each GPU; such configurations typically took 2-3 days to train.

Table 2: *Default and Improved Fine-Tuning Hyper-parameters*

Description	Default	Improved
learning rate	1e-4	4e-5
weight decay	5e-3	3e-2
warmup	1000	500
epochs	30	3

Table 3 contains the results with varying amounts of data. Data was selected randomly from the training set of 153.1 hours.

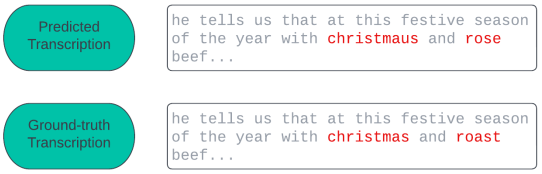
Table 3: *WER after fine-tuning wav2vec2 model vs. amount of speech*

Data	WER
15.3 hours	18.2%
38.3 hours	17.0%
76.5 hours	16.7%
153.1 hours	15.9%

Note that the 15.9% WER in particular is quite a good result for MALACH.

4.5. Results using a Language Model

The above configuration leverages the CTC algorithm for finetuning[16], which does not require an external language model or dictionary to yield acceptable audio transcriptions. However, as we can see from Figure 1, the predicted transcription can sound correct, but can often be spelled incorrectly. For example, the "christmaus" vs. "christmas" and "rose beef" vs. "roast beef". In a traditional word-based speech recognition system, there is no concept of spelling errors, and common phrases such as "roast beef" are rarely mistaken even with simple bigram language models. In the E2E literature, there are two approaches to mitigating the problem. For example, the language model can be learned together with the acoustic model. In the RNN-T Conformer (Section 4.2) an RNN Transducer architecture is used where a language model is learned end-to-end along with a powerful acoustic model. NVIDIA has recently released an RNN-T model trained on 24K hours of publicly available



he tells us that at this festive season of the year with christmaus and rose beef...

he tells us that at this festive season of the year with christmas and roast beef...

Figure 1: *Predicted vs. Ground-truth Transcriptions Using Fine-tuned wav2vec2*

master/egs/wsj/s5/steps/compute_vad_decision.sh

[2] https://github.com/usnistgov/SCTK, version 2.4.12

[3] https://github.com/espnet/espnet/tree/master/egs2/ami/asr1

speech[4] with strong performance that can be fine-tuned, but we did not have enough time to investigate before paper submission.

Another technique is to integrate a language model during the inference process. With that being said, we can train an acoustic model on some speech data, and another language model (e.g., n-gram) on some text in the same language as the speech data. Then during evaluation, the language model supports the acoustic model in predicting the transcribed words via beam search. Recent enhancements were made to the huggingface implementation to include a word-based LM in the final search. The procedure described in [22] was used. A 3-gram LM was trained on the MALACH training transcripts. The fine-tuned model used was the one built using all the training data. It was found that an lm weight of .5 produced the best performance - a WER of 14.1%, relative to the no-LM performance of 15.9%.

5. Use of Pseudo-labeled Data

Approximately half of the released MALACH data was never transcribed. Can this unlabeled data be leveraged to further improve recognition performance? Experiments were conducted to test this hypothesis using a three-stage training methodology. In Stage one, the wav2vec2 model was fine tuned on transcribed data. Stage two enriches the size and diversity of the training samples by generating high-quality pseudo-labels for the unlabeled speech utilizing the model fine-tuned from the previous stage. In stage 3, the baseline model is re-fine-tuned on the joint corpus consisting of originally transcribed samples and the pseudo-labeled samples produced from stage two.

Initial experiments were trained on a small subset of the data (1.5 hours) to allow for rapid experiment turnover, as a proof of concept. Table 4 displays Word Error Rate (WER) results on the test set. The baseline WER is 21.1%. Note that this figure, even when fine-tuned on such a small amount of data, is still better than the previous state-of-the-art WER (21.7%) on the MALACH corpus. Then, pseudo-labels were produced by decoding using the 1.5 hour fine tuned model on approximately 10 minutes of unlabeled speech. These pseudo-labels were merged with the 1.5 hour fine tuning set and the model was fine-tuned again, resulting in a 20.6% WER (Table 4).

Table 4: *Pseudo-Labeling WERs using small data subset*

Training Size (Hrs)		**WER(%)**
Original	Pseudo	
1.53	-	21.1%
1.53	0.153	20.6%

Buoyed by these initial results, we decided to try incorporating unlabelled data on a much larger scale. We used the model fine-tuned on all the training data to produce pseudo-labels on all the unlabeled data using two different configurations: one without an LM, and the other, using the 3-gram LM described in Section 4.5. We then decoded the test data with and without a language model. The results are shown in Table 5.

Table 5: *Pseudo-Labeling WERs using all the unlabeled data.*

Training	**No LM WER(%)**	**LM WER(%)**
Baseline	15.9	14.1
LM	15.4	14.5
No LM	15.4	13.5

As can be seen, improvements relative to baseline are seen when pseudo-labels are produced with or without an LM and decoded without using an LM. However, when using an LM to decode, it seems to be substantially better to NOT use an LM to produce pseudo-labels; and actually, performance is degraded when an LM is used both to produce pseudo-labels and for decoding.

Finally, as a additional test, we also used the new test data described in Section 3 to obtain a set of results[5].The new test set is a factor of two larger than the minitest, but reference transcripts were obtained by taking the manual transcriptions verbatim, without another round (or rounds) of cleaning. Therefore, there are potentially some errors remaining in the reference transcripts. The results are shown in Table 6.

Table 6: *Pseudo-Labeling WERs on new test data.*

Training	**No LM WER(%)**	**LM WER(%)**
Baseline	15.3	14.6
LM	15.0	15.1
No LM	15.2	14.1

The results on the new test data are consistent with the minitest data although perhaps less pronounced.

6. Discussion and Future Work

The best results on the old and new test data (13.5% and 14.1%) are certainly remarkable numbers relative to those obtained just a few years ago, and may no longer be a barrier to downstream NLP processing. It seems clear that pretraining on enormous amounts of data significantly improved speech recognition performance on the MALACH corpus. Note also that out of the box, the Whisper system also had excellent performance with a 18.0% WER.

A casual error analysis of the remaining errors suggests that there are still opportunities for transcript cleanup; that sometimes the data is over-segmented making it very hard to recognize some words without more LM context, there are still many errors on foreign named entities, and that sometimes, the talker's accent is so heavy that even skilled transcribers would have difficulty. Although early work on MALACH[4] did not find any clear effect due to other variables, such as background noise or age, the high WER operating point may have masked such effects so they may be worth revisiting.

Work to be done in the future includes fine-tuning on Whisper and other large pretrained systems; resegmenting the data to achieve larger lm contexts; and experimenting with rescoring n-best hypotheses using large language models. We also plan to release the new data in a speech recognition friendly form by paper publication time.

[4] `https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_conformer_transducer_xlarge`

[5] The data only became available close to paper deadline or more comparisons would have been made

7. References

[1] Ramabhadran, Bhuvana, Gustman, Samuel, Byrne, William, Hajič, Jan, Oard, Douglas, Olsson, J. Scott , Picheny, Michael, and Psutka, Josef, "USC-SFI MALACH Interviews and Transcripts English," 2012. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2012S05

[2] "USC Shoah Foundation," https://sfi.usc.edu, accessed: 2023-01-20.

[3] M. Picheny, Z. Tüske, B. Kingsbury, K. Audhkhasi, X. Cui, and G. Saon, "Challenging the Boundaries of Speech Recognition: The MALACH Corpus," in *Proc. Interspeech 2019*, 2019, pp. 326–330.

[4] W. Byrne, D. Doermann, M. Franz, S. Gustman, J. Hajic, D. Oard, M. Picheny, J. Psutka, B. Ramabhadran, D. Soergel *et al.*, "Automatic recognition of spontaneous speech for access to multilingual oral history archives," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 4, pp. 420–435, 2004.

[5] T. Imai, S. Homma, A. Kobayashi, T. Oku, and S. Sato, "Speech recognition with a seamlessly updated language model for real-time closed-captioning," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[6] G. Boulianne, J.-F. Beaumont, M. Boisvert, J. Brousseau, P. Cardinal, C. Chapdelaine, M. Comeau, P. Ouellet, and F. Osterrath, "Computer-assisted closed-captioning of live tv broadcasts in french," in *Ninth International Conference on Spoken Language Processing*, 2006.

[7] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang *et al.*, "A comparative study on transformer vs rnn in speech applications," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 449–456.

[8] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.

[9] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020. [Online]. Available: https://arxiv.org/abs/2006.11477

[10] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356

[11] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*. IEEE, 2015, pp. 5206–5210. [Online]. Available: https://doi.org/10.1109/ICASSP.2015.7178964

[12] Ramabhadran, Bhuvana, Gustman, Samuel, Byrne, William, Hajič, Jan, Oard, Douglas, Olsson, J. Scott , Picheny, Michael, and Psutka, Josef, "Usc-sfi malach interviews and transcripts english – speech recognition edition," 2019. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2019S11

[13] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proceedings of Interspeech*, 2018, pp. 2207–2211. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1456

[14] "Espnet: end-to-end speech processing toolkit," https://github.com/espnet/espnet, accessed: 2022-12-15.

[15] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.6

[16] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[17] J. Lehečka, J. V. Psutka, and J. Psutka, "Transformer-based automatic speech recognition of formal and colloquial czech in malach project," in *Text, Speech, and Dialogue: 25th International Conference, TSD 2022, Brno, Czech Republic, September 6–9, 2022, Proceedings*. Springer, 2022, pp. 301–312.

[18] S. Gandhi, P. von Platen, and A. M. Rush, "Esb: A benchmark for multi-domain end-to-end speech recognition," 2022. [Online]. Available: https://arxiv.org/abs/2210.13352

[19] B. Ramabhadran, "Exploiting large quantities of spontaneous speech for unsupervised training of acoustic models," in *INTERSPEECH 2005 - Eurospeech, 9th European Conference on Speech Communication and Technology, Lisbon, Portugal, September 4-8, 2005*. ISCA, 2005, pp. 1617–1620. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2005/i05_1617.html

[20] J. Kahn, A. Lee, and A. Hannun, "Self-training for end-to-end speech recognition," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7084–7088.

[21] "Hugging face forums," https://discuss.huggingface.co/t/wav2vec2-fix-growing-training-and-validation-loss_-after-few-epochs/8757/6, accessed: 2022-11-01.

[22] "Boosting wav2vec2 with n-grams in huggingface transformers," https://huggingface.co/blog/wav2vec2-with-ngram, accessed: 2022-12-20.