INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Language Identification Networks for Multilingual Everyday Recordings

Kiran Praveen[†], *Balaji Radhakrishnan*[†], *Kamini Sabu, Abhishek Pandey, Mahaboob Ali Basha Shaik*

Samsung R&D Institute Bangalore, India
{k.praveen.t, balaji.r, kamini.sabu, abhi3.pandey, m.shaik}@samsung.com

Abstract

This paper describes the systems SRI-B has proposed for task-1 of the inaugural MERLIon CCS challenge in the closed domain and open domain. Our system for the closed task is based on an end-to-end conformer architecture trained for the task of automatic speech recognition using RNN-T loss, which is then transfer learned for the task of language classification. We train the ASR model initially to ease the task of learning the right features for the classification task. This system achieves a 13.9% Equal Error Rate (EER) and 81.7% Balanced Accuracy (BAC) on the evaluation set. For the open track, we use an ensemble of Open AI's Whisper model and one of the ASR models used our closed track. This system achieves 9.5% EER and 78.9% BAC on the evaluation set. Compared to the challenge baseline we observe relative improvements for EER of 35.9% in the closed track and 56.2% in the open track. We achieve 1$^{\text{st}}$ position on both the closed and the open track leaderboards.

Index Terms: language identification, speech recognition

1. Introduction

Recent advancements in automatic speech processing have resulted in systems which perform very well within the constraints of particular use cases such as voice assistants having great recognition for native speakers. The robustness of these systems is closely related to the availability of training data resources and is largely biased toward standard statistics. Speech recognition for adult speech performs better compared to children's speech due to the scarcity of children's training data. Similarly, people having strong accents suffer terribly in terms of recognition performance. It's always challenging to work with natural, conversational, multilingual, and code-switched speech.

The MERLion CCS challenge focused on the shortcomings of existing systems and presented the challenge to develop robust spoken language identification and diarization systems. The systems should work reliably for non-standard accents, code-switched, and children's speech. As part of this challenge, the organizers have shared audio recordings from the Talk together study where the adults are narrating the onscreen picturebook to children, which consists of natural conversation between parents & children over a video call in a code-switched (English - Mandarin) manner. The language identification track for the challenge is split into closed and open tracks. No extra data can be used in the closed track, whereas in the open track, any publicly available pre-trained models and upto 200 hours of extra data can be utilized. Our key contributions to this paper are as follows,

- Automatic speech recognition(ASR) pre-training for the closed domain.
- Weighted cross-entropy loss to improve balanced accuracy.
- Ensemble of models for the final prediction.
- Utilization of pre-trained Whisper models which perform really well even without domain adaptation.
- Fine-tuning on the development set to significantly improve performance.

2. Related work

Inspired by the success of the acoustic models using deep neural networks (DNNs), [1] proposed DNN based language identification (LID) system and compared it with traditional i-vector based methods. They demonstrated with the increase in training data size DNNs performed better. [2] proposed the LSTM-based model to exploit the sequence modeling capability by capturing the contextual information. On the contrary, [3] used the CNN models to extract the bottleneck features and used it along with other acoustic features to train GMM-i-vector LID system. To get the best of both in terms of feature extraction and capturing temporal information, in [4] uses Convolution and LSTM (CLSTM) along with time & frequency domain attention mechanism. Recently [5] publish the architecture where the encoder learns the intermediate vector representation using 1D depth-wise separable convolutions and squeeze-and-excitation layers then pass it to a classifier for language identification.

[6] and [7] show that the acoustic representations learned by the neural network from audio learn the accent and hence fail to discriminate in case of non-native speech. The system performance is inversely correlated to the strength of the accent and is still poor even if the non-native speakers are proficient.

[8] uses MFCC and Fbank features in the RNN-T model to come up with language embeddings. They use statistical pooling to combine the frame-level encoder outputs with ASR predictions at the utterance level. Other representations like CPC (contrastive predictive coding) [9] and mel frequency spectral features [10] have been shown to possess more language discriminative power than MFCC, especially in cross-domain scenarios.

Considering the reliability of the ASR hypothesis for language identification, joint ASR and LID training has been considered by [11]. On similar lines [12] use cascaded encoder-based RNN-T model for frame-level language identification and use the decisions for 2nd stage of the ASR task. Instead of joint training, the approach of adding a lightweight classifier component on a pre-trained RNN-T-based ASR model has been considered in [13].

Various self-supervised learning (SSL) models have been

[†] denotes equal contribution

10.21437/Interspeech.2023-2047

proposed where the model inherently learns very good language representations by consuming huge amounts of unlabelled data. [14] present a wav2vec 2.0 [15] based SSL model trained on multilingual datasets which outperforms the previous wav2vec 2.0 model [15] on LID tasks. [16] studied the use of wav2vec representation for LID tasks. [17] propose a very large model trained on a massive multilingual dataset of $680k$ hours. The network is trained in a multi-task fashion and can perform transcription, translation, and language identification.

3. Data description

The challenge dataset is child-directed speech where parents narrate a picture-book to their children. Audios are recorded at home using Zoom video-conferencing on internet-enabled personal electronic devices including laptops, tablets and mobile phones. Environmental background noise varied widely during recordings and these recordings were in far-field conditions.

There are 305 recordings from 112 parent-child pairs comprising 25 hours of English and 5 hours of Mandarin speech. There is more than one recording for 103 pairs, with a maximum of three recordings per pair. Both English and Mandarin speech feature Singaporean variety leading to pronunciation that is different from standard English and Mandarin. The vocabulary and grammar may also be unique. There is frequent code-switching within and between utterances, with the proportion of Mandarin ranging from 0.85% to 80.7% per utterance.The utterances are mostly short with 20% utterances being less than 500ms long. The average utterance duration is 1.4 seconds for English and 1.2 seconds for Mandarin.

The recordings are mostly clean with a few exceptions like beeps, table tapping,and microphone tapping noises. Many recordings have the parent's speech followed by the children crying, babbling or laughing. The parent readings are very expressive and sometimes even carry song-like rhythms. Quite a few utterances consist of the child reading or trying to read. A few utterances have both the parents and the children speaking. Most of the short utterances are, however, single-syllable sounds like okay, yeah, ohh, hmm, right, etc.

The challenge dataset consists of 2 splits: development and evaluation, which we shall sometimes refer to as dev and eval respectively. The development split was provided for assessing and training the models. The evaluation set labels were hidden from the participants and this was the set that was used to rank the submissions on the leaderboards. There are a total of 50270 utterances in the development set and 48785 utterances in the evaluation set. Both the development and evaluation sets have a very similar distribution but with no speaker overlap amongst them.

Table 1: *Publicly available datasets used for training*

Dataset	Language	Hours
Librispeech (train-clean-100)	English (US)	100
NSC (preselected partition)	English (SG)	100
AIShell (preselected partition)	Mandarin	200

For closed track training, we use a combination of the publicly available datasets mentioned in Table 1 and the provided development set. We do not use any data mentioned in Table 1 for open track training and stick to only the provided development set.

4. System description for closed track

4.1. Signal domain transformations

- **Mel-Filter Banks**: Short Time Fourier Transform was performed using 25ms windows and 10ms hops, and then 80 filter banks in the mel scale, ranging from 75Hz to 8000Hz were used for generating input features.
- **SpecAugment**: The commonly used spectrogram augmentation scheme is utilized in hopes of improving the performance.
- **Utterance normalization**: After the previous 2 stages, the utterance is mean-variance normalized on a per utterance level, independently for every dimension.

During testing, the SpecAugment stage is removed from the preprocessing steps.

4.2. Model architecture

For the experiments, we use a base ASR model that closely resembles Conformer-S [18]. The only difference is that the decoder is replaced with a transformer with 320 hidden dimensions. The transcription network consists of 12 conformer blocks and uses relative positional encoding, while the prediction network has 1 transformer block and uses normal positional encoding.

4.3. Training

4.3.1. Phase 1

For the initial phase, we train a multilingual ASR network, with Byte-Pair Encodings as the targets. The only notable difference here is that before generating any linguistic token, the model is trained to output the language ID. For example, an English sentence would effectively be

```
<SOS> <EN> HELLO WORLD <EOS>
```

where $< SOS >$ and $< EOS >$ are start-of-sequence and end-of-sequence tokens respectively. The model is trained using RNN-T loss. Tokens for text was generate using Byte-Pair Encoding[19] generated using all the text data from the train set.

For validating the training, we leave out roughly 50% (equal splits from each dataset) of the training data and train on the rest. During this stage the development set for the challenge is not used.

4.3.2. Phase 2

After an ASR model has been trained, the transcription network is extracted and a single self attention layer is attached on top. The output of this layer is averaged over the time-steps to produce a single vector, which is then transformed to 2 dimensions via a fully connected network. Weighted cross entropy loss is applied with weights roughly proportional to the inverse of the number of utterances as weights for each language (Mandarin=0.8, English=0.2).

During this stage, the development set provided for the challenge is exclusively used. For estimating the number of epochs required for convergence, we initially split the dev set into 50% for training and 50% for validation. The data ratios of Mandarin and English are maintained in the split.

Once the model has been trained and the number of epochs required for best convergence is identified using the 50-50 split, we start again from the trained ASR. If we identify that the 50-50 train-val split pilot experiment indicated K epochs for con-

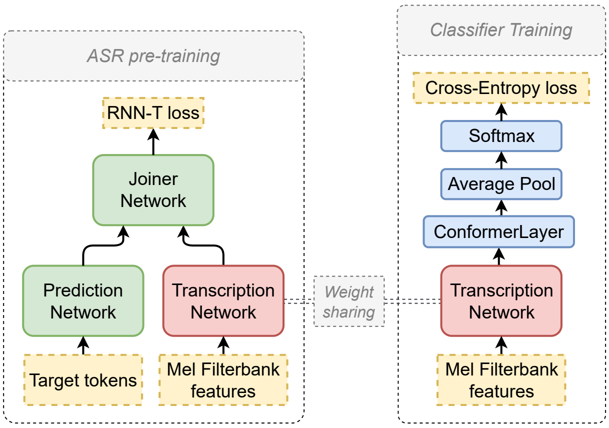

Figure 1: *Model architecture used for training the final closed track system.*

vergence, we train a model for K epochs using all of the development set. The evaluation set is then inferred using this model. The overview of the architecture is given in figure 1.

4.4. Approaches

Four approaches were tried for the experiments.

(a) **Classifier with *DEV* only**: Training the classifier network directly using the development set.

(b) **Pre-trained ASR finetuned on *DEV***: Training an ASR with all except development set, then fine-tuning the ASR to predict only the language token using only the development set.

(c) **Pre-trained ASR transfer learned to a classifier with *DEV***: Training an ASR with all except development set, then transfer learning onto the development set using 2 extra layers.

(d) **Ensemble of 2 transfer learned classifiers**: An ensemble of 2 models trained as in approach *(c)*. The probability predictions are combined using arithmetic mean.

Approach *(d)* was finalized after testing on the unseen split.

5. System description for open track

5.1. Open track models

This section describes the models tried for the open track. All the *fine-tuned* models mentioned in subsequent sections are fine-tuned on the development dataset. HuggingFace [20] and PyTorch [21] were used to implement all the open track models described below.

5.1.1. Whisper-large

For the first 2 networks in the open track, we harness the powerful multilingual capabilities of OpenAI's Whisper model [17]. A pre-trained Whisper-large model initialized with a language modeling head was chosen as the first network. This is a 32 layer network consisting of 1.55B parameters.

5.1.2. Whisper-base fine-tuned

This network consists of a Whisper-base model followed by a MLP projection head and finally a classification layer. The Whisper-Base model is a 6 layer network with 74M parameters. The MLP projection head consists of a fully connected layer network that projects to a higher dimensional latent space (size=2048) and then being projecting it back to a lower dimension (size=256) with ReLU as the non-linearity for both the layers. Another fully connected layer finally maps to raw logits for classification.

5.1.3. Wav2vec2-Conformer-large fine-tuned

A Wav2vec2 Conformer is an extension to the original Wav2vec2 model [15] where the attention-blocks are replaced by the conformer-blocks introduced in Conformer [18]. This model contains 24 transformer blocks with model dimension of 1024 and 16 attention heads. A fully connected layer projecting to 2 dimensions is added on top of the pre-trained model for fine-tuning.

5.1.4. Wav2vec2-large-XLSR fine-tuned

The XLSR-Wav2vec2 model was first proposed in [14] and builds on the conventional Wav2vec2 model with the help of cross-lingual pre-training. The large model once again consists of 24 transformer blocks with model dimension of 1024 and 16 attention heads. Similar to the previous network, a fully connected layer is added on top of the pre-trained model for the classification task.

5.1.5. WavLM-base-plus model fine-tuned

WavLM was a large scale self-supervised pre-training algorithm introduced in [22] to specifically tackle full-stack downstream speech tracks. The WavLM-base-plus architecture consists of 12 transformer encoder layers with a hidden state dimension of 768 and 8 attention heads. Keeping in line with the previous 2 networks, a fully connected layer is added on top to perform the classification.

5.2. Training

The pre-trained Whisper-large (5.1.1) network is utilized for language classification without the need for any new training. We directly infer on the evaluation set and report the results.

For the remaining 4 networks, since they are all already pre-trained, we utilize only the development set provided for training and evaluation. We partition the development set into 2 splits of 90% and 10% for training and validation respectively. We utilize stratified sampling to ensure that both splits possess the same distribution of classes. For pre-processing, we begin by resampling the speech to 16khz and then compute log mel filter banks of 80 dimensions which are in turn used as the inputs for our network. In addition, for the Wav2vec2-Conformer-large, Wav2vec2-large-XLSR, and the WavLM-base-plus models, we truncate/pad audios to a maximum duration of 2 seconds.

All the networks are trained for 10 epochs and the model with the best validation accuracy is chosen for inference. Adam [23] is the optimizer of choice for all the training experiments. Weighted cross entropy with a ratio of $4 : 1$ is used in order to account for the class imbalance. $2e - 5$ was chosen as the learning rate for training the Whisper-base model, whereas in all the other cases, we resorted to a learning rate of $3e - 5$. The Whisper-base network was trained with a batch size of 16. Wav2vec2-Conformer-large and WavLM-base-plus both utilize a batch size of 64, with gradient being accumulated and updated every 4 steps. Wav2vec2-Conformer-large follows the same number of steps for gradient accumulation with only the batch

Table 2: *Results for all closed track models. DEV set used is a custom 50% split.*

Model	Dev		Eval	
	EER (%)	BAC (%)	EER (%)	BAC (%)
Baseline	–	–	21.7	50.9
Classifier trained with *DEV* only	21.3	65.9	–	–
Pre-trained ASR finetuned on *DEV*	19.7	51.4	–	–
Pre-trained ASR transfer learned to a classifier with *DEV*	14.7	79.8	14.3	–
Ensemble of 2 transfer leanred classifiers	14.3	79.6	13.9	81.7

Table 3: *Results for all open track models. DEV set used is a custom 10% split for all but the last row for the ensemble. Since the ensemble uses the closed track model, the DEV set chosen is the same 50% split used for closed track.*

Model	Dev		Eval	
	EER (%)	BAC (%)	EER (%)	BAC (%)
Baseline	–	–	21.7	50.9
Wav2vec2-Conformer-large fine-tuned	13.1	73.2	–	–
Wav2vec2-large-XLSR fine-tuned	13.6	71.2	–	–
WavLM-base-plus model fine-tuned	14.4	67.5	–	–
Whisper-large	12.8	71.8	12.1	–
Whisper-base fine-tuned	9.3	80.9	11.6	–
Whisper-large + closed track ensemble	10.7	77.4	9.5	78.9

size being reduced to 32. For the final submission to the open track leaderboard, the probability scores from the pre-trained Whisper-large model and one of the closed track models are combined using an arithmetic mean.

6. Results

6.1. Closed track results

Table 2 presents results for all closed track models mentioned in *Section 4.4* . For the closed track, since we train on half of the development set, the results shown with the development set indicate performance on the left out validation split The performance of the models on evaluation split is shown only for the models we submitted.

The ensemble model is the winning submission which topped the leaderboard.

6.2. Open track results

Table 3 showcases results for all open track models. For the open track, we train on 90% of the development set and the results displayed below are on the left out validation split of 10%. The performance of the models on evaluation split are shown only for the models that were submitted.

The Whisper-large + closed track ensemble was the winning submission which topped the leaderboard. Due to the limited number of submissions allowed, we were unable to try an ensemble involving the Whisper-base finetuned model but expect it to perform on par with or better than our winning submission.

7. Discussion

In the closed track, we observe that pretraining an ASR has a good impact on the performance. However, when fine-tuning the ASR model using RNN-T loss to predict a single token, we don't observe much performance improvements over chance accuracy. This could be overcome by modifying the loss to include weighted loss for RNN-T. The model which is first trained as an ASR and then transfer learned to be classifier using weighted cross-entropy performs the best. In our experiments, we also observed that without providing weights for the loss, the model converges to a point where the performance is close to chance accuracy.

The open track models mostly outperform the closed track models when fine-tuned on only the development set. In self-supervised pre-trained models, the Wav2vec2-Conformer variant performs the best, highlighting the effectiveness of conformer blocks in speech-related tasks. Pre-training on 53 different languages appears to be benefitting the Wav2vec2-XLSR model when compared to the WavLM model. Whisper models, owing to the vast amount of multi-lingual training data and the language identification capabilities, benefit hugely in terms of EER. Wav2vec and WavLM, pre-trained in a self-supervised manner, appears to put them at a disadvantage in comparison. The fine-tuned Whisper-base outperforms the much larger Whisper-large (without fine-tune) on both the development and the evaluation sets. Ensemble approaches significantly improve the performance in terms of EER as seen from the last row in *Table 3*, possibly because of the complementary information learned by the models.

Similar to [13], we observe that our models perform better with longer utterances. This might be indicative of the ASR encoder's ability to capture linguistic signatures with longer context, helping the overall performance.

8. Conclusions

This paper presents the challenge winning approaches for both the open and closed tracks for the MERLion CCS Language Identification Challenge. We bring together several key ideas like ASR pre-training, utilization of Whisper networks, ensemble across the open and closed tracks etc. to achieve a relative EER improvement of 35.9% and 56.2% in the closed and open tracks respectively over the challenge baseline. We also clearly demonstrate the improvement that each of our ideas bring forth and hope that some of them will be made use of in the future for similar tasks and challenges.

9. References

[1] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez, and P. Moreno, "Automatic language identification using deep neural networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 5337–5341.

[2] J. Gonzalez-Dominguez, I. Lopez-Moreno, and H. Sak, "Automatic language identification using long short-term memory recurrent neural networks," 2014.

[3] S. Ganapathy, K. Han, S. Thomas, M. Omar, M. V. Segbroeck, and S. S. Narayanan, "Robust language identification using convolutional neural network features," in *Fifteenth annual conference of the international speech communication association*, 2014.

[4] X. Miao, I. McLoughlin, and Y. Yan, "A new time-frequency attention mechanism for tdnn and cnn-lstm-tdnn, with application to language identification." in *Interspeech*, 2019, pp. 4080–4084.

[5] F. Jia, N. R. Koluguri, J. Balam, and B. Ginsburg, "Ambernet: A compact end-to-end model for spoken language identification," *arXiv preprint arXiv:2210.15781*, 2022.

[6] C. Chandak, Z. Raeesy, A. Rastrow, Y. Liu, X. Huang, S. Wang, D. K. Joo, and R. Maas, "Streaming language identification using combination of acoustic representations and ASR hypotheses," *arXiv preprint arXiv:2006.00703v1*, 2020.

[7] K. Kukk and T. Alumäae, "Improving language identification of accented speech," in *Proc. INTERSPEECH 2022 – 23rd Annual Conference of the International Speech Communication Association*, Incheon, Korea, Sep. 2022, pp. 1288–1292.

[8] P. Shen, X. Lu, and H. Kawai, "Transducer-based language embedding for spoken language identification," in *Proc. INTERSPEECH 2022 – 23rd Annual Conference of the International Speech Communication Association*, Incheon, Korea, Sep. 2022, pp. 3724–3728.

[9] M. de Seyssel, M. Lavechin, Y. Adi, E. Dupoux, and G. Wisniewski, "Probing phoneme, language and speaker information in unsupervised speech representations," in *Proc. INTERSPEECH 2022 – 23rd Annual Conference of the International Speech Communication Association*, Incheon, Korea, Sep. 2022, pp. 1402–1406.

[10] B. M. Abdullah, T. Avgustinova, B. Möbius, and D. Klakow, "Cross-domain adaptation of spoken language identification for related languages: The curious case of Slavic languages," in *Proc. INTERSPEECH 2020 – 21st Annual Conference of the International Speech Communication Association*, Shanghai, China, Oct. 2020, pp. 477–481.

[11] R. Duroselle, M. Sahidullah, D. Jouvet, and I. Illina, "Modeling and training strategies for language recognition systems," in *Proc. INTERSPEECH 2021 – 22nd Annual Conference of the International Speech Communication Association*, Brno, Czechia, Sep. 2021, pp. 1494–1498.

[12] C. Zhang, B. Li, T. Sainath, T. Strohman, S. Mavandadi, S. Chang, and P. Haghani, "Streaming end-to-end multilingual speech recognition with joint language identification," in *Proc. INTERSPEECH 2022 – 23rd Annual Conference of the International Speech Communication Association*, Incheon, Korea, Sep. 2022, pp. 3223–3227.

[13] Z. Kons, H. Aronowitz, E. Morais, M. Damasceno, H.-K. Kuo, S. Thomas, and G. Saon, "Extending rnn-t-based speech recognition systems with emotion and language classification," in *Proc. INTERSPEECH 2022 – 23rd Annual Conference of the International Speech Communication Association*, Incheon, Korea, Sep. 2022, pp. 546–549.

[14] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, and M. Auli, "XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale," in *Proc. Interspeech 2022*, 2022, pp. 2278–2282.

[15] 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.

[16] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," in *Proc. INTERSPEECH 2021 – 22nd Annual Conference of the International Speech Communication Association*, Brno, Czechia, Sep. 2021, pp. 1509–1513.

[17] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," *arXiv preprint arXiv:2212.04356*, 2022.

[18] 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.

[19] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *Association for Computational Linguistics*, vol. 1, 2016.

[20] 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

[21] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[22] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.