ANALYSIS OF XLS-R FOR SPEECH QUALITY ASSESSMENT

Bastiaan Tamm,[1,2*] *Rik Vandenberghe,*[1] *Hugo Van hamme,*[2]

[1] Laboratory for Cognitive Neurology (LCN), Department of Neurosciences, KU Leuven, Belgium
[2] Processing Speech and Images (PSI), Department of Electrical Engineering, KU Leuven, Belgium

ABSTRACT

In online conferencing applications, estimating the perceived quality of an audio signal is crucial to ensure high quality of experience for the end user. The most reliable way to assess the quality of a speech signal is through human judgments in the form of the mean opinion score (MOS) metric. However, such an approach is labor intensive and not feasible for large-scale applications. The focus has therefore shifted towards automated speech quality assessment through end-to-end training of deep neural networks. Recently, it was shown that leveraging pre-trained wav2vec-based XLS-R embeddings leads to state-of-the-art performance for the task of speech quality prediction. In this paper, we perform an in-depth analysis of the pre-trained model. First, we analyze the performance of embeddings extracted from each layer of XLS-R and also for each size of the model (300M, 1B, 2B parameters). Surprisingly, we find two optimal regions for feature extraction: one in the lower-level features and one in the high-level features. Next, we investigate the reason for the two distinct optima. We hypothesize that the lower-level features capture characteristics of noise and room acoustics, whereas the high-level features focus on speech content and intelligibility. To investigate this, we analyze the sensitivity of the MOS predictions with respect to different levels of corruption in each category. Afterwards, we try fusing the two optimal feature depths to determine if they contain complementary information for MOS prediction. Finally, we compare the performance of the proposed models and assess the generalizability of the models on unseen datasets.

Index Terms— speech quality assessment, MOS prediction

1. INTRODUCTION

Given the increased dependence on online conferencing applications in recent years, the demand for a reliable automated method to assess perceived speech quality has grown. Common factors that can degrade conversational quality include jitter, latency, echo, packet loss, and distortion [1]. The ground truth for perceived speech quality is derived from human judgments, usually in the form of Absolute Category Ratings (ACR). These ratings are used to calculate the mean opinion score (MOS), which is used as the ground truth for the perceived speech quality of a given audio sample. However, the collection of human judgments is extremely time- and labor-intensive, making it impractical for large-scale evaluations of speech quality. Many objective metrics such as the speech-to-reverberation modulation energy ratio do not necessarily correlate with the perceived speech quality [2]. Efforts have therefore been dedicated towards machine learning approaches for speech quality assessment [3, 4], for example based on long short-term memory (LSTM) networks [5]. In recent work on the ConferencingSpeech 2022 challenge [6], it was shown that leveraging pre-trained wav2vec-based XLS-R [7] embeddings leads to state-of-the-art performance for the MOS prediction task [8]. Most notably, the model performed exceptionally well on the unseen TUB dataset, outperforming the next-closest competitor by 27.4% and the overall second-place model by 42.9% for the RMSE metric [6, 8]. Whether the embeddings generalize well to other unseen datasets has not yet been investigated.

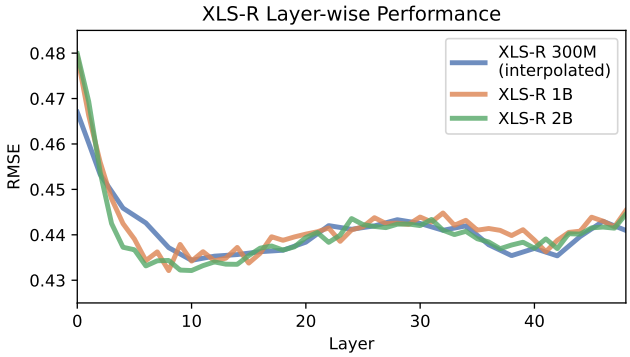

Figure 1: Layer-wise performance of pre-trained XLS-R models on speech quality assessment task. The performance of each layer's activations is plotted for the three model sizes. This is measured using the best validation RMSE from all model configurations. This analysis was done on 35% of the full dataset.

This paper aims to perform an in-depth analysis of the wav2vec-based XLS-R model for the task of speech quality assessment. We first note that the original paper [8] simply used the final hidden layer of the pre-trained 300M parameter XLS-R to train a MOS-prediction model instead of determining a feature depth that is optimal for the downstream task. We will therefore analyze the performance of embeddings extracted from each layer of XLS-R and also for each size of the pre-trained model (300M, 1B, 2B parameters). Also, we aim to validate the performance of the model on other unseen datasets.

2. EXPERIMENTAL SETUP

2.1. Datasets

We use the same four corpora as the ConferencingSpeech 2022 challenge [6], specifically

- the **Tencent corpus**, consisting of around 14,000 Chinese

*This research was supported by KU Leuven Special Research Fund grant C24M/22/025 and the Flemish Government under Onderzoeksprogramma AI Vlaanderen.

979-8-3503-2372-6/23/$31.00 ©2023 IEEE

speech clips with and without reverberation, ranging from 5 to 13.5 seconds long;

- the **Public switched telephone network (PSTN) corpus**, consisting of 80,000 English speech samples based on LibriVox with a length of 10 seconds each, containing both clean samples as well as samples with artificial background noise;
- the **Non-intrusive speech quality assessment (NISQA) corpus** [9], a collection of more than 14,000 English and German speech clips with real as well as simulated noise;
- the **IU Bloomington (IUB)** corpus, comprising 36,000 English speech samples from the VOiCES [10] and COSINE [11] datasets, ranging from three to six seconds long.

Each corpus is labeled with MOS values in a rating range of 1–5, which are derived from ACRs based on the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) recommendation P.808 [12]. The only exception is the IU Bloomington corpus, which follows ITU-R BS.1534 [13] and has a range of 0–100. These are converted to the range 1–5 for the experiments. Additionally, the count and standard deviation of the ACRs is provided for each audio sample in all corpora except for Tencent.

2.2. Dataset Division

We use the same approach to dataset division as [8]. We define the challenge subset as the combination of the Tencent and PSTN corpora and the full dataset as the combination of all four corpora. The full training and validation sets are constructed by shuffling the samples in the full dataset and using 85% for training and 15% for validation. Finally, the training and validation subsets are constructed by keeping only the Tencent and PSTN samples from the original training and validation sets. We define an additional dataset, which will be referred to as the unseen dataset, which only contains the samples from NISQA and IUB. Thus, the models will be trained on the challenge subset and evaluated on the "unseen" datasets.

2.3. Model Architecture

The model takes as input a sequence of 384 extracted features, which can be pre-trained XLS-R embeddings from a specific layer or MFCC features for comparison. The features are extracted as a preprocessing step and are not finetuned. The speech quality prediction model consists of three modules: a linear down-projection to the size of the hidden space, a bidirectional LSTM (Bi-LSTM) or transformer module to model temporal dependency, and an attention-based pooling module [9] to map to the output space.

The number of Bi-LSTM or transformer layers as well as the hidden size are varied across models. The best transformer models use a hidden size of 32, 4 attention heads and 4 layers; the best Bi-LSTM models use a hidden size of 32 in each direction and 2 layers. We apply batch normalization at the input and after the Bi-LSTM/transformer. Subsequently, the outputs from the attention pooling module are mapped to the range (0,1) with a sigmoid function. These final outputs are referred to as normalized MOS values. This intermediate normalization step ensures that the output range of the attention pooling layer is unrestricted. During the evaluation of the model, the normalized MOS predictions are mapped to the original 1–5 MOS range.

2.4. Training Details

For MFCC calculation, we use the implementation by torchaudio [14] with the default parameters and a sample rate of 16 kHz. The XLS-R feature extraction uses the *facebook/wav2vec2-xls-r-{300m,1b,2b}* models available on HuggingFace [15].

The models are implemented using the PyTorch (v.1.11.0) and PyTorch Lightning (v.1.8.6) libraries in Python 3.9. Training is performed using the PyTorch Lightning trainer. The network is trained using the ADAM optimizer [16], a learning rate of 3×10^{-3}, batch size of 60, and MSE loss. Each model is trained for a total of 30 epochs, and the model with the lowest validation loss is selected.

3. XLS-R LAYER-WISE PERFORMANCE

First, we look at the performance of embeddings extracted from each layer of XLS-R and also for each size of the model (300M, 1B, 2B parameters). The results of the training are shown in Figure 1. On the horizontal axis, the layer from which the activations are taken is displayed. This includes the output of the CNN (layer 0) and all transformer layers (layer 1–48). We interpolate the results of XLS-R 300M for this visualization since this model only has 24 transformer layers. On the vertical axis, we display the performance of the downstream speech quality prediction model on the validation split of the full dataset (RMSE metric, lower is better).

We hypothesized that the performance would rapidly improve over the first few layers and reach an optimum in the lower- or mid-level features, where room acoustics and noise characteristics are best modeled. Then, we believed that the performance would gradually degrade as the layer index further increased, as the highly contextualized speech representations would probably be less suitable to detect localized sources of speech degradation.

The first part of the hypothesis appears to be validated across the three model sizes, as the models reach an optimum around layer 10 (layer 5 for XLS-R 300M). Surprisingly, there seems to be a second local optimum around layer 41 (layer 21 for XLS-R 300M). It appears that there is a certain level of contextualization that is beneficial for speech quality assessment. In the following sections, we will investigate the properties of the low-level and high-level XLS-R embeddings.

4. CORRUPTION SENSITIVITY ANALYSIS

Next, we investigate the reason for the two distinct optima in Figure 1. We hypothesize that the lower-level features better model the more typical conditions that affect the quality of audio in online applications (e.g., room acoustics, echo, packet loss, distortion) and that the higher-level features capture some sort of speech content and intelligibility. Moreover, we expect the two levels of representation to have complementary information and that a fusion model will outperform each feature level individually. This section focuses on the first hypothesis, while Section 5 addresses the latter.

To investigate what types of information are represented at each level, we artificially inject different types of noise and corruption and observe what effect this has on the predictions. We test a variety of corruption techniques: white Gaussian noise, overlapping speech, simulated reverb, low-/high-pass filter, time masking, and MP3 compression. The goal is to determine if a particular feature depth is more sensitive to certain types of degradation. The implementation is done using the audiomentations[1] Python package.

[1] https://github.com/iver56/audiomentations

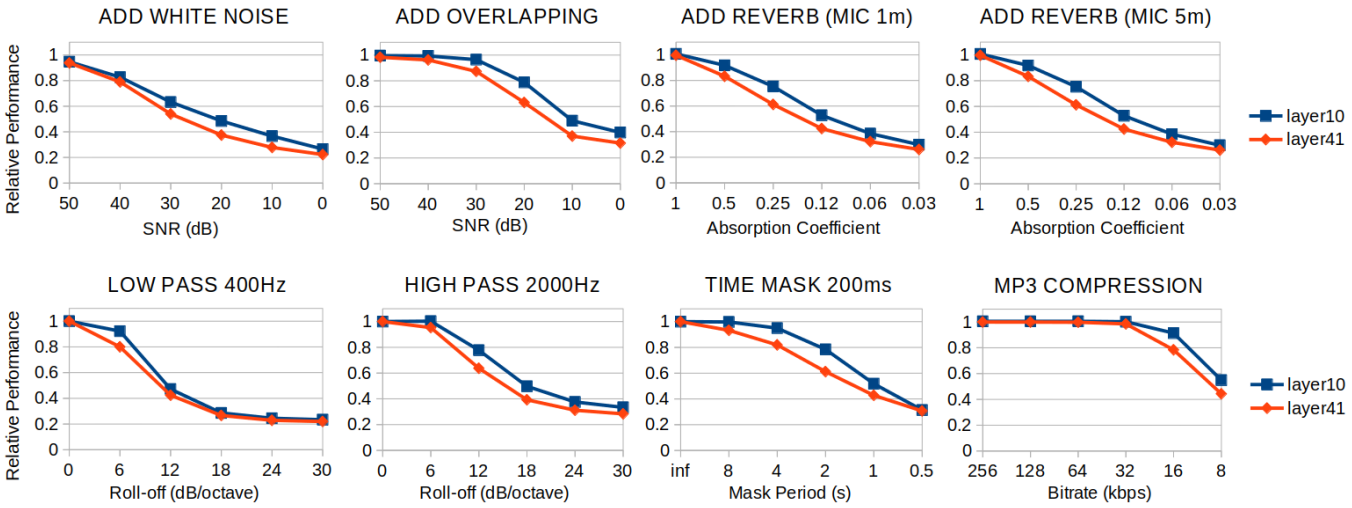

Figure 2: Corruption sensitivity analysis of 2B-parameter model for full dataset.

To estimate the sensitivity, we keep the ground-truth labels fixed and calculate the RMSE between the predictions of the corrupted audio and the ground-truth labels. Naturally, as the level of corruption increases, the predictions will be affected and the RMSE will increase. For visualization, we plot the relative performance of the corrupted predictions compared to the original predictions. A constant value of 1 indicates that the model is insensitive to the injected corruption, whereas a steep negative slope means the predictions are highly sensitive to the corruption.

The results are shown in Figure 2. It can be seen that the model based on high-level XLS-R embeddings (orange line) is more sensitive to all types of corruption, a finding which does not directly support the hypothesis. It seems that perturbations in a given hidden layer of XLS-R propagate and may slightly magnify in later layers. This result can most likely be attributed to the fact that the wav2vec2 training procedure is not designed to generate embeddings that are insensitive to degraded audio. During pre-training, the model is trained to predict masked quantized representations derived from raw audio. The only robustness we would expect is to masking in the quantized representation space. Apparently, this does not translate to insensitivity to time masking, but this could also be due to the relatively long mask window.

This gives an idea of why the XLS-R embeddings are useful for speech quality assessment in the first place. We expect that a noise-robust version of wav2vec2, such as those proposed in [17, 18, 19], would not be useful for speech quality assessment. We have seen in our experiments, for example, that the version of XLS-R finetuned on multilingual speech translation[2] performs very poorly when embeddings are extracted from the final hidden layer, presumably because these embeddings focus on speech content and are more invariant to noise characteristics.

5. MODEL COMPARISON

To assess if the two feature depths contain complementary information for speech quality assessment, we developed layer-fusion models and compared their performance to the single-layer models. We also included the performance of the baseline model from [8] and the popular DNSMOS [20] model for comparison. As a final metric for comparison, we calculate the RMSE of human annotations with respect to the mean opinion score. This can be derived for all corpora except for Tencent since the count and standard deviation of the ACRs are provided per audio sample.

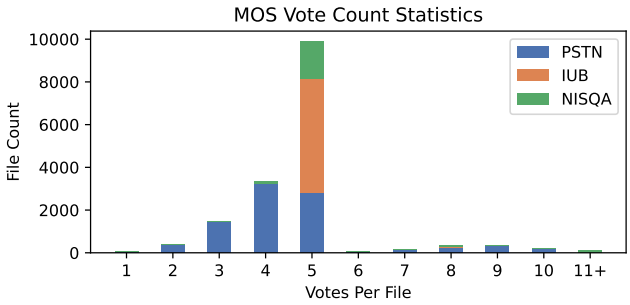

Figure 3: MOS vote count statistics for PSTN, IUB and NISQA validation sets. No counts were available for the Tencent corpus.

$$RMSE_{\text{human}} = \sqrt{\frac{\sum_j s_j^2 \cdot (N_j - 1)}{\sum_j N_j}} \quad (1)$$

The expression $s_j^2 \cdot (N_j - 1)$ is equal to the sum of squared errors for a given audio sample j with respect to the MOS (s_j is the Bessel-corrected standard deviation). We sum this expression over all samples j to obtain the global sum of squared errors and divide by the total number of votes to obtain the mean squared error. This is followed by a square root operation to obtain the desired metric. Contrary to the model predictions, humans are generally restricted to integer scores, so this must be considered as an extra source of variance. Finally, a histogram of the number of votes per audio sample is shown in Figure 3.

The model comparison results are shown in Table 1. Values of interest are shown in bold, and the overall best model per column is underlined as well. We achieve slightly better performance than the

[2]https://huggingface.co/facebook/wav2vec2-xls-r-2b-22-to-16

Group	Train Set	XLS-R Layer	Tencent	PSTN	NISQA	IUB	Subset	Unseen	Full
Baseline [8]	subset	24	**0.3037**	**0.5022**	**0.5907**	**0.5067**	**0.4759**	**0.5323**	*(0.5000)*
DNSMOS [20]	/	/	*(0.8338)*	*(0.7037)*	**0.8718**	**0.5452**	*(0.7262)*	**0.6565**	*(0.6982)*
MFCC	full	/	0.5932	0.5924	*(0.6734)*	*(0.3854)*	0.5925	*(0.4865)*	**0.5511**
Transformer	subset	/	0.5762	0.5992	0.8280	0.7775	0.5955	0.7924	*(0.6840)*
	full	5	0.3340	0.5002	*(0.4251)*	*(0.3711)*	0.4774	*(0.3875)*	0.4423
	full	21	0.3119	**0.4953**	*(0.4310)*	*(0.3758)*	**0.4706**	*(0.3925)*	0.4400
XLS-R 300M	full	5+21	0.3115	0.4976	*(0.4148)*	*(0.3768)*	0.4726	*(0.3882)*	**0.4396**
Transformer	subset	5	0.3212	0.5036	0.6256	0.5049	0.4790	0.5425	*(0.5063)*
	subset	21	0.3003	0.5068	0.5694	0.5025	0.4796	0.5227	*(0.4979)*
	subset	5+21	**0.2948**	0.5055	**0.5683**	**0.4886**	0.4779	**0.5129**	*(0.4927)*
	full	10	0.3127	**0.4988**	*(0.4285)*	*(0.3676)*	**0.4738**	*(0.3862)*	**0.4396**
	full	41	**0.3014**	0.5007	*(0.4389)*	*(0.3689)*	0.4743	*(0.3904)*	0.4415
XLS-R 1B	full	10+41	0.3188	0.5021	*(0.4658)*	*(0.3983)*	0.4774	*(0.4189)*	0.4541
Transformer	subset	10	0.3198	0.5126	**0.5456**	0.5815	0.4868	0.5713	*(0.5235)*
	subset	41	0.3168	0.5118	0.5657	**0.4656**	0.4858	**0.4966**	*(0.4903)*
	subset	10+41	0.3380	0.5050	0.5748	0.5288	0.4821	0.5425	*(0.5080)*
	full	10	0.3520	0.5139	*(0.4717)*	*(0.3739)*	0.4915	*(0.4046)*	0.4575
	full	41	0.3236	**0.4992**	*(0.4297)*	*(0.3813)*	**0.4754**	*(0.3959)*	**0.4442**
XLS-R 2B	full	10+41	0.3111	0.5037	*(0.4217)*	*(0.3987)*	0.4780	*(0.4055)*	0.4494
Transformer	subset	10	0.3034	0.5175	0.6277	0.4899	0.4894	0.5334	*(0.5081)*
	subset	41	**0.2977**	0.5054	**0.5724**	0.4897	0.4781	0.5150	*(0.4937)*
	subset	10+41	0.3069	0.5031	0.6036	**0.4743**	0.4770	**0.5150**	*(0.4931)*
Human	/	/	/	*(0.7889)*	**0.6738**	**0.6573**	/	**0.6629**	/
without quantization	/	/	/	*(0.7342)*	0.6088	0.6571	/	/	/

Table 1: Model comparison for each corpus individually and the challenge subset (Tencent+PSTN), unseen (NISQA+IUB), and full datasets. The metric is RMSE on the respective validation set, lower is better. Values of interest are shown in bold. The overall best model per column is underlined. Some values are not relevant for the discussion but are provided for completeness: these are displayed in a smaller font between parentheses. For example, models trained on the full dataset have technically seen the so-called "unseen" dataset. Also, comparing DNSMOS, which has not been trained on the challenge subset, to models where this is the case would not be a fair comparison. The final rows display the "RMSE" of the human annotations and the estimated human RMSE without integer limitation (modeled as uniform quantization noise).

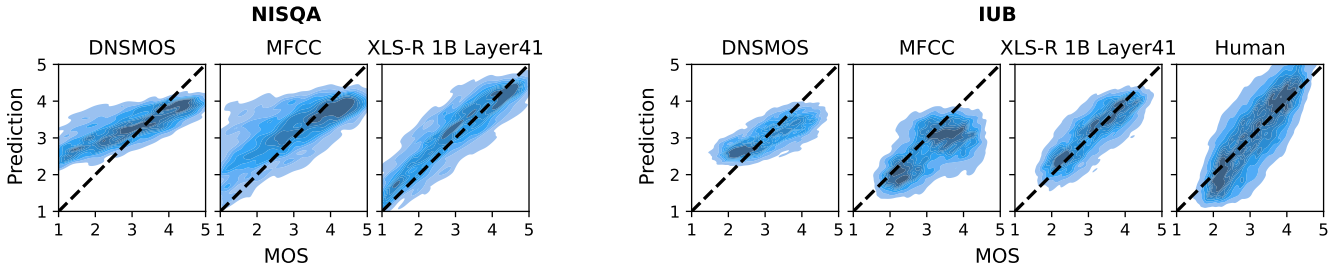

Figure 4: Visualization of MOS predictions on unseen corpora. The human ACRs are also visualized for the IUB corpus.

baseline on the challenge subset (1.1% better). More importantly, we validate the claim that the XLS-R-based model indeed performs exceptionally well on unseen data [8]. We show that the baseline model achieves an RMSE of 0.5323 on the validation set of the unseen NISQA+IUB corpora. This already outperforms DNSMOS (0.6565) and the RMSE of human annotations (0.6629). XLS-R 1B Layer41 performs even better with an RMSE of 0.4966 (6.7 / 24.4 / 25.1 % better than baseline / DNSMOS / human respectively). Figure 4 shows a visualization of the model predictions compared to DNSMOS and the MFCC model. Regarding layer fusion, we do not see a consistent improvement by applying early fusion to the two feature depths (weighted sum of inputs). The model attends to both inputs with weights -0.75/0.10 for layers 10/41 respectively.

6. CONCLUSION

In this paper, we have performed an analysis of the pre-trained XLS-R models for the task of speech quality assessment. We found that using specific layer activations results in improved performance compared to using the final hidden layer. Specifically, there are two local optima for feature depth selection around layers 10 and 41 (layers 5 and 21 for XLS-R 300M); however, the reason for the two distinct optima is still unclear. Finally, we showed that the proposed models[3] substantially outperform DNSMOS and have lower variance than human annotators.

[3]Models: `https://github.com/lcn-kul/xls-r-analysis-sqa`

7. REFERENCES

[1] V. Grancharov and W. Kleijn, "Speech Quality Assessment," in *Springer Handbook of Speech Processing*. Springer, 2008.

[2] A. R. Avila, H. Gamper, C. Reddy, R. Cutler, I. Tashev, and J. Gehrke, "Non-intrusive Speech Quality Assessment Using Neural Networks," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 631–635.

[3] Y. Leng, X. Tan, S. Zhao, F. Soong, X.-Y. Li, and T. Qin, "MBNET: MOS Prediction for Synthesized Speech with Mean-Bias Network," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 391–395.

[4] W.-C. Tseng, C.-y. Huang, W.-T. Kao, Y. Y. Lin, and H.-y. Lee, "Utilizing Self-Supervised Representations for MOS Prediction," in *Interspeech*, 2021, pp. 2781–2785.

[5] B. Cauchi, K. Siedenburg, J. F. Santos, T. H. Falk, S. Doclo, and S. Goetze, "Non-Intrusive Speech Quality Prediction Using Modulation Energies and LSTM-Network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 7, pp. 1151–1163, 2019.

[6] G. Yi, W. Xiao, Y. Xiao, B. Naderi, S. Möller, W. Wardah, G. Mittag, R. Culter, Z. Zhang, D. S. Williamson, F. Chen, F. Yang, and S. Shang, "ConferencingSpeech 2022 Challenge: Non-intrusive Objective Speech Quality Assessment (NISQA) Challenge for Online Conferencing Applications," in *Proc. Interspeech 2022*, 2022, pp. 3308–3312.

[7] 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.

[8] B. Tamm, H. Balabin, R. Vandenberghe, and H. Van hamme, "Pre-trained Speech Representations as Feature Extractors for Speech Quality Assessment in Online Conferencing Applications," in *Proc. Interspeech 2022*, 2022, pp. 4083–4087.

[9] G. Mittag, B. Naderi, A. Chehadi, and S. Möller, "NISQA: A Deep CNN-Self-Attention Model for Multidimensional Speech Quality Prediction with Crowdsourced Datasets," in *Interspeech*, 2021, pp. 2127–2131.

[10] C. Richey, M. A. Barrios, Z. Armstrong, C. Bartels, H. Franco, M. Graciarena, A. Lawson, M. K. Nandwana, A. Stauffer, J. van Hout, P. Gamble, J. Hetherly, C. Stephenson, and K. Ni, "Voices Obscured in Complex Environmental Settings (VOICES) corpus," in *Interspeech*, 2018, pp. 1566–1570.

[11] A. Stupakov, E. Hanusa, J. Bilmes, and D. Fox, "COSINE - A corpus of multi-party COnversational Speech In Noisy Environments," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2009, pp. 4153–4156.

[12] B. Naderi and R. Cutler, "An Open source Implementation of ITU-T Recommendation P.808 with Validation," in *Interspeech*, 2020, pp. 2862–2866.

[13] B. Series, "Method for the subjective assessment of intermediate quality level of audio systems," *International Telecommunication Union Radiocommunication Assembly*, 2014.

[14] Y.-Y. Yang, M. Hira, Z. Ni, A. Chourdia, A. Astafurov, C. Chen, C.-F. Yeh, C. Puhrsch, D. Pollack, D. Genzel, D. Greenberg, E. Z. Yang, J. Lian, J. Mahadeokar, J. Hwang, J. Chen, P. Goldsborough, P. Roy, S. Narenthiran, S. Watanabe, S. Chintala, V. Quenneville-Bélair, and Y. Shi, "TorchAudio: Building Blocks for Audio and Speech Processing," *arxiv preprint arXiv:2110.15018*, 2022.

[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. L. 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 (EMNLP): System Demonstrations*, 2020.

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

[17] Y. Wang, J. Li, H. Wang, Y. Qian, C. Wang, and Y. Wu, "Wav2vec-switch: Contrastive learning from original-noisy speech pairs for robust speech recognition," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7097–7101.

[18] Q.-S. Zhu, J. Zhang, Z.-Q. Zhang, M.-H. Wu, X. Fang, and L.-R. Dai, "A noise-robust self-supervised pre-training model based speech representation learning for automatic speech recognition," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 3174–3178.

[19] H. Wang, Y. Qian, X. Wang, Y. Wang, C. Wang, S. Liu, T. Yoshioka, J. Li, and D. Wang, "Improving noise robustness of contrastive speech representation learning with speech reconstruction," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6062–6066.

[20] C. K. Reddy, V. Gopal, and R. Cutler, "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *ICASSP 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6493–6497.