Appendix

1 NORESQA Architecture

NORESQA's architecture comprises of three key components: a *feature-extraction* block, a *temporal-learning* block and *task-specific* heads.

Feature-Extraction Block.

We use the Inception architecture in the feature extraction block. It consists of 4-block Inception modules: each module consisting of 64 convolutional filters - *24* 1x1 filters, *32* 3x3 filters, and *8* 5x5 filters. These filters are concatenated and finally passed through a 1x4 maxpool block, preserving the temporal dimension. This is repeated for each of the 4 inception blocks. The final output dimensions of the model is $B \times 64 \times T \times 2$, where B is the batch size, and T is the number of frames. ReLU is used as the activation function after each layer.

Temporal Learning Block.

We use temporal convolutional networks (TCNs) in the temporal learning block. The network consists of 4 temporal blocks. Each block consists of 2 convolutional layers with a kernel size of 1x3, with each layer consisting of 32, 64, 64 and 128 channels for each of the 4 blocks respectively. After the convolutional layers, each layer uses weight normalization which is a reparameterization trick that decouples the magnitude of a weight tensor from its direction. The outputs are then passed to ReLU as an activation function, and finally to dropout having a value of 0.2. The convolutional layers in each block consist of dilated convolutions with dilation factors of 2, 4, 8 and 16 for each of the 4 blocks. Use of dilated convolutions increases the effective history of our model. The initial weights of this network are chosen from the normal distribution $\mathcal{N}(0, 1e^{-2})$.

The parameters for both these blocks (feature-extraction and temporal-learning) are shared between the two inputs to our model. The embeddings for each input are concatenated (along the channel dimension), and are then passed next to the task-specific heads. The input to this model is $B \times T \times 128$, and the output is also $B \times T \times 128$, since TCNs can maintain the same length of the signal.

Task Specific Heads.

Each of the two tasks (*preference-task* amd *quantification-task*) each has a separate head. Their architecture is described next:

Preference Task: The output of the preference task is a frame-level prediction of which input is cleaner. This head consists of 3 convolutional layers, each consisting of 32, 8 and 2 channels respectively. The kernel size for each layer is 1x5, with each layer also having BatchNorm and dropout (0.2). The input to this model is $B \times T \times 256$ (after concatenating along the channel dimension of the two inputs) and the output is $B \times T \times 2$ with a framewise prediction of which input is cleaner.

Quantification Task: Refer to Sec 3.3 (main paper). The objective of the quantification task is to quantify the framewise quality difference between the two inputs. Here we formulate this as a classification problem, where we divide the whole range of SNR (Δsnr_{max}) and SI-SDR (Δsdr_{max}) into K equal intervals. The output of this head is a probability distribution over all K classes. Similar to the Preference task, this network also produces frame-level distributions. For both objectives (SNR and SI-SDR), we take $K = 40$. This head consists of 3 convolutional layers, each consisting of 64, 50 and 40 channels respectively. The kernel size for each layer is 1x5, with each layer also having BatchNorm and dropout (0.2). The input to this model is again $B \times T \times 256$ (after concatenating along the channel dimension of the two inputs) and the outputs are $B \times T \times 40$ for both SI-SDR and SNR for a framewise prediction of relative quality.

2 Experimental setup

Dataset. For the training and validation set, we choose the clean audio recordings from the DNS Challenge. The noise perturbations are sampled from the FSDK50 dataset that consists of over 51k audio samples encompassing around 100 hours of audio manually labelled using 200 classes drawn

from the Audioset ontology. As additional examples of distortions, we also use Clipping distortion, and Frequency masking as very common examples of distortions found in audio processing tasks. All data is divided into 90% train and 10% validation so that there is no overlap of train and validation data.

For the test set, we use the TIMIT dataset for the clean recordings. The noise perturbations are sampled from the ESC-50 dataset that consists of 2000 labeled recordings equally balanced between 50 classes (exactly 40 clips per class). We also use Gaussian noise and Mu law compression as two examples of unseen test distortions.

The training and testing pairs are created by adding the same type/category of noise to both recordings, but at two *different* noise levels to two *different* clean recordings. We design a simulation environment where we sample the clean recordings and noisy recordings from their respective datasets, and create the degraded recordings at differences levels of noise.

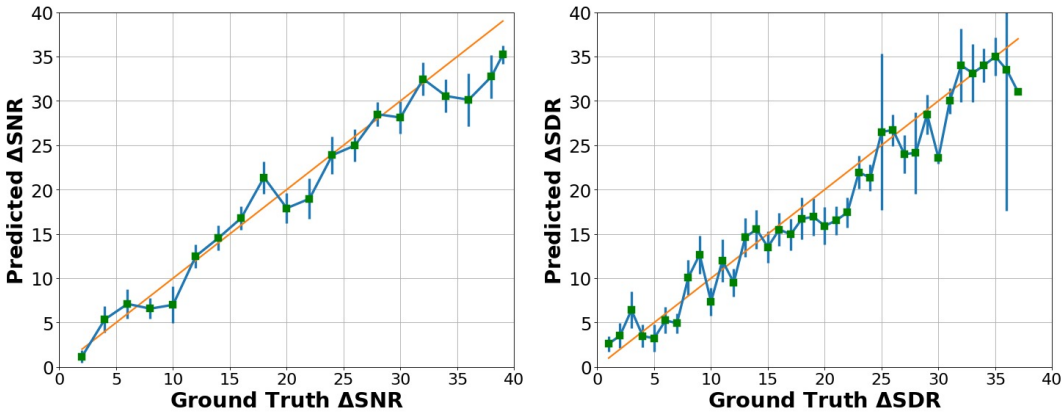
We also include reverberations in our audio recordings to make them more realistic and help the model generalize better. We sample impulse responses from the DNS Challenge dataset and convolve with the noisy recordings before input to the model. We use the "medium-room" and "small-room" RIRs, where the length and width of the room are sampled from 1 to 30m.

3 Objective Evaluations

3.1 In-variance to language

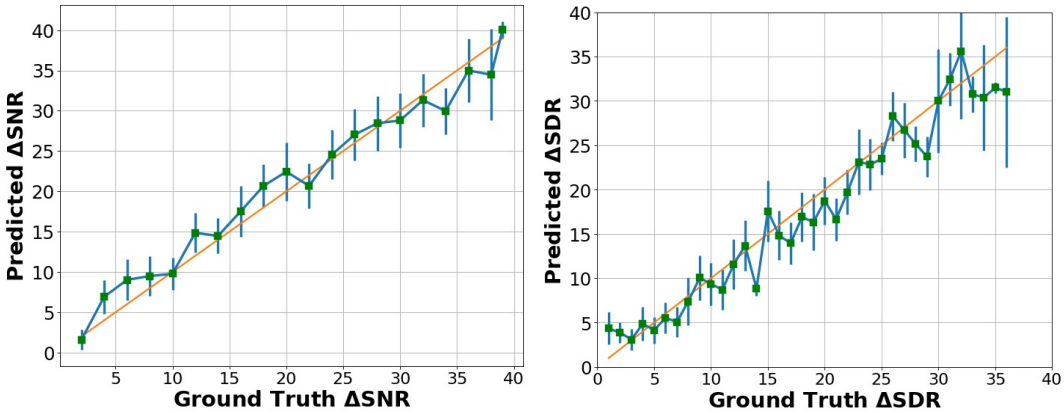
Fig 1 shows our metric's outputs with increasing SNR and SI-SDR difference for an unseen set of clean recordings that were randomly chosen from various languages including Mandarin, French - Arabic - Turkish - Spanish, Indian languages - Bengali, Gujarati, Hindi and Marathi, and noise recordings from ESC-50. Note that, the model itself was trained on English speech and we are testing with different languages. We see that the model performs fairly well for these unseen languages.

We studied robustness with respect to language by using non-matching references as well. In this case our test recording is always randomly selected from the English TIMIT dataset, and our reference recording is randomly chosen from any of the multi-langauge datasets (Fig 2). We observe that the general trend of the model is quite similar to Fig 2(a) and (b), (from the main paper, that showed the variation with English) which suggests that the model is invariant to language, and only considers acoustic differences to assess relative quality. Overall, these evaluations show that the trained model is not only robust to the language of test recording but can also compare quality of two speech signals with different languages.



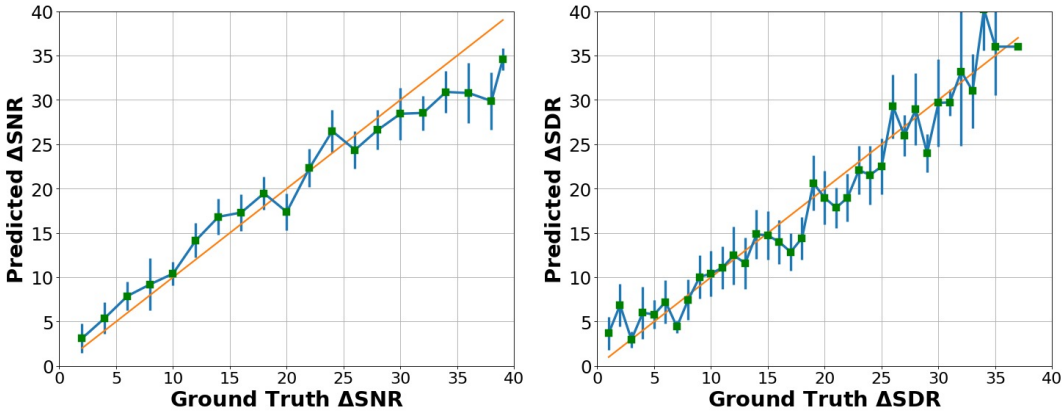
(a) (b)

Figure 1: Variation of our models output with increasing (a) SNR and (b) SI-SDR using *test and reference* recordings randomly chosen from Mandarin, French, Arabic, Turkish, Spanish, Bengali, Gujarati, Hindi and Marathi.



(a) (b)

Figure 2: Variation of our models output with increasing (a) SNR and (b) SI-SDR using *test* speech from English (TIMIT), and *reference* recordings from Mandarin, French, Arabic, Turkish, Spanish, Bengali, Gujarati, Hindi and Marathi.



(a) (b)

Figure 3: Variation of our models output with increasing (a) SNR and (b) SI-SDR using unseen clean *male* speech from the DAPS dataset.

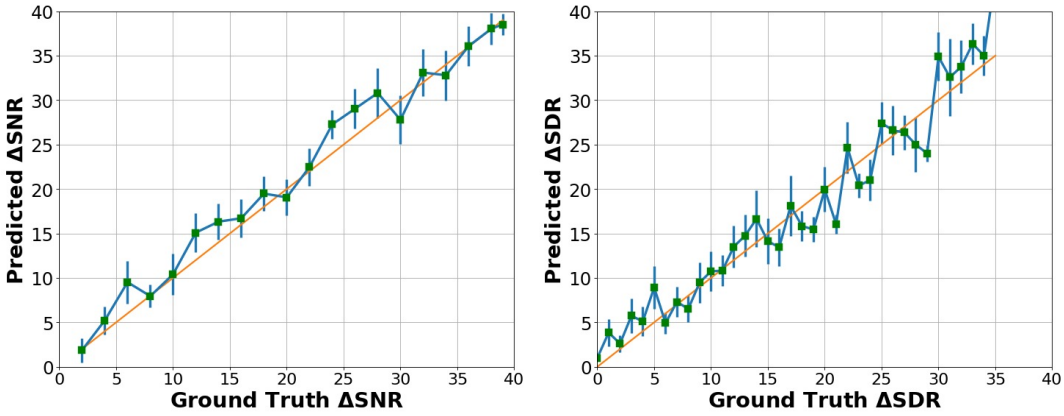
3.2 In-variance to gender

We now evaluate how NORESQA behaves with respect to speaker's gender. Once again we try to disentangle behaviour w.r.t to speaker's gender through two experiments. First, we see how the trained models behave for each gender, male and female. We test the models in male-only speaker condition (test and references are all male speeches) as well as in female-only speaker condition. Fig 3 shows the trends for male and 4 shows it for female. We note that the model works well in both cases.

Second, we evaluate how stable NORESQA is when the gender of the test recording *does not* match the reference, i.e., test recording is male speech, and reference recording is female speech and vice-versa (Fig 5). Once again, we observe that mis-matching speaker's gender in test and references does not adversely affect model's behaviour. Overall, based on these evaluation, we conclude that the model is invariant to gender of the speaker and is primarily learning quality related characteristics.

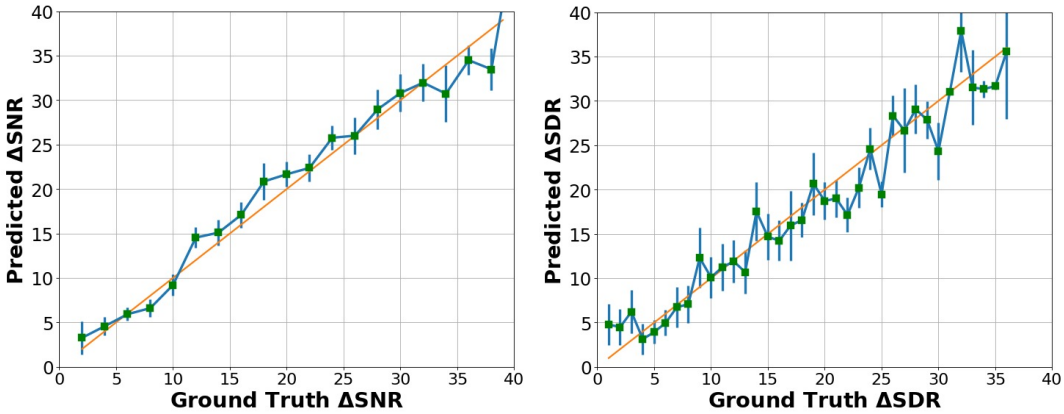
3.3 Commutativity: $\mathcal{N}(x_{test}, x_{ref}) = \mathcal{N}(x_{ref}, x_{test})$

We empirically evaluate the model to check if it satisfies the commutative property i.e., $\mathcal{N}(x_{test}, x_{ref}) = \mathcal{N}(x_{ref}, x_{test})$



(a) (b)

Figure 4: Variation of our models output with increasing (a) SNR and (b) SI-SDR using clean *female* speech from the DAPS dataset.



(a) (b)

Figure 5: Variation of our models output with increasing (a) SNR and (b) SI-SDR under mismatched gender conditions (i.e., test recording is male speech, and reference recording is female speech and vice-versa)

To evaluate the *preference-task*, we check the overall recording-wise predictions of both cases (i.e., $\mathcal{N}(x_{test}, x_{ref})$ and $\mathcal{N}(x_{ref}, x_{test})$ and see how well does the quality preference order swap when swapping the order of the inputs. We calculate accuracy by counting the number of times the preference order correctly gets swapped, and divide it by the total number of recordings. Our metric gets an accuracy of *98.3%* that empirically shows that it obeys the commutative property.

To evaluate the *quantification task* (Fig 6), we compute the predictions of both cases (i.e., d_1 = $\mathcal{N}(x_{test}, x_{ref})$ and d_2 = $\mathcal{N}(x_{ref}, x_{test})$). We then plot the distribution $abs(d_1 - d_2)$ for 1000 different pairs of recordings. We observe that the distribution is centered around 0dB which empirically suggests that it obeys the commutative property.

Overall, our model learns these two desirable properties without any specific training, and this suggests the usefulness of our framework for audio quality judgment.

3.4 Indiscernibility of Identicals: $\mathcal{N}(x_{test}, x_{test})$

Here, we show the output of our model when passed the same inputs (Refer to Sec 5.1 in the main paper). Ideally, the model should predict no quality difference when passed the same inputs. However, since our learning mechanism does not explicitly enforce the framework to have this property, so small errors are expected. For a fair comparison, we calculate the recording-level

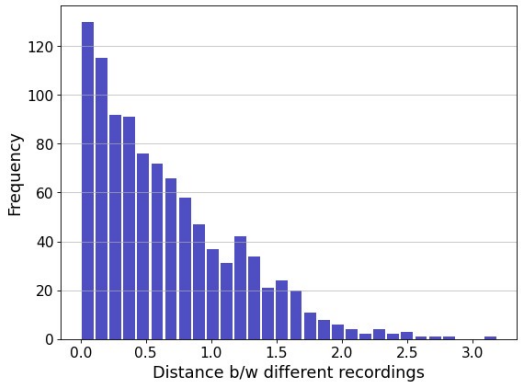
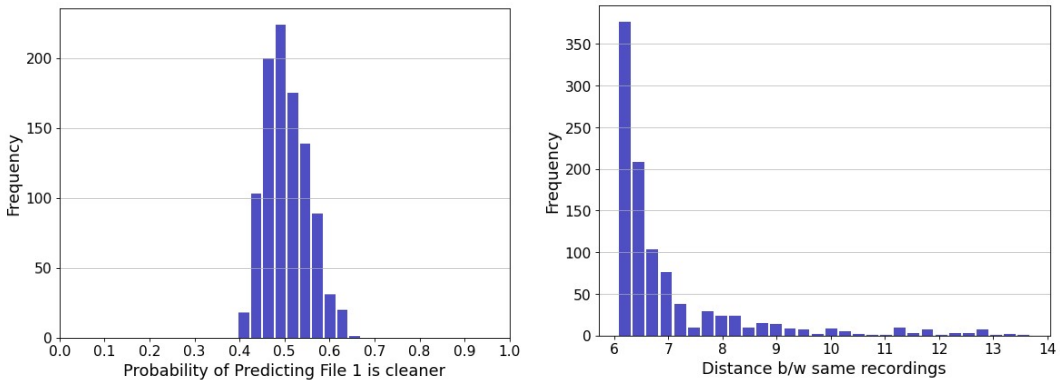


Figure 6: **Commutative Property:** Histogram plot of abs($\mathcal{N}(x_{test}, x_{ref})$ - $\mathcal{N}(x_{ref}, x_{test})$)

predictions from our model. Fig 7(a) shows the probability outputs on the *preference-task*. The outputs are close to 0.5 that suggests that the model is unable to confidently identify which output is cleaner. Fig 7(b) shows the output from the *quantification-task*. The distribution of the scores is centered around zero that suggests that the model correctly predicts that the two same inputs have similar quality levels, hence the near-zero quality difference.



(a) (b)

Figure 7: **Indiscernibility of Identicals**: (a) Evaluating our metric's performance on the *preference-task* and (b) *quantification-task* for the *same* inputs

3.5 Framewise detection

We also analyse the framewise performance of our model (Fig 8), using the same test bench that we created earlier, that consists of different recordings at various noise levels where each recording is 3 seconds. We concatenate different recordings together in *decreasing* levels of noise (i.e from high noise to low noise equally spaced from -10dB to +30dB) which becomes the test input to our model. Equivalently, the reference input to our model consists of a random concatenation of NMRs at +30dB. We observe that the framewise output of task 1 (preference-task) is almost *97%* accurate in the preference task of predicting which frame is cleaner between the two input recordings, which suggests that it learns this quite well. We also compare the framewise outputs from task 2 (quantitative-task) that predicts their relative quality difference. We find that the framewise predictions are not monotonic but still follow the general trend. This is expected, since we always optimize on a recording level, and not on a frame level. This findings suggest that our metric can detect and quantify which frames are degraded in quality between the inputs.

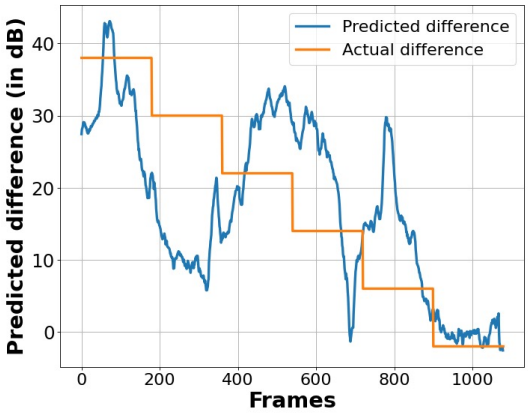


Figure 8: **Framewise Predictions:** Evaluating the framewise predictions of our model using a test recording that has decreasing levels of noise (from -10dB to +30dB) and the reference set consists of NMRs at +30dB.

4 Subjective Evaluation Datasets

Refer to Sec 5.2 in the main paper. We evaluate our framework subjectively on two tasks: (i) Correlation with MOS across 8 existing datasets, and (ii) 2AFC accuracy, where we show the performance of our metric on triplet comparison questions from 4 different datasets. MOS checks for aggregated ordering, scale and consistency, whereas 2AFC checks for exact ordering of similarity at per sample basis. Most of the datasets lie within the range of -15dB to 60dB SNR and -15dB to 25dB SI-SDR, which also roughly matches our chosen training intervals.

The following are the datasets that we used for evaluation:

1. *VoCo*: This dataset is based on comparing 6 different *word synthesis and insertion* algorithms. The MOS tests were asked to rate which algorithm could synthesize a new word such that it blends seamlessly in the context of the existing narration. This consists of non-sample aligned pairs of recordings.
2. *Dereverberation*: This dataset is based on evaluating improvements across 5 deep-learning based *speech enhancement* models including BLSTM, Wavenet, StarGAN-VC etc. MOS tests were done to evaluate which algorithm was rated the highest by subjects from Amazon Mechanical Turk (AMT). They obtained more than 100 ratings per condition.
3. *PEASS*: The dataset contains separated sources and specifically defined anchor signals to assess audio *source separation* performance across 4 metrics: *global quality*, *preservation of target source*, *suppression of other sources*, and *absence of additional artifacts*. Here, we only look at *global quality*. It consists of scores from 20 subjects over a set of 80 sounds.
4. *Voice Conversion (VC)*: c This objective of this dataset is to compare the performance on speaker (voice) conversion. It consists of two tasks: (i) parallel (*HUB*) and (ii) non-parallel data (*SPO*). Here we only consider *HUB*. The dataset consists of 4 source and 4 target speakers, where each speaker utters the same sentence set consisting of around 80 sentences.
5. *Noizeus*: The dataset was developed to encourage comparison of non-deep learning based *speech enhancement* algorithms, and consists of 30 IEEE sentences from 3 male and female speakers. The recordings are corrupted by 8 real-world noises, where the noises are taken from the AURORA database. The evaluation is done across 3 metrics: *SIG*-speech signal alone; *BAK*-background noise; and *OVRL*-overall quality. Here, we only look at *OVRL*.
6. *TCD-VoIP*: This dataset was developed to help assess degradations that can occur in VoIP (Voice over IP calls). IT contains speech samples with a range of common VoIP degradations background noise, echo, chop (packet loss), clipping etc., and the corresponding set of subjective scores from 24 listeners.
7. *HiFi-GAN*: This dataset is based on evaluating the improvement across 10 deep-learning based *speech enhancement* models including BLSTM, Wavenet, MetricGAN, SpecGAN etc. MOS tests were done to evaluate which algorithm was rated the highest by subjects from Amazon

Mechanical Turk (AMT), with each method receiving around 14k ratings. Further, it also consists of a 2AFC preference dataset consisting of around 1200 triplets, where each triplet received around 900 judgment ratings.

8. *FFTnet*: This dataset is based on evaluating the performance of 5 *speech synthesis* algorithms across 2 male and female speakers. It introduces artifacts specific to synthesis and are not sample-aligned due to phase change. Further, it also consists of a 2AFC preference dataset consisting of around 2050 triplets, where each triplet received around 480 ratings.
9. *Bandwidth Expansion*: This dataset consists of subjective tests for 3 different *bandwidth expansion* algorithms, aiming at increasing sample rate by filling in the missing high-frequency information. These audio samples consist of very subtle high-frequency differences. Further, it also consists of a 2AFC preference dataset consisting of around 1020 triplets, where each triplet received around 400 judgments.
10. *Simulated*: This dataset consists of 2AFC preference triplets, totalling to 1210. These are all based on adding *common-realistic degradations* like background noises, speech distortions like clipping and other miscellaneous types of degradation's like compression and EQ.

5 Ablations

In this section, we evaluate the influence of different components of our framework.

5.1 Relative vs Absolute Quantification Task

One key aspect of NORESQA is that it tries to model relative quality differences rather absolute quality measures in terms of SNR and SI-SDR. We discussed motivations and intuitive reasons for it in the main paper. We empirically study the relative vs absolute modeling of the quality measures here. We compare our framework to models that are trained to predict SNR and SI-SDR directly.

For the absolute quality prediction models, we consider two cases (i) [*Single Input Absolute Quantification*]: a single input model that takes a test recording and directly predicts the absolute quality measure , SNR and SI-SDR. It resembles the conventional formulation of a non-intrusive metric, and (ii) [*Two Input Absolute Quantification*]: a pairwise model that takes a test recording and a non-matching clean reference and predicts the absolute score. This is oriented towards our NORESQA framework but instead of learning relative differences it tries to learn absolute quality measures.

Results are shown in Table 1. We observe that our *relative* quality prediction model NORESQA performs the best by a considerable margin. This empirically corroborate our hypothesis that learning to model relative differences is much better than absolute measures. Their correlations with MOS turns out to be much better than "absolute methods". Moreover, we also observe that providing *any* (even a non-matched) clean reference to the model improves the performance over the *Single* case even for absolute quantification tasks. This demonstrates the inherent challenge of the conventional formulation of the non-intrusive metric that does not provide any reference. This highlights the usefulness of two features of our metric: (i) predicting relative quality scores, and (ii) providing non-matching references.

Name	**Type**	**VoCo**		**Dereverb**		**HiFi-GAN**		**FFTnet**	
		PC	**SC**	**PC**	**SC**	**PC**	**SC**	**PC**	**SC**
Absolute	**Sing. Inp.**	0.32	0.31	0.19	0.17	0.19	0.30	0.16	0.15
	Two Inp.	0.41±0.15	0.35±0.03	0.26±0.08	0.27±0.01	0.42±0.07	0.45±0.06	0.17±0.01	0.09±0.01
NORESQA		**0.85±0.01**	**0.68±0.03**	**0.66±0.02**	**0.67±0.02**	**0.68±0.01**	**0.78±0.01**	**0.33±0.01**	**0.44±0.01**

Table 1: **Ablations (1)**: Understanding the influence of predicting quality measures (single and pairwise) with our NORESQA using *Global-Fixed*$_{100}$ strategy. MOS Correlations: Spearman (SC), Pearson (PC). ↑ is better.

5.2 Multi-objective Learning of Quantification Task

To evaluate the impact of the multi-objective optimization (i.e., optimizing over both SI-SDR and SNR) for the quantification task on correlation to subjective ratings, we compare our trained

metric: (i) only using the SNR head; (ii) only using the SI-SDR head; and (iii) after combining both SNR and SI-SDR heads. Results are shown in Table 2. For simplicity, we only look at the *Unpaired-Global-Fixed*$_{100}$ strategy, which is evaluating a test recording using 100 clean NMRs randomly chosen from the DAPS dataset. We observe that using either head alone performs worse than using both together, which suggests that using a multi-objective optimization helps learn a better general representation.

Type	Name	VoCo		Dereverb		HiFi-GAN		FFTnet	
		PC	SC	PC	SC	PC	SC	PC	SC
NORESQA	**SNR only**	0.43	0.39	0.39	0.38	0.49	0.42	0.2	0.1
	SI-SDR only	0.6	0.48	0.48	0.49	0.54	0.65	0.25	0.28
	SNR and SI-SDR	**0.85**	**0.68**	**0.66**	**0.67**	**0.68**	**0.78**	**0.33**	**0.44**

Table 2: **Ablations (2)**: Understanding the influence of using (multi-objective) SI-SDR and SNR for Task 2 using *Global-Fixed*$_{100}$ strategy. MOS Correlations: Spearman (SC), Pearson (PC). ↑ is better.

5.3 Number of NMRs

Here we report the MOS correlation scores obtained when considering a set of 1, 10 and 100 NMRs for each test recording. This is shown for all 3 unpaired strategies described in Sec 5.2 (of the main paper) - *Unpaired, Unpaired-Local-Fixed, and Unpaired-Global-Fixed*. Results are shown in Table 3. We observe that averaging the scores over a larger set of NMRs reduces the standard deviation in the scores which leads to more stable predictions. We also observe that PC values are more consistent and stable over many iterations, and have a lower standard deviation than SC values. Since SC measures monotonic relationships, it can easily overfit to a complex function, leading to a higher standard deviation per iteration. However, since PC maps linear relationships, it is more stable. Finally, we observe no significant difference between the scores from *Unpaired-Local-fixed* and *Unpaired-Global-fixed* which suggests that our metric works equally well for scenarios where we take *any* random set of clean recordings as NMRs.

Type	Category	VoCo		Dereverb		HiFi-GAN		FFTnet	
		PC	SC	PC	SC	PC	SC	PC	SC
Unpaired	$\mathbf{NMR}_1$	0.76±0.1	0.27±0.2	0.57±0.03	0.62±0.04	0.63±0.01	0.70±0.02	0.43±0.10	0.45±0.11
	$\mathbf{NMR}_{10}$	0.87±0.01	0.43±0.07	0.64±0.01	0.73±0.03	0.63±0.01	0.70±0.01	0.45±0.03	0.48±0.06
	$\mathbf{NMR}_{100}$	0.88±0.01	0.41±0.06	0.63±0.01	0.75±0.02	0.63±0.01	0.71±0.01	0.46±0.01	0.51±0.02
+Local-Fixed	$\mathbf{NMR}_1$	0.65±0.23	0.40±0.23	0.53±0.10	0.57±0.15	0.56±0.08	0.64±0.08	0.38±0.10	0.31±0.13
	$\mathbf{NMR}_{10}$	0.79±0.1	0.44±0.2	0.61±0.05	0.69±0.05	0.61±0.02	0.67±0.03	**0.48±0.03**	0.50±0.04
	$\mathbf{NMR}_{100}$	**0.89±0.01**	0.44±0.06	0.63±0.01	**0.75±0.01**	0.61±0.01	0.73±0.01	0.46±0.01	**0.51±0.02**
+Global-Fixed	$\mathbf{NMR}_1$	0.79±0.20	0.54±0.20	0.44±0.16	0.41±0.19	0.56±0.08	0.63±0.10	0.29±0.10	0.36±0.12
	$\mathbf{NMR}_{10}$	0.84±0.05	0.63±0.08	0.62±0.08	0.62±0.09	0.63±0.01	0.71±0.02	0.33±0.03	0.41±0.07
	$\mathbf{NMR}_{100}$	0.85±0.01	**0.68±0.03**	**0.66±0.02**	0.67±0.02	**0.68±0.01**	**0.78±0.01**	0.33±0.01	0.44±0.02

Table 3: **Ablations (3)**: Understanding the effect of number of recordings used as non-matching references (NMR). MOS Correlations: Pearson (PC), Spearman (SC). Each cell shows the mean and standard deviation after 10 iterations. ↑ is better.

6 Speech enhancement

We use the VCTK dataset that consists of around 11,572 utterances for training and 824 files for validation. The dataset consists of 28 speakers equally split between male and female speakers, containing 10 unique background noise types across 4 different SNR conditions. Our denoising network is similar to Tan et al. and consists of a multi-layer convolutional encoder and decoder with U-Net skip connections, and a sequence modeling network applied on the encoders output. The input to the model are the real and imaginary components of the STFT of the signal, and the outputs are the complex ratio mask. The encoder (and decoder) consists of 5 gated convolutional layers with a filter size of 2×6, and use sigmoid activation for the gating mechanism. The sequence modeling network takes the encoders output and outputs a non-linear transformation of the same size. Since we design a *causal* model, the network consists of 2 uni-directional LSTM layers with 256 hidden units in each layer.

For evaluation, we use the audio clips from the VCTK test set and evaluate scores on that dataset. We evaluate the quality of enhanced speech using objective measures. We use: i) PESQ (from 0.5 to 4.5); (ii) Short-Time Objective Intelligibility (*STOI*) (from 0 to 100); (iii) Segmental Signal-to-Noise Ratio (*SNRseg*): average of SNR values of short segments (15 to 20ms) ; (iv) *CSIG*: MOS prediction of the signal distortion attending only to the speech signal (from 1 to 5); (v) *CBAK*: MOS prediction of the intrusiveness of background noise (from 1 to 5); (vi) *COVL*: MOS prediction of the overall effect (from 1 to 5). We compare the baseline approach with our model across the various paired-data constrained strategies.

As shown in Table 4 (main paper), SE models trained using NORESQA obtain higher objective scores than the baseline models across all three strategies. We observe that the difference between the scores from our model and the baseline keeps increasing as we get more paired data which shows the usefulness of our metric, especially for sparse labeled-data situations (e.g., low-resource languages) since our approach leverages unlimited unpaired data.

Most notable is the improvement in *STOI* that shows our metrics' utility as an optimization objective for pretraining. Given sparse-labeled data, the baseline model can fit the test set only as much. However, since our model can effectively leverage unlimited unpaired data during pretraining, given a model of sufficient capacity, it has the flexibility to learn a more complex mapping. This is precisely why we observe higher objective scores for quality and intelligibility than the baseline approaches which also highlights the usefulness of our framework.