SCOREQ: Speech Quality Assessment with Contrastive Regression

Alessandro Ragano
School of Computer Science
University College Dublin
Dublin, Ireland
alessandro.ragano@ucd.ie

Jan Skoglund
Google LLC
San Francisco, CA
jks@google.com

Andrew Hines
School of Computer Science
University College Dublin
Dublin, Ireland
andrew.hines@ucd.ie

Abstract

In this paper, we present SCOREQ, a novel approach for speech quality prediction. SCOREQ is a *triplet loss function for contrastive regression* that addresses the domain generalisation shortcoming exhibited by state of the art no-reference speech quality metrics. In the paper we: (i) illustrate the problem of L2 loss training failing at capturing the continuous nature of the mean opinion score (MOS) labels; (ii) demonstrate the lack of generalisation through a benchmarking evaluation across several speech domains; (iii) outline our approach and explore the impact of the architectural design decisions through incremental evaluation; (iv) evaluate the final model against state of the art models for a wide variety of data and domains. The results show that the lack of generalisation observed in state of the art speech quality metrics is addressed by SCOREQ. We conclude that using a triplet loss function for contrastive regression improves generalisation for speech quality prediction models but also has potential utility across a wide range of applications using regression-based predictive models.

1 Introduction

No-reference speech quality assessment has seen significant advancements in recent years, thanks to supervised learning [1, 14, 56, 50, 43, 6, 53, 3]. Supervised speech quality models learn to map input features (waveform domain, mel spectrograms) to Mean Opinion Score (MOS), a continuous target value derived by averaging individual listener ratings on a predefined Absolute Category Rating (ACR) scale (1=Bad, 2=Poor, 3=Fair, 4=Good, 5=Excellent).

While no-reference deep learning MOS predictors outperform traditional full-reference metrics (e.g, PESQ [54], POLQA [4], ViSQOL [8], CDPAM [40]), they struggle to generalise to unseen audio degradations. Addressing the domain mismatch in speech quality metrics is urgent given the fast growing research in generative speech e.g., neural speech coding [64], speech enhancement [36] and speech synthesis [49]. In Figure 5 we show the significant performance gap of no-reference speech quality metrics between in-domain and out-of-domain test sets.

The difficulty of no-reference speech quality models is the attempt to map high-dimensional data $\mathcal{R}^D$ to a monodimensional continuous space. Speech data include several entangled factors that

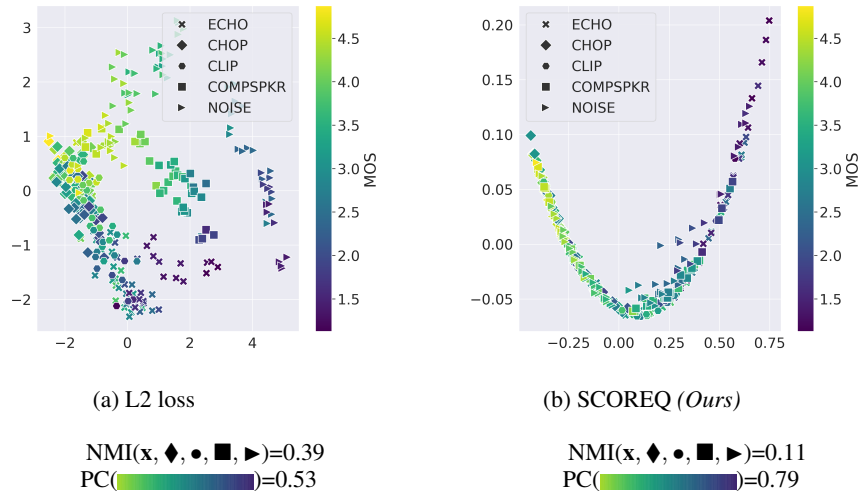|	
(a) L2 loss	(b) SCOREQ *(Ours)*
NMI(**x**, ♦, •, ■, ▶)=0.39	NMI(**x**, ♦, •, ■, ▶)=0.11
PC( )=0.53	PC( )=0.79

Figure 1: Embeddings of L2 loss (a) vs SCOREQ (b) on TCD VOIP data [17]. Color shows quality labels (MOS) while markers identify the degradations. We compute the Normalised Mutual Information (NMI) between k-Means clusters and degradation labels, as well as the Pearson's Correlation (PC) between embedding distance with respect to random clean speech and MOS targets. Higher NMI indicates representations are clustered based on degradations while higher PC means representations are ordered with respect to MOS targets. Results indicate that the L2 loss embeddings tend to capture degradation information (NMI=0.39, PC=0.53) while SCOREQ quality (NMI=0.11, PC=0.80). See Appendix I for more details.

are irrelevant to quality and they make it difficult for models to learn a low-dimensional space e.g., speaker identity, pitch, spoken content, phonetic information, degradation separability, etc. State-of-the-art methods that attempt to improve generalisation are based on either expanding the dataset, developing better architectures, or finetuning pre-trained self-supervised learning (SSL) methods. However, as we will show in this paper, domain mismatch is still a problem when tested on unseen degradations.

We observe that the majority of no-reference metrics are either trained or finetuned end-to-end by minimising the L2 loss between ground truth MOS and predictions. This approach is simple but it does not induce an ordered representation with respect to the regression targets in the feature space. The same would apply to L1 loss-based regression. Figure 1 (a) shows PCA-projected embeddings of the SSL wav2vec 2.0 [2] model finetuned with the L2 loss. It can be observed how the representation learned by the layer attached to the output layer is fragmented and tends to cluster degradations while MOS is only partially projected along one of the 2 PCA dimensions. Other studies such as RnC [66], have observed this problem in non-audio regression minimising the L1 loss. When attaching a linear projection layer to representations that are ordered with respect to targets, RnC is able to improve all the baselines trained with the L1 loss across a wide range of regression tasks.

Motivated by the issues above, we present Speech COntrastive REgression for Quality (SCOREQ) assessment. We propose the SCOREQ loss function which is based on contrasting triplets to learn a quality manifold as shown in Figure 1 (b). We adapt the batch-all triplet loss for person re-identification [13, 18] which addresses a classification task. Since MOS is a continuous label, the batch-all strategy is not trivial. By assuming that closeness is relative in regression, our loss function generates on-the-fly masks that identify valid triplets based on the MOS and are able to contrast all the possible triplet combinations in the batch based on regression targets. In addition, we investigate an alternative version that replaces the fixed margin with an adaptive margin initially proposed in [16]. This study suggests that supervised contrastive learning for regression performs significantly better than L2 minimisation for speech quality assessment. Our contributions are (i) We conduct a systematic evaluation to prove that domain mismatch i.e. a lack of robustness to both out-of-distribution (ODS) and out-of-domain (ODM) test sets, is a serious problem in speech

quality metrics; (ii) We propose a *triplet loss function for contrastive regression* that is based on on-the-fly triplet mining and adaptive margin; (iii) We conduct an extensive statistical analysis on the results showing that our contrastive loss improves significantly results on both ODS and ODM data using two architectures and 11 test sets from several speech domains; (iv) We propose 2 ready-to-use speech quality metrics based on the SCOREQ loss for natural and synthetic speech that both work in two modes: no-reference (NR SCOREQ) and non-matching reference (NMR SCOREQ) mode i.e., where quality is measured with unpaired clean speech.

2 Related Work

Speech Quality Assessment   The earliest no-reference speech quality methods are based on signal processing [38, 29] but showed poor generalisation. Data-driven no-reference metrics improved results. However, often they are trained with quality labels obtained from full-reference speech quality metrics [30, 6, 14] which already suffer from generalisation issues. Other MOS predictors were trained on MOS scores [53, 43, 1] and showed improvement across several databases. Recent methods are based on finetuning pre-trained self-supervised learning (SSL) models [59, 3, 22] such as HuBERT [20] or wav2vec 2.0 [2]. Although large improvement has been observed, e.g., in out-of-domain languages [3] we show in this paper that SSL finetuning still suffers from domain mismatch. Other metrics consider predicting speech quality without relying on the MOS. For example, SpeechLMScore [37] uses a speech-language model, and VQScore [15] uses vector quantization from clean speech only. The NORESQA framework [41, 39] and the NOMAD model [51] also propose solutions without MOS and they address the relative nature of quality assessment by using non-matching references. However, as we show in our experiments, both models still suffer from domain generalisation and overfitting of their corresponding training data similar to the no-reference models.

NOMAD vs. SCOREQ   NOMAD is trained by minimising the contrastive loss in a regression fashion, using the Neurogram Similarity Index Measure (NSIM) [19] as a proxy for quality. SCOREQ shares the same principle as NOMAD but enhances both training efficiency and performance. A significant drawback of NOMAD is the offline preparation of triplets, which is inefficient because not all triplets are contrasted with each other, leading to suboptimal results. Additionally, offline triplet requires defining *hard* and *easy* triplets either by measuring the embedded distance at each epoch or by using labels. NOMAD employs the latter method, using the NSIM space to establish harder triplets. However, the NSIM does not always align with human perception. In this paper, we replicate the NOMAD approach using MOS and demonstrate that it leads to suboptimal results, particularly in out-of-domain evaluations. To address this issue, SCOREQ is trained by learning all valid triplet combinations within the batch and removing the easy triplets. One more issue is that, because NSIM is a relative similarity criterion unlike MOS, triplets are composed only of the same clean reference. This approach is not optimal for learning a quality manifold where data are ordered based on quality, regardless of speech content. Therefore, we replace NSIM with MOS, allowing us to compare any combination of speech samples. Our proposed adapted triplet loss resolves all these issues simultaneously, yielding better results without NOMAD's overhead of preparing triplets offline. While NOMAD does not require human labels such as MOS for training, we use MOS labels to demonstrate that the SCOREQ loss improves generalisation performance and robustness over the L2 loss when MOS labels are available. Unlike NOMAD which targets unsupervised quality prediction, our objective is to present a supervised quality metric as well as to show how the SCOREQ loss improves generalisation and robustness.

Contrastive Learning   Contrastive learning is a widely adopted technique across various fields, such as computer vision, natural language processing, and audio processing, for self-supervised representation learning [7, 58, 32, 34]. In the speech domain, general-purpose speech representation models based on contrastive learning have emerged to tackle tasks like speech recognition and speaker classification with minimal supervision [2, 44, 42, 45, 24, 9]. This technique is typically framed as a classification task, where categorical labels are defined. Contrastive learning has also been applied to quality prediction, though it remains within a classification framework. For example, in CDPAM, content and acoustic representations are disentangled [40]. Beyond representation learning, some studies have employed contrastive learning for supervised classification tasks. SupCon [28] demonstrates contrastive learning for image classification, proving more robust than the cross-entropy loss in various scenarios [26, 35, 65].

However, there has been less focus on using contrastive learning for regression tasks. Unlike classification, contrastive regression approaches are primarily supervised, using label distance to determine feature similarity [67, 12, 61]. Rank-and-Contrast (RnC) [66] is a supervised contrastive learning framework designed for regression problems. Although it is supervised, RnC uses label distance as a pre-training step for feature extraction. Their work suggests that speech quality prediction could benefit from such a representation, given the continuous nature of MOS. Research has shown that contrastive learning pre-training often outperforms end-to-end supervised learning across several domains [27] and is more robust to data corruptions [68], making it a suitable approach for addressing domain mismatch issues [61]. Our SCOREQ loss contributes not only to the speech quality domain but also to the relatively unexplored area of contrastive learning for regression.

3 Contrastive Regression for MOS prediction

We consider a training dataset $\mathbb{S} = \{\boldsymbol{x}_i, y_i\}$ where $\boldsymbol{x}_i \in \mathbb{X}$ are raw representations of the i-th speech sample and $y_i \in \mathbb{Y}$ is the corresponding MOS label such that a distance $r(\boldsymbol{x}_j, \boldsymbol{x}_k) = |y_j - y_k|$ can be defined between any two samples in the set $\mathbb{S}$.

Let us consider an encoder $g : \mathbb{X} \mapsto \mathbb{H}$, that maps raw audio $\mathbb{X}$ to representations $\mathbb{H}$ and a projection head $f : \mathbb{H} \mapsto \mathbb{Z}$ that reduces the representations to a low-dimensional embedding space. Given 3 random speech files defined as anchor $\boldsymbol{x}_a$, positive $\boldsymbol{x}_p$, and negative $\boldsymbol{x}_n$ our goal is to learn a continuous-aware feature space where the distance between embeddings D is ordered based on the similarity criteria $r(\boldsymbol{x}_j, \boldsymbol{x}_k)$:

$$r(\boldsymbol{x}_a, \boldsymbol{x}_p) < r(\boldsymbol{x}_a, \boldsymbol{x}_n) \Rightarrow D(f(g(\boldsymbol{x}_a)), f(g(\boldsymbol{x}_p))) < D(f(g(\boldsymbol{x}_a)), f(g(\boldsymbol{x}_n))) \tag{1}$$

Since we define the similarity criteria $r(\cdot, \cdot)$ as the MOS distance, equation 1 becomes:

$$|y_a - y_p| < |y_a - y_n| \Rightarrow D(f(g(\boldsymbol{x}_a)), f(g(\boldsymbol{x}_p))) < D(f(g(\boldsymbol{x}_a)), f(g(\boldsymbol{x}_n))) \tag{2}$$

Triplet Loss for Regression To induce the defined order in the embedded space $\mathbb{Z}$, we propose a triplet loss inspired by the classification framework but adapted for regression. Given a training batch of size N, the classification triplet loss is defined as [55]:

$$\sum_i^N \max\left(0, \|f(g(\boldsymbol{x}_a^i)) - f(g(\boldsymbol{x}_p^i))\|_2^2 - \|f(g(\boldsymbol{x}_a^i)) - f(g(\boldsymbol{x}_n^i))\|_2^2 + m\right) \tag{3}$$

The objective of this loss function is to maximise the similarity in the embedded space between the positive sample $\boldsymbol{x}_p$ and the anchor $\boldsymbol{x}_a$, while minimising the similarity between the negative sample $\boldsymbol{x}_n$ and the anchor. The margin m is a constant that defines the distance threshold between the anchor and the negative sample, beyond which training stops. The classification triplet loss requires categorical labels such that $y^a = y^p$ and $y^a \neq y^n$. However, the triplet loss can be adapted for regression by considering that the concept of closeness is relative in regression problems. For example, if the MOS of the anchor is 2.0, this should be closer to MOS 1.5 (positive) than to MOS 3.0 (negative). This concept has been successfully applied by the NOMAD speech quality metric [51] and by the RnC framework with a different contrastive loss function [66]. Nonetheless, as discussed earlier, NOMAD does not leverage the triplet loss efficiently.

Batch All Strategy for Regression The SCOREQ loss adapts the batch-all (BA) strategy used in [13, 18] for regression. To achieve this, we compute masks for each batch using the available MOS labels, assuming that every sample can serve as an anchor with potential valid triplets. The masks are generated with the understanding that in regression problems, the concept of closeness is relative, unlike the absolute nature of classification settings. After finding the valid triplets, we also remove the easy triplets i.e., the ones where the positive is closer to the anchor than the negative in the embedding space.

Given a training batch of size N, for every possible distinct triplet combination in the batch (i, j, k), we compute a 3D mask $\mathcal{M}(i, j, k)$ such that:
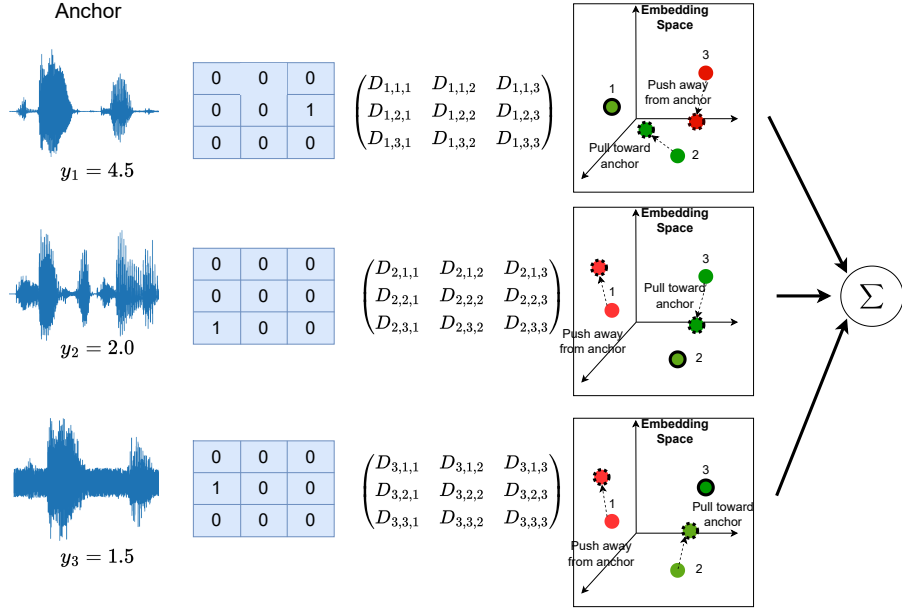
Figure 2: Example of the SCOREQ loss using 3 samples in a training batch with corresponding MOS labels 4.5, 2.0, and 1.5. The distance matrix entries are defined as $D_{i,j,k} = ||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_j))||_2 < ||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_k))||_2$. The intuition behind this contrastive loss for regression is shown in how the negative embeddings change in the anchor sample 1 where MOS is 4.5. We observe that the negative (sample 3 with MOS 1.5) will be further from the anchor with respect to sample 2. Indeed, because of the anchor 2 loss (where MOS is 2.0), sample 3 embeddings are pushed towards sample 2.
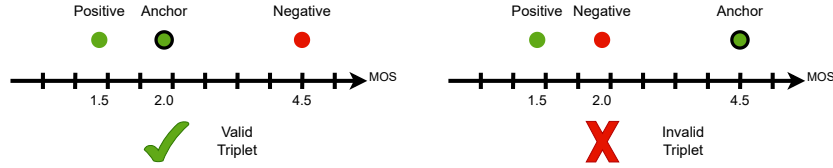

Figure 3: Example of how the mask $M(i, j, k)$ assigns 0 or 1. If the distance between the anchor and positive is lower than the distance between anchor and negative we consider it as a valid triplet. The inequality condition must also be verified i.e., $i \neq j \neq k$

$$\mathcal{M}(i,j,k) = \begin{cases} 1, & \text{if } (|y_i - y_j| < |y_i - y_k|) \wedge (i \neq j \neq k) \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

The SCOREQ loss for one batch is defined as:

$$\mathcal{L} = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} \mathcal{M}(i,j,k) \cdot (||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_j))||_2 - ||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_k))||_2 + m) \tag{5}$$

By multiplying by the mask $\mathcal{M}(i, j, k)$, we ensure that only triplets where the MOS distance between the anchor and the positive is smaller than the MOS distance between the anchor and the negative are used. Figure 2 illustrates an example of the SCOREQ loss function using a batch size of 3. This approach enables the model to compare every possible triplet in the batch in a relative manner, reflecting the human capacity to judge quality by using relative anchors for sensory judgement [31].
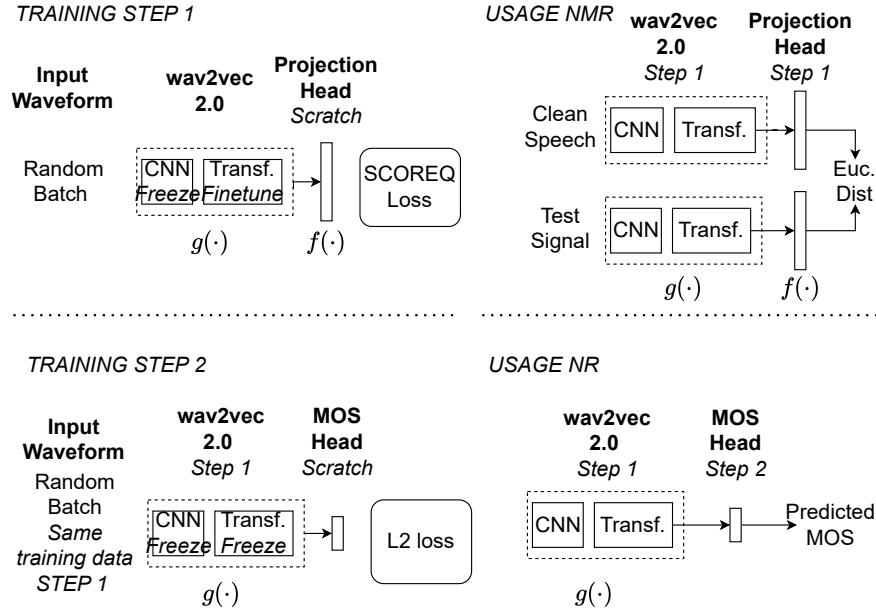
Figure 4: SCOREQ modes. No-Reference (NR) mode is trained in 2 steps. We first pre-train the encoder $g(\cdot)$ with the SCOREQ loss. Next, we learn a linear layer (MOS head) that predicts an interpretable numerical MOS.

The SCOREQ loss allows for training quality assessment models by sampling random data in the batch and fully utilising fast broadcasting computations. Figure 3 shows examples of valid and invalid triplets, indicating that the validity of training on a specific triplet combination depends on the relative distances concerning the anchor. The mask $\mathcal{M}(i, j, k)$ ensures that the training process ignores the invalid triplets.

Adaptive Margin Since we are addressing a regression problem, using a fixed margin m is not optimal. In classification problems, the margin is typically set to a constant to ensure that training stops once the distance in the embedding space between the anchor and the negative exceeds m. However, given our continuous label space, we can leverage the labels to model an adaptive triplet loss. An adaptive margin for the triplet loss was initially proposed in Ha and Blanz [16]. In our paper, we explore combining the adaptive margin from [16] with our batch all strategy loss for regression. Our experiments demonstrate that the significant performance improvement stems primarily from adopting the batch-all strategy over the offline triplet sampling used in NOMAD. The additional benefit of using an adaptive margin compared to a constant margin is minimal.

The idea behind the adaptive triplet loss is to replace the fixed margin m with a margin based on the MOS distance, as follows:

$$\mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \mathcal{M}(i, j, k) \cdot (||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_j))||_2 - ||f(g(\boldsymbol{x}_i)) - f(g(\boldsymbol{x}_k))||_2 + \tag{6}$$
$$(|y_i - y_j| - |y_i - y_k|)/(N - 1))$$

The adaptive margin is divided by $N - 1$ to normalise it within the range [0,1] [16]. The adaptive margin allows training to continue based on the MOS distance. This indicates that training will only stop if the distance between the negative sample and the anchor exceeds at least the normalised MOS distance. This is confirmed by computing the gradient (Appendix J).

SCOREQ Training The SCOREQ model has the flexibility to be used in non-reference (NR) mode to predict interpretable MOS ratings and in non-matching reference (NMR) mode to predict

a distance from unpaired clean speech. The latter has the advantage of being used as a perceptual training loss [51, 40] since SCOREQ is differentiable. For both approaches, we train an encoder $g : \mathbb{X} \mapsto \mathbb{H}$ that extracts representations $h_i = g(\boldsymbol{x}_i)$ and a projection head (embedding layer) that maps the representations to the space where the SCOREQ loss is applied: $f : \mathbb{H} \mapsto \mathbb{Z}$ producing d-dimensional embeddings $z_i = f(\boldsymbol{h}_i)$. For the NMR mode, we do not do further training. When evaluating we take random clean speech and measure the distance in the embedding space as done in NOMAD [51] as follows: $D(\boldsymbol{x}_{test}, \boldsymbol{x}_{nmr}) = ||f(g(\boldsymbol{x}_{test})) - f(g(\boldsymbol{x}_{nmr}))||_2$ where $\boldsymbol{x}_{test}$ is the degraded signal we want to predict quality for, $\boldsymbol{x}_{nmr}$ is a non-matching reference clean speech. For the NR-mode we freeze the encoder $g(\cdot)$ trained with SCOREQ and we train a linear layer that maps the embeddings $\boldsymbol{h}_i$ to MOS using the L2 loss. Since the encoder is frozen, the L2 loss does not affect the learned representation $\boldsymbol{h}_i$ which only depends on the SCOREQ loss. Our NR-mode model is compared with the L2 loss baseline which trains directly the same encoder $g(\cdot)$ using the L2 loss. In the baseline, the learned feature representation $\boldsymbol{h}_i$ depends only on the L2 loss minimisation. Notice that in the SCOREQ NR mode, we remove the projection head $f(\cdot)$ as done in other pre-trained contrastive learning approaches [40, 7]. This also makes sure that the L2 loss baseline architecture is the same as the NR mode SCOREQ architecture.

Architecture   We train the SCOREQ loss models with 2 architectures. The first is based on finetuning wav2vec 2.0 (w2v) [2] *BASE* model as done in NOMAD. The other architecture (w2vlight) is a smaller version of w2v where the transformer network is reduced from 12 to 4 layers and trained from scratch. We evaluate w2vlight to study whether without using the pre-trained transformer and a less powerful architecture we obtain the same effect with SCOREQ. More details in Appendix G. Similar to NOMAD, for SCOREQ the encoder output $g(\boldsymbol{x}_i)$ is the average of the last transformer layer along the time dimension. The embedding layer $f(\cdot)$ is a small network consisting of a ReLU followed by a fully connected layer projecting the encoder representations to a 256-dimensional feature vector. In all experiments, we only finetune the transformer of w2v, leaving the CNN frozen as done in other studies [51]. The best model is found with early stopping on a validation set. More details are in Appendix H.

4 Generalisation of Speech Quality Metrics

In this section, we demonstrate that certain no-reference speech quality models do not generalise well. Table 1 lists the metrics and their corresponding training sets used in our experiments. For each quality metric, we compare in-domain (IN), out-of-distribution (ODS), and out-of-domain (ODM) performances. We define ODS as test sets that are within the same domain as the training set but exhibit a distribution shift. For example, with simulated telephone degradations, a set consisting of real phone calls or a subset of degradations is out-of-distribution but within the same training domain. Similarly, a set in another language with the same degradations is considered a distribution shift but remains within the same domain. In contrast, we label test sets designed for other application domains as ODM. For example, speech synthesis is ODM with respect to telephone speech training data. Detailed information about training and test sets used is provided in the Appendix D and E respectively. We selected three domains for our experiments: (i) We train the NISQA metric on the NISQA TRAIN SIM dataset [43] that includes simulated degradations for telephone speech; (2) The no-reference (NR) versions of PESQ and SI-SDR provided in *TorchAudio-Squim* [30] that are trained on deep learning-based speech enhancement artifacts and noisy speech; (3) NORESQA-MOS [39] which was trained on the train partition of the VoiceMOS challenge dataset [22] that includes text-to-speech and voice conversion speech. We use the version provided in *TorchAudio-Squim* [30].

4.1   Performance Evaluation

In Table 2 and Figure 5, we report the results on the test datasets using the Pearson Correlation Coefficient (PC) to measure the linear relationship between predictions and MOS ground truth. Notice

Table 1: Metrics and corresponding training datasets

Metric	DB name	Dom. ID	Domain	BW (kHz)
NISQA	NISQA TRAIN SIM [43]	D1	Sim Degradations	48
NR-PESQ, NR-SI SDR	DNS Squim [30]	D2	DL Speech Enhancement	16
NORESQA-MOS	VoiceMOS Train [22]	D3	Speech Synthesis	16

7

Table 2: Domain mismatch evaluated with Pearson's correlation (PC). Each metric performs the best in its corresponding training domain. We indicate the domain shift between training and test with IN●, ODS◉, ODM○. Our proposed metrics show the best generalisation performance across most of the datasets. Note that results on the dataset DNS Squim are taken from [30] since the dataset is not publicly available.

|		NISQA	NR-PESQ	NR-SI SDR	NORESQA-M	NR-SCOREQ (*Ours*)
Dataset		D1	D2		D3	D1
NISQA TEST FOR	D1●D2◉D3○	0.91	0.79	0.74	0.68	**0.97**
NISQA TEST P501		0.94	0.88	0.81	0.70	**0.96**
DNS Squim	D2●	//	**0.96**	**0.99**	//	//
VoiceMOS Test 1	D1○D2○D3●	0.54	0.71	0.67	0.85	**0.86**
VoiceMOS Test 2		0.64	0.49	0.55	**0.91**	0.82
NOIZEUS	D1◉D2◉D3○	0.85	0.75	0.70	0.15	**0.91**
NISQA TEST LT		0.84	0.66	0.56	0.60	**0.86**
P23 EXP3		0.82	0.77	0.17	0.71	**0.94**
TCD VOIP		0.76	0.76	0.76	0.61	**0.85**
TENCENT		0.78	0.78	0.77	0.57	**0.86**
P23 EXP1	D1◉D2○D3○	0.76	0.70	0.82	0.40	**0.96**
TENCENT-Rev	D1○D2○D3○	0.40	0.36	0.32	0.36	**0.79**
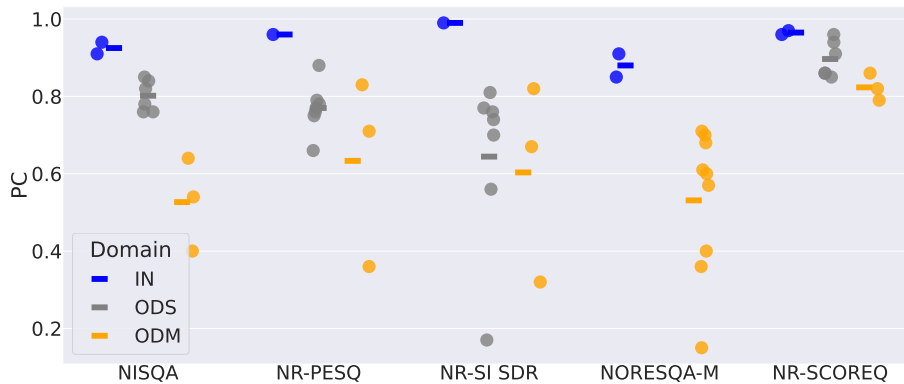
Figure 5: Domain mismatch. Each dot is a dataset, while horizontal lines represent the PC average for each domain shift (IN, ODS, ODM). PC values are the same of Table 2.

that each test set is ODS or ODM relative to the training set domains D1 (simulated telephone speech), D2 (deep learning-based speech enhancement), and D3 (speech synthesis). We also include one of our SCOREQ loss models in no-reference (NR-SCOREQ) mode. Our proposed method is outlined in the below sections. It is included in Table 2 to show the improvement over ODM and ODS domains.

Comprehensive results, including Spearman's rank correlation (SC) and root mean squared error (RMSE) are similar to PC performance (Appendix L). Our results confirm the hypothesis that speech quality metrics show difficulties in reaching satisfactory results in new domains (ODM) or even within the same domain but with a distribution shift (ODS).

5 Experiments

Experiments were conducted to compare L2 loss training against SCOREQ loss and evaluate performance across three scenarios: IN, ODS, and ODM. We assessed two domains: telephone speech degradations using the NISQA TRAIN SIM dataset, and speech synthesis using the VoiceMOS training partition.

5.1 Simulated Degradations: Models

Offline Model   This model replicates the approach used in the NOMAD speech quality metric but with MOS as a label. We prepared 46783 triplets from the NISQA TRAIN SIM dataset, which were chosen using the "hard" approach reported in the NOMAD paper that improved results. In this approach, for every sampled anchor we take the 10 most difficult triplets based on the MOS i.e., the ones where the MOS distance between anchor-positive difference and the anchor-negative difference is the shortest.

SCOREQ Models   We evaluate two variants: SCOREQ *Const* and SCOREQ *Adapt* consisting of using the SCOREQ loss with fixed margin (Eq. 5) and adaptive margin (Eq. 6) respectively.

L2 loss   The L2 loss model is the baseline and consists of finetuning the wav2vec 2.0 transformer by attaching an output layer that minimises the L2 loss between MOS targets and predictions.

NOMAD   We evaluate this model since is based on a similar principle and uses the same SCOREQ architecture. The original model only comes in NMR mode. However, we train an NR mode with the same SCOREQ approach. We freeze the NOMAD model and train an output layer to map NOMAD representations to an interpretable MOS scale. This is the only NR model where step 1 is trained without using MOS since we use the NOMAD pre-trained model that is unsupervised.

5.2 Simulated Degradations: Results

We evaluate the linearity (PC) and the average precision (RMSE) for NR metrics. For NMR metrics we still use linearity (PC) but also the Spearman's coefficient (SC) since NMR models do not predict MOS magnitude but they learn how to rank based on quality. Results for the RMSE are reported after performing a one-degree polynomial mapping between predictions and ground truth to adjust for the listening test bias in each dataset, as it is typically recommended in quality metric evaluation. NMR performance is calculated by extracting 50 random speech files from the test-clean partition of Librispeech [47]. These are used in all the NMR models found in this paper.

Table 3: Performance evaluation using PC($\uparrow$), RMSE($\downarrow$) for NR metrics and PC and SC($\uparrow$) for NMR metrics. Domain shift (IN, ODS, ODM) is labelled with respect to the training set NISQA TRAIN SIM.

		NR									NMR				
		L2 loss		NOMAD		Offline		SCOREQ *Ours*				NOMAD		SCOREQ *Ours*			
								Const		*Adapt*				*Const*		*Adapt*	
Test Set	Domain	PC	RMSE	PC	RMSE	PC	RMSE	PC	RMSE	PC	RMSE	PC	SC	PC	SC	PC	SC
NISQA TEST FOR	IN	0.96	0.23	0.89	0.38	0.89	0.38	0.96	0.22	**0.97**	**0.21**	0.73	0.60	**0.97**	**0.95**	0.96	**0.95**
NISQA TEST P501	IN	0.95	0.29	0.91	0.40	0.88	0.46	**0.96**	**0.29**	**0.96**	**0.28**	0.79	0.76	0.93	0.95	0.92	**0.96**
P23 EXP1	ODS	0.95	0.25	0.93	0.27	0.85	0.41	**0.96**	**0.21**	0.96	0.22	0.89	0.90	**0.97**	**0.97**	0.96	0.96
P23 EXP3	ODS	**0.95**	**0.23**	0.92	0.28	0.86	0.37	0.92	0.29	0.94	0.25	0.83	0.70	0.90	**0.81**	0.87	0.80
NOIZEUS	ODS	0.84	0.26	0.72	0.33	**0.92**	**0.19**	0.90	0.21	0.91	0.20	0.47	0.42	0.91	**0.91**	0.87	0.87
NISQA TEST LIVETALK	ODS	0.85	0.43	0.84	0.44	0.79	0.50	0.87	**0.40**	0.86	0.42	0.79	0.73	0.85	0.85	**0.88**	**0.88**
TCD VOIP	ODS	0.80	0.59	0.80	0.58	0.78	0.60	**0.85**	**0.51**	**0.85**	**0.51**	0.65	0.57	0.80	**0.66**	0.74	**0.66**
TENCENT	ODS	0.81	0.71	0.80	0.72	0.82	0.69	0.84	0.66	**0.86**	**0.62**	0.67	0.76	0.80	**0.86**	0.85	**0.86**
TENCENT-Rev	ODM	0.74	0.61	0.71	0.64	0.69	0.65	0.78	**0.57**	**0.79**	**0.56**	0.57	0.40	0.73	0.74	0.78	**0.76**
VoiceMOS Test 1	ODM	0.75	0.53	0.79	0.49	0.81	0.47	0.84	0.43	**0.86**	**0.41**	0.74	0.71	0.78	0.76	0.74	**0.77**
VoiceMOS Test 2	ODM	0.69	0.68	0.67	0.70	0.57	0.77	0.75	0.62	**0.82**	**0.53**	0.42	0.50	0.74	**0.76**	0.75	0.75

Table 3 illustrates the results for IN, ODS, and ODM scenarios. We observe that both SCOREQ *Const* and SCOREQ *Adapt* are the best for both some ODS data and all ODM data. On average, our proposed loss contributes to improving the performance of no-reference speech quality prediction. The *Offline* model does not perform as well as our, showing inferior results than the L2 loss. We also evaluated the *Offline* model encoder representations in NMR mode (Appendix M), showing that it does not capture quality as well as SCOREQ. In Table 8 (Appendix) we show that the same improvement trend is confirmed with the w2vlight architecture. This suggests that our loss function can help training more compact networks without a significant performance drop. We notice that NOMAD underperforms SCOREQ but it has been trained with NSIM labels [19] used as a proxy for MOS quality.

Table 4: Performance evaluation using PC(↑), RMSE(↓) for NR metrics and PC and SC(↑) for NMR metrics. Domain shift (IN, ODM) is labelled with respect to the training set VoiceMOS Train.

		NR						NMR					
		L2 loss		SCOREQ *Ours*				NORESQA-M		SCOREQ *Ours*			
				Const		*Adapt*				*Const*		*Adapt*	
Test Set	Domain	PC	RMSE	PC	RMSE	PC	RMSE	PC	SC	PC	SC	PC	SC
VoiceMOS Test 1	IN	0.90	0.36	0.90	0.35	**0.91**	**0.34**	0.85	**0.87**	0.87	0.86	0.83	0.80
VoiceMOS Test 2	IN	**0.98**	**0.18**	0.97	0.21	0.97	0.21	0.91	**0.90**	0.93	0.85	0.88	0.83
NISQA TEST FOR	ODM	0.82	0.47	0.85	0.44	**0.87**	**0.41**	0.68	0.58	0.85	0.78	0.84	**0.79**
NISQA TEST P501	ODM	0.86	0.50	**0.89**	**0.45**	**0.89**	0.45	0.70	0.81	0.86	0.87	0.84	**0.88**
P23 EXP1	ODM	0.92	0.31	**0.93**	**0.29**	0.92	0.30	0.40	0.35	0.84	0.91	0.93	**0.95**
P23 EXP3	ODM	0.85	0.38	0.88	**0.35**	0.87	0.36	0.71	0.48	0.83	0.69	0.90	**0.85**
NOIZEUS	ODM	0.59	0.39	0.69	**0.35**	0.67	0.36	0.15	0.15	0.48	0.46	**0.63**	**0.60**
NISQA TEST LIVETALK	ODM	0.81	0.47	0.83	0.46	0.84	**0.44**	0.60	0.51	0.78	0.74	**0.83**	**0.80**
TCD VOIP	ODM	0.83	0.54	0.87	**0.48**	0.86	0.49	0.61	0.71	0.81	**0.81**	0.77	0.61
TENCENT	ODM	0.78	**0.76**	0.76	0.78	0.77	0.78	0.57	0.59	**0.71**	0.78	0.78	0.79
TENCENT-Rev	ODM	0.35	0.85	0.43	0.82	0.44	**0.81**	0.36	0.20	0.48	**0.33**	**0.49**	0.38

5.3 Speech Synthesis: Results

The same approach is evaluated using the VoiceMOS training sets [22] to examine the generalisation from speech synthesis to telephone speech degradations. We report results from NORESQA-MOS which has been trained on the same sets. Again, we compare the SCOREQ loss models against the L2 loss. We do not evaluate the *Offline* model since it did not perform well in the simulated degradations domain. Results in Table 4 show that the SCOREQ representations improve results over the L2 loss. Also, the NMR SCOREQ models significantly improve ODM performances over NORESQA-MOS. No difference can be found for IN sets (Table 3).

5.4 Statistical Analysis and Encoder Representation Performance

We conduct a bootstrap procedure to understand whether the SCOREQ correlation improvement observed above is due to chance. Our results show that the SCOREQ models perform significantly better than the NR L2 loss in most of the ODM and ODS sets (Appendix N). We compare the encoder representations $g(\cdot)$ in NMR mode to understand which one explains quality information the best, showing that L2 loss encoders perform significantly worse than encoders trained with SCOREQ (Appendix M).

6 Conclusions

This paper presents SCOREQ, a contrastive regression loss to learn continuous representations for speech quality metrics. We have demonstrated that substituting the classic L2 loss with our proposed SCOREQ loss results in consistent performance improvements across 11 datasets and 2 speech domains. The SCOREQ loss can encourage a new research direction to train speech quality metrics. By learning ordered representations that follow the continuous nature of MOS, SCOREQ improves generalisation. In addition, we made available the SCOREQ metric in both no-reference mode and non-matching reference (NMR) mode for both simulated speech degradations and speech synthesis. The NMR models have been shown to be better than other NMR metrics. They can be used as a perceptual training loss and serve better applications where the reference is available (e.g., speech codecs) or quality wants to be measured relative. Finally, we believe that the SCOREQ loss can be applied to any application where models can be trained with regression-based targets.

7 Acknowledgements

References

[1] Anderson R Avila, Hannes Gamper, Chandan Reddy, Ross Cutler, Ivan Tashev, and Johannes Gehrke. Non-intrusive speech quality assessment using neural networks. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 631–635. IEEE, 2019.

[2] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[3] Helard Becerra, Alessandro Ragano, and Andrew Hines. Exploring the influence of fine-tuning data on wav2vec 2.0 model for blind speech quality prediction. In *Proc. Interspeech*, pages 4088–4092, 2022. doi: 10.21437/Interspeech.2022-10766.

[4] John G Beerends, Christian Schmidmer, Jens Berger, Matthias Obermann, Raphael Ullmann, Joachim Pomy, and Michael Keyhl. Perceptual objective listening quality assessment (POLQA), the third generation ITU-T standard for end-to-end speech quality measurement part I — temporal alignment. *Journal of the Audio Engineering Society*, 61(6):366–384, 2013.

[5] Denis Burnham, Dominique Estival, Steven Fazio, Jette Viethen, Felicity Cox, Robert Dale, Steve Cassidy, Julien Epps, Roberto Togneri, Michael Wagner, et al. Building an audio-visual corpus of Australian English: large corpus collection with an economical portable and replicable Black Box. In *Proc. Interspeech 2011*, 2011.

[6] Andrew A Catellier and Stephen D Voran. Wawenets: A no-reference convolutional waveform-based approach to estimating narrowband and wideband speech quality. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 331–335. IEEE, 2020.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[8] Michael Chinen, Felicia SC Lim, Jan Skoglund, Nikita Gureev, Feargus O'Gorman, and Andrew Hines. ViSQOL v3: An open source production ready objective speech and audio metric. In *2020 Twelfth international conference on quality of multimedia experience (QoMEX)*. IEEE, 2020.

[9] Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. w2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250. IEEE, 2021.

[10] Erica Cooper and Junichi Yamagishi. How do Voices from Past Speech Synthesis Challenges Compare Today? In *Proc. 11th ISCA Speech Synthesis Workshop (SSW 11)*, pages 183–188, 2021. doi: 10.21437/SSW.2021-32.

[11] Isin Demirsahin, Oddur Kjartansson, Alexander Gutkin, and Clara Rivera. Open-source multi-speaker corpora of the English accents in the British isles. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6532–6541, 2020.

[12] Mohamad Dhaini, Maxime Berar, Paul Honeine, and Antonin Van Exem. Contrastive learning for regression on hyperspectral data. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5080–5084. IEEE, 2024.

[13] Shengyong Ding, Liang Lin, Guangrun Wang, and Hongyang Chao. Deep feature learning with relative distance comparison for person re-identification. *Pattern Recognition*, 48(10): 2993–3003, 2015.

[14] Szu-Wei Fu, Yu Tsao, Hsin-Te Hwang, and Hsin-Min Wang. Quality-net: An end-to-end non-intrusive speech quality assessment model based on BLSTM. In *Proc. Interspeech*, pages 1873–1877, 2018. doi: 10.21437/Interspeech.2018-1802. URL https://doi.org/10.21437/Interspeech.2018-1802.

[15] Szu-Wei Fu, Kuo-Hsuan Hung, Yu Tsao, and Yu-Chiang Frank Wang. Self-supervised speech quality estimation and enhancement using only clean speech. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ale56Ya59q.

[16] Mai Lan Ha and Volker Blanz. Deep ranking with adaptive margin triplet loss. *arXiv preprint arXiv:2107.06187*, 2021.

[17] Naomi Harte, Eoin Gillen, and Andrew Hines. TCD-VoIP, a research database of degraded speech for assessing quality in VoIP applications. In *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2015.

[18] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[19] Andrew Hines and Naomi Harte. Speech intelligibility prediction using a neurogram similarity index measure. *Speech Communication*, 54(2):306–320, 2012.

[20] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021.

[21] Yi Hu and Philipos C Loizou. Subjective comparison of speech enhancement algorithms. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages I–I. IEEE, 2006.

[22] Wen Chin Huang, Erica Cooper, Yu Tsao, Hsin-Min Wang, Tomoki Toda, and Junichi Yamagishi. The VoiceMOS Challenge 2022. In *Proc. Interspeech 2022*, pages 4536–4540, 2022. doi: 10.21437/Interspeech.2022-970.

[23] ITU-T Recommendation ITU-TP. P.862.2: Wideband extension to recommendation p.862 for the assessment of wideband telephone networks and speech codecs,. Technical report, Tech. rep, 2016.

[24] Dongwei Jiang, Wubo Li, Miao Cao, Wei Zou, and Xiangang Li. Speech SimCLR: Combining Contrastive and Reconstruction Objective for Self-Supervised Speech Representation Learning. In *Proc. Interspeech 2021*, pages 1544–1548, 2021. doi: 10.21437/Interspeech.2021-391.

[25] Peter Kabal. Tsp speech database. *McGill University, Database Version*, 1(0):09–02, 2002.

[26] Bingyi Kang, Yu Li, Sa Xie, Zehuan Yuan, and Jiashi Feng. Exploring balanced feature spaces for representation learning. In *International Conference on Learning Representations*, 2020.

[27] Ananya Karthik, Mike Wu, Noah Goodman, and Alex Tamkin. Tradeoffs between contrastive and supervised learning: An empirical study. *arXiv preprint arXiv:2112.05340*, 2021.

[28] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

[29] Doh-Suk Kim and Ahmed Tarraf. ANIQUE+: A new American national standard for non-intrusive estimation of narrowband speech quality. *Bell Labs Technical Journal*, 12(1):221–236, 2007.

[30] Anurag Kumar, Ke Tan, Zhaoheng Ni, Pranay Manocha, Xiaohui Zhang, Ethan Henderson, and Buye Xu. Torchaudio-squim: Reference-less speech quality and intelligibility measures in torchaudio. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[31] Harry T Lawless and Hildegarde Heymann. *Sensory evaluation of food: principles and practices*, volume 2. Springer, 2010.

[32] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[33] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey. Sdr–half-baked or well done? In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630. IEEE, 2019.

[34] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=KmykpuSrjcq.

[35] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 316–325, 2022.

[36] Yen-Ju Lu, Zhong-Qiu Wang, Shinji Watanabe, Alexander Richard, Cheng Yu, and Yu Tsao. Conditional diffusion probabilistic model for speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7402–7406. IEEE, 2022.

[37] Soumi Maiti, Yifan Peng, Takaaki Saeki, and Shinji Watanabe. SpeechLMScore: Evaluating Speech Generation Using Speech Language Model. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE*, 2023. doi: 10.1109/ICASSP49357.2023. 10095710.

[38] Ludovic Malfait, Jens Berger, and Martin Kastner. P. 563—The ITU-T standard for single-ended speech quality assessment. *IEEE Transactions on Audio, Speech, and Language Processing*, 14 (6):1924–1934, 2006.

[39] Pranay Manocha and Anurag Kumar. Speech Quality Assessment through MOS using Non-Matching References. In *Proc. Interspeech 2022*, pages 654–658, 2022. doi: 10.21437/ Interspeech.2022-407.

[40] Pranay Manocha, Zeyu Jin, Richard Zhang, and Adam Finkelstein. CDPAM: Contrastive learning for perceptual audio similarity. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 196–200. IEEE, 2021.

[41] Pranay Manocha, Buye Xu, and Anurag Kumar. NORESQA: A framework for speech quality assessment using non-matching references. *Advances in Neural Information Processing Systems*, 34:22363–22378, 2021.

[42] Benjamin Milde and Chris Biemann. Unspeech: Unsupervised Speech Context Embeddings. In *Proc. Interspeech 2018*, pages 2693–2697, 2018. doi: 10.21437/Interspeech.2018-2194.

[43] Gabriel Mittag, Babak Naderi, Assmaa Chehadi, and Sebastian Möller. NISQA: A Deep CNN-Self-Attention Model for Multidimensional Speech Quality Prediction with Crowdsourced Datasets. In *Proc. Interspeech 2021*, pages 2127–2131, 2021. doi: 10.21437/Interspeech. 2021-299.

[44] Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, et al. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1179–1210, 2022.

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

[46] M Ott. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*, 2019.

[47] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[48] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[49] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Gradtts: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, pages 8599–8608. PMLR, 2021.

[50] Alessandro Ragano, Emmanouil Benetos, and Andrew Hines. More for less: Non-intrusive speech quality assessment with limited annotations. In *2021 13th International Conference on Quality of Multimedia Experience (QoMEX)*, pages 103–108. IEEE, 2021.

[51] Alessandro Ragano, Jan Skoglund, and Andrew Hines. NOMAD: Unsupervised Learning of Perceptual Embeddings For Speech Enhancement and Non-Matching Reference Audio Quality Assessment. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1011–1015. IEEE, 2024.

[52] Chandan K.A. Reddy, Vishak Gopal, Ross Cutler, Ebrahim Beyrami, Roger Cheng, Harishchandra Dubey, Sergiy Matusevych, Robert Aichner, Ashkan Aazami, Sebastian Braun, Puneet

Rana, Sriram Srinivasan, and Johannes Gehrke. The INTERSPEECH 2020 Deep Noise Suppression Challenge: Datasets, Subjective Testing Framework, and Challenge Results. In *Proc. Interspeech 2020*, pages 2492–2496, 2020. doi: 10.21437/Interspeech.2020-3038.

[53] Chandan KA Reddy, Vishak Gopal, and Ross Cutler. DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE, 2021.

[54] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *IEEE international conference on acoustics, speech, and signal processing. Proceedings*, volume 2, pages 749–752. IEEE, 2001.

[55] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[56] Joan Serrà, Jordi Pons, and Santiago Pascual. SESQA: semi-supervised learning for speech quality assessment. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 381–385. IEEE, 2021.

[57] Ke Tan and DeLiang Wang. Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:380–390, 2019.

[58] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020.

[59] Wei-Cheng Tseng, Chien yu Huang, Wei-Tsung Kao, Yist Y. Lin, and Hung yi Lee. Utilizing Self-Supervised Representations for MOS Prediction. In *Proc. Interspeech 2021*, pages 2781–2785, 2021. doi: 10.21437/Interspeech.2021-2013.

[60] International Telecommunication Union. ITU-T P. Supplement 23 coded-speech database, 1998.

[61] Yaoming Wang, Yangzhou Jiang, Jin Li, Bingbing Ni, Wenrui Dai, Chenglin Li, Hongkai Xiong, and Teng Li. Contrastive regression for domain adaptation on gaze estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19376–19385, 2022.

[62] Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Artyom Astafurov, Caroline Chen, Christian Puhrsch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z Yang, et al. Torchaudio: Building blocks for audio and speech processing. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6982–6986. IEEE, 2022.

[63] Gaoxiong Yi, Wei Xiao, Yiming Xiao, Babak Naderi, Sebastian Möller, Wafaa Wardah, Gabriel Mittag, Ross Culter, Zhuohuang Zhang, Donald S. Williamson, Fei Chen, Fuzheng Yang, and Shidong Shang. ConferencingSpeech 2022 Challenge: Non-intrusive Objective Speech Quality Assessment (NISQA) Challenge for Online Conferencing Applications. In *Proc. Interspeech 2022*, pages 3308–3312, 2022. doi: 10.21437/Interspeech.2022-10597.

[64] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021.

[65] Zhiyuan Zeng, Keqing He, Yuanmeng Yan, Zijun Liu, Yanan Wu, Hong Xu, Huixing Jiang, and Weiran Xu. Modeling discriminative representations for out-of-domain detection with supervised contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 870–878, 2021.

[66] Kaiwen Zha, Peng Cao, Jeany Son, Yuzhe Yang, and Dina Katabi. Rank-N-Contrast: Learning Continuous Representations for Regression. *Advances in Neural Information Processing Systems*, 36, 2024.

[67] Shihao Zhang, Linlin Yang, Michael Bi Mi, Xiaoxu Zheng, and Angela Yao. Improving deep regression with ordinal entropy. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[68] Y Zhong, H Tang, J Chen, J Peng, and Y-X Wang. Is self-supervised learning more robust than supervised learning? In *Proc ICML Workshop on Pre-training*, 2022.

APPENDIX

A   Software

Every model is trained on an Nvidia Tesla V100 GPU. We trained our models in PyTorch [48] and Torchaudio [62]. The wav2vec 2.0 model is taken from the fairseq toolkit [46]. The repository associated with this paper can be found here `https://github.com/alessandroragano/scoreq`.

B   Limitations

In this paper and its appendix, we present an exhaustive list of experiments. However, it is crucial to address certain limitations that pave the way for future research, especially in understanding the scope of our proposed SCOREQ loss.

Paper Scope   We believe the SCOREQ loss holds potential for various regression tasks. This belief is underpinned by the motivation to address generalisation issues in speech quality assessment. Despite this, domain mismatch remains a challenge across numerous applications. A significant limitation of our study is that we did not evaluate the loss outside the speech domain, as this was beyond the scope of our current investigation.

Execution Time   We computed both L2 and SCOREQ loss using the same batch of data, loading 4 waveforms and measuring the computational time on the GPU for the entire operation required for one batch, including time to load waveforms, forward pass, loss calculation, backward pass, and zero gradients. On average, we found that the L2 loss takes 0.65 seconds, while the SCOREQ loss takes 0.75 seconds using our GPUs. This slight increase is expected since the batch-all strategy needs to compute distances between all triplet combinations, including invalid triplets which are discarded with the 3D boolean mask.

Cross-domain between encoder and target   We have assessed two speech domains: simulated degradations on natural speech and synthetic speech. However, one scenario we have not explored is the performance of the model when the encoder is trained on one domain (step 1 in NR SCOREQ training) and the MOS mapping is trained on a different domain (step 2 in NR SCOREQ training).

C   Impact

Assessing speech quality is critically important, as it helps avoid the expensive and time-consuming process of conducting listening tests. Given the rapid advancements in generative AI research within the speech domain—including text-to-speech, voice conversion, and neural speech coding—it is increasingly challenging to apply quality metrics that were developed for other domains such as old speech codecs. Understanding domain mismatches and working towards a generalisable speech quality metric could fundamentally assist researchers in generative AI. Moreover, contrastive regression remains a relatively unexplored approach. Through this paper, we aim to contribute to the development of improved regression solutions.

D   Training and Validation Sets

NISQA TRAIN SIM   The NISQA TRAIN SIM dataset [43] is part of the NISQA Corpus[1] and includes several degradations for telephone speech combined and isolated. Additive white Gaussian noise, signal correlated MNRU noise, randomly sampled noise clips taken from the DNS-Challenge dataset, lowpass, highpass, and bandpass filters with random cutoff frequencies, amplitude clipping, speech level changes, codecs in various bitrate modes, packet-loss conditions with random and bursty patterns. The dataset is made of 10000 speech samples, each degraded with a different condition. Source samples are in English and taken from the Librivox clips of the DNS challenge [52], the TSP database [25], the UK and Ireland English dialect data [11], and AusTalk [5] which includes

---

[1]NISQA Corpus: `https://github.com/gabrielmittag/NISQA/wiki/NISQA-Corpus`

interviews from Australian English. Overall, the dataset is composed of 2322 speakers. The sampling rate is 48000 kHz and MOS labels are obtained from 5 listeners on average.

NISQA VAL SIM   The NISQA VAL SIM dataset [43] is a partition made with the same conditions and source samples of NISQA TRAIN SIM. This dataset consists of 2500 speech samples with 938 different speakers. We use this dataset for early stopping in all the experiments and to find the best combination of learning rate and batch size of the L2 loss model.

DNS Squim   This dataset was used to train the NR-PESQ and NR-SI SDR models that are available in *TorchAudio-Squim* [30]. The dataset is made of deep learning-based speech enhancement models (GCRN architecture [57]) and noisy speech trained on the DNS challenge dataset [52]. Training and test partitions are made of 364500 and 22800 speech samples respectively. This dataset is labelled with full-reference metric scores from PESQ wideband [23] and SI SDR [33].

VoiceMOS Train   The training partition of the VoiceMOS challenge [22] includes several text-to-speech and voice conversion samples from the Blizzard and the Voice Conversion challenges [10]. The VoiceMOS challenge dataset is labelled with 8 listeners for each clip. The training partition includes 175 synthesis systems. We include this dataset since it has been used to train the NORESQA-MOS [39] metric available in *TorchAudio-Squim*. The challenge includes two training partitions, main and OOD tracks. We combine both together to train SCOREQ in the speech synthesis domain.

E   Test Sets

In all experiments, we use 11 test sets labelled with MOS obtained from several speech domains and languages. Details for each dataset are shown in Table 5.

Table 5: Test datasets

DB name	Domain	Cond	Samples	Lang	BW(kHz)
NISQA TEST P501 [43]	Sim Degradations	60	240	en	48
NISQA TEST FOR [43]	Sim Degradations	60	240	en	48
NISQA TEST LIVETALK [43]	Real Phone Calls	58	232	de	48
P23 EXP 1 [60]	Codecs	50	200	en	16
P23 EXP 3 [60]	Codec, Bursty patterns, Noise	50	200	en	16
TCD VOIP [17]	Noise, Chop, Clip, Comp. Speakers, Echo	80	320	en	48
NOIZEUS [21]	Sig. Proc. Speech Enhancement	96	1536	en	8
TENCENT-Rev [63]	Real-world Reverberation	3197	3197	cn	16
TENCENT [63]	Sim Degradations	8336	8366	cn	16
VoiceMOS Test 1 [22]	Text-to-speech, voice conversion	187	1066	en	16
VoiceMOS Test 2 [22]	Text-to-speech, voice conversion	24	540	en,jp,cn	16

All the test sets are used in their original version except for TCD VOIP where we remove the 4 MNRU conditions since they are anchors and also included in the NISQA TRAIN SIM dataset. By doing that we test a more difficult scenario.

16

F Reproducing State of the Art Quality Metric Performances

NISQA We train NISQA with the default parameters, including fullband mode (48000 kHz), and we use the NISQA early stopping approach with patience of 20 epochs on the validation set NISQA VAL SIM to find the best model. Depending on the bandwidth of the test set we only change the parameter *ms_fmax* that determines the maximum frequency to use to calculate the mel spectrograms. See Table 6 for details. We do not resample the test files since the NISQA architecture adapts to any

Table 6: NISQA *ms_fmax* values chosen based on the input bandwidth.

Bandwidth Hz	*ms_fmax* Hz
48000	20000
16000	8000
8000	4000

input sampling rate. Notice that the NISQA metric provided in the NISQA repo has been trained on multiple datasets and in a multi-task fashion predicting several quality dimensions beyond MOS. In our experiments, we are only interested in examining how each metric performs when trained in one domain only, so that we can run controlled experiments. This explains why the results reported in the NISQA paper [43] are different from ours.

***TorchAudio-Squim* Results** For the three metrics NR-PESQ, NR-SI SDR, and NORESQA-MOS we follow the tutorial provided in the Torch documentation[2]. To predict NORESQA-MOS we extract 50 random clean speech samples from the test-clean partition of the Librispeech dataset [47]. All the test files are resampled to 16 kHz which is the expected sampling rate from all 3 metrics.

G SCOREQ Architectures

wav2vec 2.0 SCOREQ architecture is based on finetuning the transformer of the self-supervised learning (SSL) wav2vec 2.0 model. We use the *BASE* version. This model has been pre-trained to learn general-purpose representations using a contrastive learning approach. The architecture is made of 7 1D convolutional neural network (CNN) layers that encode audio frames of 25 ms into 400-dimensional feature vectors. Each frame-wise CNN output is fed into a transformer made of 12 layers that outputs T 768-dimensional context vectors. Positional encoding for the transformer is learned with a convolutional neural network. We take the average of the T frames and attach the output of the average, which is one vector of dimension 768 to an embedding layer that is made of a ReLU followed by a linear layer of dimension 256. This represents the dimension of the final embeddings. This approach was used by the NOMAD metric [51].

w2vlight We train the same approach with a smaller architecture. The w2vlight is built by reducing the number of the transformer layers from 12 to 4 and finetuning the transformer from scratch. We keep the same dimensionality and the same positional encoding approach of the original wav2vec 2.0 architecture as well as the number of neurons of the feedforward neural network that composes each transformer layer.

H Training Details

SCOREQ loss Training is done by trimming all the input files to 4 seconds which is enough to capture quality and avoids using large memory. All the results on the test data are computed without trimming, using the original length for each file. We finetune the transformer with a learning rate set to 0.00001, while the embedding layer trained from scratch uses a learning rate of 0.001. We use the Adam optimizer with default PyTorch settings except for the learning rate. The batch size is set to 128 in all SCOREQ experiments. Both learning rates decay exponentially with a decay factor of 0.99 every 10 epochs without improvement. In all the SCOREQ experiments, training is stopped if the Spearman correlation coefficient (SC) on the NISQA VAL SIM dataset does not improve for more

---

[2]*TorchAudio-Squim*

than 100 epochs. The best model is taken as the one with the highest SC. SC is calculated in NMR mode by extracting 200 random clean speech files from the dev-clean partition of the Librispeech dataset [47].

L2 loss   The models with the L2 loss are trained with the same architectural configurations of the SCOREQ loss models. The only difference is in the output layer. Instead of the embedding layer, the average of the transformer output is attached to a linear layer with one output neuron to predict MOS. Input files are also trimmed to 4 seconds to have a fair comparison while the original waveform length is used in all the test sets. Unlike SCOREQ, for the L2 loss we perform a gridsearch using several combinations of learning rate and batch size to select the best model on the validation set NISQA VAL SIM. This is done to avoid any potential result difference only due to different hyperparameters. In all the experiments, training is stopped if the loss function on the NISQA VAL SIM dataset does not improve for more than 100 epochs. The best model is taken as the one with the lowest L2 validation loss. We did some informal experiments by stopping with the best SC (as in SCOREQ) but we did not notice any significant difference in the results. Notice that even step 2 of NR-SCOREQ models has been validated using the L2 loss as a validation metric and not with the SC.

Table 7: Hyperparameter search for the L2 loss model.

Batch Size	Learning Rate	Validation Loss
32	0.1	0.2970
64	0.1	0.2865
128	0.1	0.2913
32	0.01	0.2505
64	0.01	**0.2466**
128	0.01	0.2730

Table 7 shows that the best configuration is batch size 64 and learning rate 0.01. We use the Adam optimizer with default PyTorch settings except for the learning rate. Notice that these parameters have been found using the original wav2vec 2.0 architecture but we also used them for the w2vlight. We observe that our proposed models based on the SCOREQ loss are not trained with an equivalent preliminary gridsearch. This means that better results could be found for SCOREQ.

I   Embedding Visualisation - Figure 1

Figure 1 shows the embeddings of the L2 loss model against our proposed SCOREQ loss. We use the models trained on NISQA TRAIN SIM and compute embeddings on TCD VOIP [17]. Embedding vectors are projected onto 2D space using PCA. We project the encoder $g(\cdot)$ (i.e., the last transformer layer) of the L2 loss model trained on the original wav2vec 2.0 architecture. This is the layer that is attached to the output. For the SCOREQ-based model, we project the SCOREQ *Const* version since it represents the novelty of our work. In this case, we project the output of the embedding layer $f(\cdot)$ which is the input of the SCOREQ loss. Notice that in Table 11 we compute MOS correlation performance using the encoder representations $g(\cdot)$ demonstrating that quality is predicted significantly better than the L2 loss also in the encoder representations and not only in the embedding layer. The Pearson's correlation (PC) results reported in the Figure are calculated in NMR mode, they correspond to the same results reported in Table 3 for the SCOREQ *Const* model. The PC of the L2 loss is taken from Table 11. PC is computed to show the amount of quality information that is explained in both feature representations. The Normalised Mutual Information (NMI) scores are computed after running K-Means with 5 clusters using the same representations of PC calculation. The NMI is computed between the output of the K-Means algorithm and the 5 degradation labels. This is done to quantify the amount of degradations explained by both feature representations. The results show that the L2 loss feature representation tends to cluster degradation information more than SCOREQ loss. Instead, our proposed loss function learns ordered representations with respect to target MOS. Although degradation information is linked to quality it is also orthogonal i.e., two degradation conditions are labelled with equal MOS. So having representations with lower NMI between cluster embeddings and degradations is preferred for MOS quality prediction.

J Gradient Adaptive Margin

The usefulness of the adaptive margin is confirmed by inspecting the gradient. Assuming that the anchor is i, the positive is j, and the negative is k, the gradient with respect to the anchor sample is:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{x}_i} = \begin{cases} \boldsymbol{x}_k - \boldsymbol{x}_j, & \text{if } ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2 + \frac{|y_i - y_j| - |y_i - y_k|}{N-1} > ||\boldsymbol{x}_i - \boldsymbol{x}_k||_2 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

The gradient for the positive and the negative samples can be obtained with the same approach.

K Experiments using w2vlight architecture

We run the same experiments using the NISQA TRAIN SIM corpus but with the w2vlight architecture. Table 8 shows that the SCOREQ loss improves over the L2 loss in every test set. This means that less powerful architectures could also benefit from our proposed approach.

Table 8: Pearson's Correlation (PC) performance using w2vlight. We use the SCOREQ *Adapt* model, which was the best with the original wav2vec 2.0 architecture.

Dataset	L2 loss	SCOREQ (*Ours*)
NISQA TEST FOR	0.83	**0.86**
NISQA TEST P501	0.86	**0.90**
P23 EXP1	**0.85**	0.85
P23 EXP3	0.85	**0.92**
NOIZEUS	0.71	**0.80**

Dataset	L2 loss	SCOREQ (*Ours*)
NISQA TEST LIVETALK	0.66	**0.80**
TCD-VOIP	0.85	**0.87**
TENCENT	0.80	**0.83**
TENCENT-Rev	0.25	**0.35**
VoiceMOS Test 1	0.69	**0.75**
VoiceMOS Test 2	**0.75**	0.69

L Evaluation of Ranking and Average Precision

We compare the state-of-the-art quality metrics against our proposed models using Spearman's Rank Correlation Coefficient (SC) and the root mean squared error (RMSE) to evaluate the monotonic relationship and the average precision between predicted and ground truth MOS. In all our experiments we consider the predictions after one degree polynomial mapping to adjust for the bias of each listening test. This will only affect the RMSE values. We observe that each quality metric performs the best in test sets corresponding to the domain they are trained for. The NISQA metric's highest performance is reported for NISQA TEST P501 and NISQA TEST FOR which include several simulated degradations similar to the training set. Performances on other sets that are in the same domain but produced with different listening tests and unseen conditions are much lower (P23 EXP1, P23 EXP3, TCD VOIP). The model does not equally perform for real phone calls (NISQA TEST LIVETALK) showing generalisation issues when training quality metrics with artificial degradations that try to emulate real phone calls. The significant gap between TENCENT-Rev and TENCENT

Table 9: Performance evaluation with Spearman's Rank Correlation Coefficient (SC).

Dataset		NISQA	NR-PESQ	NR-SI SDR	NORESQA-M	NR-SCOREQ (*Ours*)
		D1		D2	D3	D1
NISQA TEST FOR	D1●D2◉D3○	0.89	0.73	0.74	0.58	**0.95**
NISQA TEST P501		0.94	0.88	0.81	0.81	**0.97**
VoiceMOS Test 1	D1○D2○D3●	0.50	0.71	0.65	**0.87**	**0.87**
VoiceMOS Test 2		0.35	0.51	0.62	**0.90**	0.81
NOIZEUS	D1◉D2◉D3○	0.86	0.72	0.66	0.15	**0.90**
NISQA TEST LT		0.78	0.59	0.58	0.51	**0.85**
P23 EXP3		0.74	0.62	0.04	0.48	**0.89**
TCD VOIP		0.81	0.79	**0.87**	0.71	**0.85**
TENCENT		0.77	0.80	0.76	0.59	**0.87**
P23 EXP1	D1◉D2○D3○	0.70	0.84	0.83	0.35	**0.96**
TENCENT-Rev	D1○D2○D3○	0.39	0.16	0.30	0.20	**0.78**

Table 10: Performance evaluation root mean squared error (RMSE).

Dataset		NISQA D1	NR-PESQ D2	NR-SI SDR	NORESQA-M D3	NR-SCOREQ (Ours) D1
NISQA TEST FOR	D1●D2◉D3○	0.34	0.50	0.55	0.60	**0.21**
NISQA TEST P501		0.32	0.47	0.56	0.69	**0.28**
VoiceMOS Test 1	D1○D2○D3●	0.68	0.57	0.60	0.42	**0.41**
VoiceMOS Test 2		0.72	0.81	0.78	**0.39**	0.53
NOIZEUS	D1◉D2◉D3○	0.26	0.32	0.34	0.48	**0.20**
NISQA TEST LT		0.44	0.61	0.67	0.65	**0.42**
P23 EXP3		0.41	0.46	0.71	0.51	**0.25**
TCD VOIP		0.62	0.63	0.63	0.77	**0.51**
TENCENT		0.50	0.42	0.44	0.70	**0.62**
P23 EXP1	D1◉D2○D3○	0.70	0.84	0.83	0.35	**0.22**
TENCENT-Rev	D1○D2○D3○	0.39	0.16	0.30	0.20	**0.56**

suggests that quality metrics suffer when there are unseen conditions such as real-world reverberation (TENCENT-Rev), even if tested on distant languages (testing on Chinese and training on English). The no-reference PESQ and SI-SDR do not achieve the same results on their domain as reported in the Squim paper. In particular, they fail significantly for real phone calls and Chinese reverberated speech. NORESQA-MOS shows a performance drop in all the datasets that are not synthetic speech, showing a lower correlation for P23 EXP1 which includes speech encoded up to three times (transcoding).

M Assessment of the Encoder $g(\cdot)$ representations.

As mentioned in the paper, the NR SCOREQ models are trained in 2 steps. We first train encoder $g(\cdot)$ and the embedding layer (projection head) $f(\cdot)$ with the SCOREQ loss. The second step is based on training a mapping between the encoder $g(\cdot)$ representations and MOS with the encoder frozen from step 1. We discard the embedding layer in this stage. Our main contribution is that encoder representations learned with the SCOREQ are not fragmented and allow easier mapping to MOS prediction than L2 end-to-end training. For this reason, here we analyse how much the encoder representations capture quality. To do that, we extract the output of the encoder $g(\cdot)$ and measure the Euclidean distance with non-matching reference as done in the NMR models illustrated in the paper. We evaluate linearity (PC) and ranking (SC) between the distance and the MOS labels. We use the same 50 unpaired clean speech samples from the Librispeech test-clean partition as non-matching references. We evaluate the encoder representations of the L2 loss baseline, NOMAD, the *Offline* model, SCOREQ *Const* and SCOREQ *Adapt*.

Table 11: MOS correlations for encoder $g(\cdot)$ representation distances. Domain shift (IN, ODS, ODM) is labelled with respect to the training set NISQA TRAIN SIM.

Test Set	Domain	L2 loss PC	SC	NOMAD PC	SC	Offline PC	SC	SCOREQ (Ours) Const PC	SC	Adapt PC	SC
NISQA TEST FOR	IN	0.75	0.75	0.71	0.65	0.59	0.54	**0.94**	**0.92**	0.86	0.85
NISQA TEST P501	IN	0.76	0.85	0.84	0.84	0.62	0.64	**0.92**	**0.95**	0.80	0.87
P23 EXP1	ODS	0.89	0.90	0.88	0.88	0.57	0.62	**0.97**	**0.96**	0.93	0.94
P23 EXP3	ODS	0.54	0.54	0.83	0.70	0.54	0.58	**0.90**	**0.82**	0.64	0.65
NOIZEUS	ODS	0.58	0.63	0.53	0.52	0.53	0.52	**0.90**	**0.89**	0.83	0.81
NISQA TEST LIVETALK	ODS	0.75	0.73	0.78	0.71	0.55	0.55	**0.87**	**0.87**	0.86	0.85
TCD VOIP	ODS	0.53	0.60	0.67	0.67	0.44	0.45	**0.82**	**0.85**	0.57	0.50
TENCENT	ODS	0.74	0.76	0.63	0.73	0.73	0.71	**0.84**	**0.86**	0.84	0.81
TENCENT-Rev	ODM	0.60	0.59	0.55	0.43	0.54	0.55	**0.80**	**0.78**	0.75	0.76
VoiceMOS Test 1	ODM	0.80	0.79	0.79	0.79	0.54	0.58	**0.83**	**0.85**	0.74	0.77
VoiceMOS Test 2	ODM	**0.76**	**0.78**	0.35	0.45	0.45	0.38	**0.80**	**0.79**	0.73	0.71

The results indicate that both SCOREQ models effectively capture quality information in the encoder output. We find that the L2 loss correlates less with MOS, suggesting that additional factors are entangled in the representation. This observation implies that our model could be trained on a large available MOS dataset to tune the encoder, and then adapt the linear layer using a smaller, domain-specific annotated dataset. This approach is particularly beneficial in domains where data is scarce, as collecting MOS through listening tests is often resource-intensive.

N Statistical Analysis

We consider the case of overlapping and dependent correlation coefficients i.e. when one variable is in common (MOS ground truth in our case) and we want to test whether correlation $\rho(X_{MOS}, Y_{SCOREQ})$ is different from correlation $\rho(X_{MOS}, Y_{L2})$. For each test set T, we resample N_T times with replacement, where N_T is the sample size of dataset T. We compute PC using bootstrap samples and then we compute the difference $\rho_d = \rho(X_{MOS}, Y_{SCOREQ}) - \rho(X_{MOS}, Y_{L2})$. We execute this procedure 15000 times in each dataset and use the empirical distribution to compute a 95% confidence interval of the difference between the two correlation coefficients. We conclude that the 2 correlations are significantly different if the confidence interval does not include 0. We conduct a two-tail test and report the corresponding p-values to evaluate significance. The null hypothesis is that the difference between the two correlations includes the zeros. If p-value is lower than 0.05 we have evidence to reject this hypothesis and conclude that the two correlation coefficients are statistically different. We conduct a statistical difference between the NR SCOREQ *Adapt* model against the L2 loss model trained on both NISQA TRAIN SIM (Table 3) and VoiceMOS Train 4. In this way, we test the generalisation of training on simulated degradations and testing on synthetic speech and vice-versa. These statistical tests examine the flexibility of SCOREQ that can be adapted to train models in more than one domain.

Simulated degradations Our statistical analysis in Table 12 finds that SCOREQ is significantly better than the L2 loss model in all the ODS and ODM datasets except for the NISQA LIVE TALK datasets. There is no difference for IN sets, which further confirms the generalisation issues of deep learning-based speech quality metrics. Figure 6 shows the bootstrapping procedure on 2 datasets.

Table 12: Bootstrap statistics of PC computed on NR SCOREQ *Adapt* against L2 loss trained on the NISQA TRAIN SIM datasets. Domain shift (IN, ODS, ODM) is labelled relatively to the simulated degradations domain of the training set.

Dataset	Domain	p-value	Confidence Interval	Outcome Test	SCOREQ (PC)	L2 (PC)
NISQA TEST FOR	IN	0.185	[-0.002, 0.014]	No Diff.	0.97	0.96
NISQA TEST P501	IN	0.348	[-0.004, 0.0012]	No Diff.	0.96	0.95
P23 EXP1	ODS	0.319	[-0.011, 0.033]	No Diff.	0.96	0.95
P23 EXP3	ODS	0.325	[-0.035, 0.009]	No Diff.	0.92	0.95
NOIZEUS	ODS	0.000	[0.230, 0.444]	**SCOREQ**	0.90	0.84
NISQA TEST LIVETALK	ODS	0.450	[-0.020, 0.037]	No Diff.	0.87	0.83
TCD VOIP	ODS	0.000	[0.030, 0.091]	**SCOREQ**	0.85	0.80
TENCENT	ODS	0.000	[0.043, 0.052]	**SCOREQ**	0.86	0.81
TENCENT-Rev	ODM	0.000	[0.034, 0.055]	**SCOREQ**	0.79	0.74
VoiceMOS Test 1	ODM	0.000	[0.069, 0.150]	**SCOREQ**	0.86	0.75
VoiceMOS Test 2	ODM	0.001	[0.043, 0.248]	**SCOREQ**	0.82	0.69

Speech synthesis We conduct same bootstrapping for the models trained on the speech synthesis domain. Table 13 shows that SCOREQ is significantly better than the L2 loss in 5 test sets worse in only one. This confirms that the SCOREQ loss can improve generalisation for speech quality metrics.
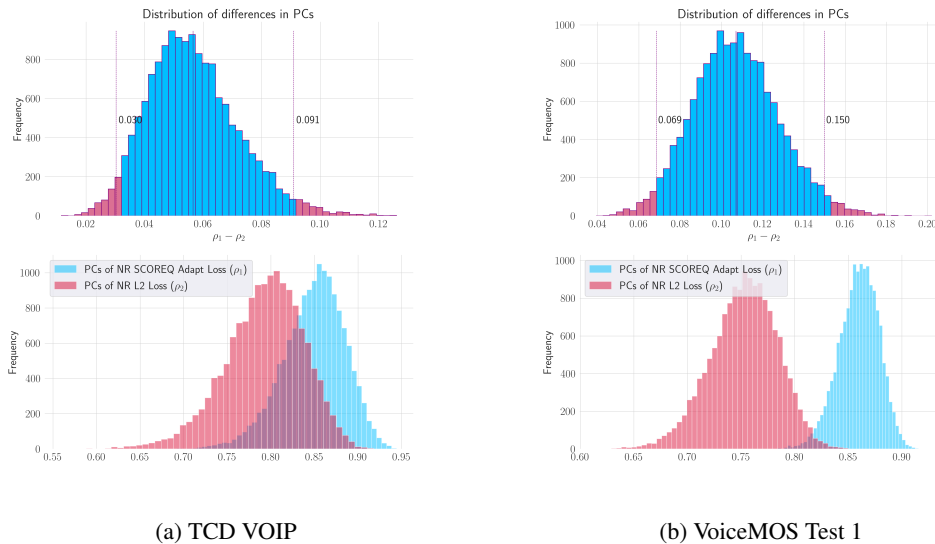
(a) TCD VOIP   (b) VoiceMOS Test 1

Figure 6: Bootstrapping on 2 datasets where NR SCOREQ *Adapt* performs better than the L2 loss. We can see how in many subsets, our model shows improvement. We only report these 2 datasets but the same trend is confirmed in all the test sets where SCOREQ is better as shown in Table 12

.

Table 13: Bootstrap statistics of PC computed on NR SCOREQ *Adapt* against L2 loss trained on the VoiceMOS Train dataset. Domain shift (IN, ODS, ODM) is labelled relatively to synthetic speech domain of the training set.

Dataset	Domain	*p*-value	Confidence Interval	Outcome Test	SCOREQ (PC)	L2 (PC)
VoiceMOS Test 1	IN	0.341	[0.016, 0.096]	No Diff.	0.91	0.90
VoiceMOS Test 2	IN	0.167	[0.043, 0.248]	No Diff.	0.97	0.98
NISQA TEST FOR	ODM	0.011	[0.016, 0.096]	**SCOREQ**	0.87	0.82
NISQA TEST P501	ODM	0.029	[0.002, 0.057]	**SCOREQ**	0.89	0.86
P23 EXP1	ODM	0.730	[-0.021, 0.030]	No Diff.	0.92	0.92
P23 EXP3	ODM	0.627	[-0.052, 0.089]	No Diff.	0.87	0.85
NOIZEUS	ODM	0.002	[0.034, 0.145]	**SCOREQ**	0.67	0.59
NISQA TEST LIVETALK	ODM	0.154	[-0.008, 0.064]	No Diff.	0.84	0.81
TCD VOIP	ODM	0.005	[0.015, 0.068]	**SCOREQ**	0.86	0.83
TENCENT	ODM	0.000	[-0.019, -0.08]	L2	0.77	0.78
TENCENT-Rev	ODM	0.000	[0.079, 0.105]	**SCOREQ**	0.44	0.35

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our contribution is clearly stated in abstract and introduction

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.

- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We created a separate section for limitations (Appendix B). We outline the scope of our paper and the limitations of our proposed solution.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Paper not including theoretical results

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper is fully reproducible. We have provided the weights, as well as detailed instruction in Appendices G and H Although we will release the full code, experiments are fully reproducible with the information provided.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We used both publicly available data and pre-trained models. Also, we submitted the necessary code to run the metrics. The full code will be released upon acceptance of the paper as well as a pip package.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Training details are all provided in Appendix H and G. Other details throughout the paper. We used publicly available train/validation/test partitions.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Bootstrapping with confidence interval are calculated to compare our metric with the standard L2 loss approach. We compute statistics on 11 test sets. Details in Appendix N.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We give details on the GPU used and number of workers in Appendix H.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Our work uses available resources that are well known by the community. No human-participants has been directly involved by us.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We created a separate section for societal impact in Appendix C. We believe there is no negative impact since our work is a speech quality metric.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our model predicts quality scores.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We did not provide a licence. Our work is based on publicly available assets for the research community. We cited original work.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We submitted the minimal code with instructions to run 2 speech quality metrics. Upon acceptance of the paper we will release the full code and create an exhaustive README file to run experiments.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.