

Investigation of preference-based speech quality assessment by integrating adaptive pair selection and pseudo-labeling

ChengHung Hu 

Nagoya University, Nagoya, Japan

Yusuke Yasuda

*Digital Content and Media Sciences Research Division,
National Institute of Informatics, Chiyoda-ku, Japan, and*

Tomoki Toda

Nagoya University, Nagoya, Japan

Abstract

Purpose – This paper aims to investigate the efficiency and viability of online pair selection methods for preference score collection in Speech Quality Assessment (SQA). It addresses the challenge of high annotation costs in traditional pairwise comparisons and explores whether preference data collected in a sparse manner can successfully produce accurate rankings and support robust SQA model training.

Design/methodology/approach – We propose an improved aggregation method to derive system-level scores from raw preference data. To equalize uncertainty across system pairs, we introduce a novel bias-based pair selection method. Furthermore, we develop a pseudo-labeling strategy that utilizes unlabeled utterances to infer missing preferences, thereby augmenting the training data and enhancing SQA model performance.

Findings – Experiments demonstrate that the proposed bias-based pair selection method achieves the most robust performance. Additionally, the proposed pseudo-labeling approach effectively compensates for data sparsity, significantly improving the SQA model's performance by leveraging unlabeled datasets.

Originality/value – This paper fulfills an identified need for cost-effective SQA data collection. It bridges the gap between active pair selection and deep learning-based SQA model training by introducing a novel bias-based selection mechanism and a pseudo-labeling strategy that leverages system-level scores to predict missing preferences.

Keywords Speech quality assessment, Preference prediction, Direct assessment scores, Online pair selection method

Paper type Research paper

1. Introduction

Speech quality is commonly evaluated using subjective listening tests based on direct assessment (DA) scores, in which a group of listeners rate the perceived quality of speech



samples. The final quality score of a speech sample is typically obtained by aggregating these scores into a mean opinion score (MOS). For speech processing systems, performance is further evaluated by averaging the MOS across all generated samples. However, ensuring the robustness of these subjective metrics requires a substantial number of listeners, making DA-based evaluation inherently resource-intensive and time-consuming.

To reduce the cost and time required for subjective evaluation, speech quality assessment (SQA) aims to automatically evaluate or rank the quality of speech samples or speech processing systems. In recent years, driven by the broader success of artificial intelligence and deep learning with state-of-the-art methods across various domains (Naoumi *et al.*, 2026; Delamou *et al.*, 2023), data-driven approaches have gained increasing attention. In these approaches, machine learning models are trained to predict perceived speech quality directly from speech signals, using human-annotated scores (such as DA scores or MOS) as ground truth (Mittag *et al.*, 2021; Huang *et al.*, 2022; Lo *et al.*, 2019). Thanks to the availability of large-scale public datasets from various domains, such as voice conversion (Toda *et al.*, 2016; Lorenzo-Trueba *et al.*, 2018; Yi *et al.*, 2020; Huang *et al.*, 2022; Cooper *et al.*, 2023; Huang *et al.*, 2024), simulated and real noisy speech (Mittag *et al.*, 2021), noisy and denoised speech (Hu and Loizou, 2007), text-to-speech systems (Black and Tokuda, 2005; King and Karaikos, 2013; Perrotin *et al.*, 2023; Maniati *et al.*, 2022), and synthesized singing voice (Cooper *et al.*, 2023; Huang *et al.*, 2023; Tang *et al.*, 2024), the research on SQA using DA scores for training and testing has grown rapidly.

However, despite the increasing use of DA scores in SQA research, previous research has pointed out that DA scores are often subject to various biases (Zielinski *et al.*, 2007), which hinder an accurate evaluation. Furthermore, since DA scores are ordinal data, calculating an average score from them is considered meaningless (Marcus-Roberts and Roberts, 1987). Therefore, the results obtained using DA scores for SQA are possibly inaccurate and lack mathematical significance.

In contrast to DA scores, preference scores can also be used for SQA. Similar to DA scores, a group of listeners is gathered to conduct listening tests to compare the quality of two speech samples. Preference scores are generally considered easier and faster for human raters to evaluate than DA scores (Thurstone, 1927), which contributes to their less noisy nature (Shah *et al.*, 2014; Teodorescu *et al.*, 2016). By preference scores, utterance scores or system scores can be computed and ranked using voting methods or mathematical modeling (Thurstone, 1927; Bradley and Terry, 1952). The scores or rankings obtained this way are more accurate and mathematically meaningful than those based on DA scores. However, unlike DA scores, large-scale publicly available datasets of real preference scores are novel and scarce. While some studies have used DA-score-derived preference scores (Valentini-Botinhao *et al.*, 2022; Hu *et al.*, 2023; Hu *et al.*, 2024), research using real preference scores remains limited (Hu *et al.*, 2025).

The primary bottleneck causing the scarcity of real preference score datasets is the quadratic cost of pairwise comparisons: fully evaluating n systems requires comparing $O(n^2)$ pairs, whereas DA scores only scale linearly [$O(n)$]. To mitigate this, Yasuda and Toda (2024) proposed a dynamic preference test that successfully addresses the cost issue by reducing the number of required pairs through online adaptation. However, while these methods achieve efficiency, the reliability and downstream utility of the scores obtained from such optimization remain insufficiently verified. To address this gap, we assess the effectiveness of pair selection strategies from two critical evaluation perspectives: (a) True score derivation: The validity of scores derived from such sparsely sampled subsets remains insufficiently verified. Yasuda and Toda (2024) evaluated their method primarily by comparing the ranking consistency with MOS. However, MOS is a fundamentally different

metric with its own biases and cannot serve as the absolute ground truth for preference tests. Crucially, prior work did not verify whether the scores from the sparse subset actually match the true scores calculated from the full preference dataset, leaving the accuracy of the sampling strategy unconfirmed. (b) Model training performance: The utility of using such actively sampled scores for downstream SQA model training remains unexplored. [Yasuda and Toda \(2024\)](#) focused solely on ranking analysis and did not investigate the feasibility of using the resulting sparse dataset for model training. This is a critical gap because pair selection inherently drastically reduces the number of labeled utterance pairs compared to the full combinatorial dataset. Consequently, the model is exposed to significantly fewer comparisons during training. It remains to be determined whether this reduced supervision is sufficient for model convergence, and crucially, how we can effectively utilize the vast majority of “unselected” (unlabeled) utterance pairs without conducting additional listening tests for model training.

Guided by these evaluation perspectives, we systematically compare three distinct pair selection strategies under a fixed annotation budget. First, as a baseline, we examine a pair balanced strategy that distributes the budget uniformly across all system pairs. Second, we investigate the applicability of the adaptive preference-testing framework by [Yasuda and Toda \(2024\)](#), referred to here as a standard Merge-Rank (MR). While this framework was originally designed to output a stochastic ranking based on transitivity, we propose a novel extension to this approach: specifically, we aggregate the raw comparison outcomes collected during its execution to derive precise system-level scores via voting methods or mathematical modeling. Third, we propose a novel strategy termed MR Bias Equalization. Unlike the standard MR which focuses on sorting convergence, this method actively guides the pair selection process to equalize the estimated uncertainty – specifically the MR-based bias error – across all system pairs during data collection. We evaluate these approaches by simulating subset sampling from a full preference dataset and measuring the correlation between the derived scores and the ground truth, as well as their effectiveness in training SQA models.

Regarding model training performance, while pair selection drastically improves efficiency, it inherently reduces the number of labeled utterance pairs, leaving the majority of potential comparisons undefined. To compensate for this data scarcity and enable the model to use the vast pool of unselected utterances, we propose a pseudo-labeling strategy. Pseudo-labeling is a highly effective semi-supervised approach widely recognized for mitigating limited data issues across diverse domains ([Pham et al., 2021](#); [Njima et al., 2022](#)). Inspired by its success, we adapt this concept to the unique context of preference tests. Specifically, we use the preference scores collected from the sparse subset to estimate system-level quality, assigning them as “pseudo-scores” to all utterances. Crucially, we use these pseudo-scores as a numerical standard to conduct pairwise comparisons between these previously unselected utterances. This process generates synthetic preference labels, effectively transforming the vast amount of unlabeled data into a rich supervision signal for model training.

The structure of this paper is as follows: in Section 2, we delve into related research on SQA and pseudo-labeling methods; in Section 3, we introduce our proposed pair selection methods and preference-based SQA framework used for experiments. In Section 4, we present our simulations and experiments. Finally, in Section 5, we discuss the limitations of our study and conclude our findings.

2. Related work

2.1 Speech quality assessment model

Previous speech quality assessment models, such as MOSNet ([Lo et al., 2019](#)), LDNet ([Huang et al., 2022](#)), and MBNet ([Leng et al., 2021](#)), have primarily focused on training and

testing with individual speech samples and their corresponding DA scores. Recently, many model architectures have incorporated the process of learning preferences into their training. This is because incorporating references during training allows the model to avoid learning the distribution of hypothetical perfect speech samples assumed by raters, thereby improving the model's performance (Manocha *et al.*, 2021). In addition, since humans are generally better at determining preferences than assigning absolute scores, imitating the human preference decision process has been shown to improve model performance (Manocha *et al.*, 2022). However, most current approaches only use this technique for pre-training and do not incorporate preference scores as a method for final evaluation.

DPAM (Manocha *et al.*, 2020) and CDPAM (Manocha *et al.*, 2021) leverage their collected Just Noticeable Differences dataset to perform metric and contrastive learning using neural networks. By introducing noise at various levels to speech samples, they generate paired reference samples. The model then learns to extract similar or dissimilar utterance-level embeddings depending on whether human-annotated perceptible differences exist between them.

For NORESQA (Manocha *et al.*, 2021), nonmatching utterance pairs are created during training by adding noise separately to two speech samples. The noise levels are then compared to derive preference scores. NORESQA trains on these nonmatching utterance pairs and their associated preference scores. In the testing phase, one or more clean, nonmatching reference samples are used as perfect quality anchors to predict preference scores. The quality score is then calculated based on the preference score between the test utterance and the clean reference.

SQAPP (Manocha *et al.*, 2022) uses a two-stage training process. In the first stage, like NORESQA, the model is trained on nonmatching utterance pairs and preference scores. In the second stage, it trains directly on individual speech samples and their associated DA scores. During evaluation, the model predicts DA scores for the speech samples directly.

PrefNet (Valentini-Botinhao *et al.*, 2022) compares multiple speech samples' DA scores during training to generate preference scores for training. Although this model relies entirely on preference scores for both training and testing, it focuses only on the accuracy of preference prediction during testing, rather than on utterance- or system-level scores or rankings.

Compared with methods that may use references for training or testing in specific cases, we use E2EPref (Hu *et al.*, 2025), which incorporates four additional components, such as Pair Generation, Preference Function, Threshold Selection, and Preference Aggregation, into the typical DA-score-based SQA framework, allowing it to predict system-level quality scores and rankings in an end-to-end manner.

2.2 Pseudo-labeling approach

The pseudo-labeling approach is a semi-supervised method designed to address situations where labeled data is scarce. This approach has been applied to various tasks, such as image classification (Pham *et al.*, 2021; Xie *et al.*, 2020), automatic speech recognition (Synnaeve *et al.*, 2020; Kahn *et al.*, 2020), indoor localization (Njima *et al.*, 2022), and speech emotion recognition (Venkateswaran *et al.*, 2021). These tasks typically use a pre-trained model, a teacher model, or a model in the training process to predict pseudo labels for unlabeled inputs, which are then used to train the model.

In our approach, we assume that the quality of the speech samples generated by the system remains constant. We use a preference aggregation method to combine the collected preference scores into system scores. These system scores are considered as quality ratings for all speech samples produced by the system. We then compare these scores to generate derived preference scores for training.

3. Methodology

3.1 Problem formulation and method overview

In this study, we focus exclusively on the evaluation of preference-based pair selection methods. Our primary objective is to investigate the effectiveness of different pair selection strategies applied within a crowdsourcing environment. Specifically, we address the following core question:

Q1. Given a fixed annotation budget for preference scores, how can we minimize the degradation in quality prediction accuracy?

To address this, we outline an online pair selection framework intended for real-world crowdsourcing scenarios. As illustrated in the Application Stage (top half) of Figure 1, the system operates as an iterative cycle. In each step, a selection algorithm determines a specific system pair (e.g. System A vs System B) based on a sampling strategy. From this pair, an utterance pair is randomly sampled and presented to listeners for subjective evaluation. This cycle continues until a termination condition is met. However, a critical limitation of this real-world deployment is the absence of ground truth: simply collecting sparse scores does not allow us to quantify how much the predicted quality deviates from the true rankings.

Therefore, to overcome this limitation and enable evaluation against a valid ground truth, we adopt a simulation-based experimental procedure, as depicted in the Evaluation Stage (bottom half) of Figure 1. Instead of deploying the system to live listeners, we simulate the pair selection process using a previously collected complete preference dataset. Specifically, we use the selection algorithm to iteratively select utterance pairs and retrieve their

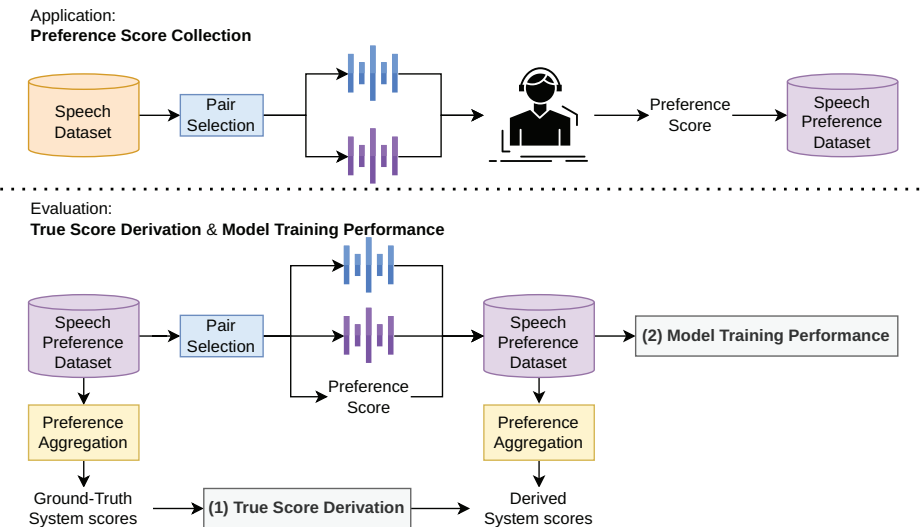


Figure 1. The application and evaluation workflows of the proposed pair selection methods. In the application stage, representative utterance pairs are selected and evaluated by human listeners to collect preference scores, forming a speech preference dataset. In the evaluation stage, the effectiveness of the selection methods is assessed by simulating the pair selection process using a full speech preference dataset

Source: Authors' own work

corresponding preference scores from the full dataset, forming a sub-dataset. We then evaluate this sub-dataset from two perspectives:

- *Ground-truth derivation:* We define the ground truth as system scores aggregated from the full dataset using preference aggregation methods. We then compute the correlation between this ground truth and system scores derived from the subset selected by each pair selection method. A high correlation indicates that the method successfully extracts representative pairs suitable for efficient listening tests.
- *SQA model training performance:* We also investigate how well a model can be trained when only a limited number of comparisons are available. In this context, we further evaluate the effectiveness of our proposed pseudo-labeling strategy, which leverages the vast pool of unselected utterance pairs to augment the training supervision. A high correlation indicates that the method successfully extracts representative pairs suitable for effective SQA model training.

It is important to note that this objective fundamentally differs from reward-maximizing strategies, such as standard multi-armed bandit algorithms. While such methods use sampling to support decision-making (e.g. identifying the best systems), our framework instead uses it to obtain reliable pairwise preference estimates for downstream aggregation and model training.

In this paper, we investigate four pair selection methods: one trivial baseline and three based on the MR method (Falahatgar *et al.*, 2017). Section 3.2 introduces the MR method, explaining how it reduces the number of required preference comparisons while preserving the accuracy of system-level rankings, along with its mathematical foundations. Then, in Section 3.3, we describe all investigated pair selection methods in detail. Section 3.4 introduces a pseudo-labeling approach to further enhance SQA model training efficiency. Finally, to evaluate by SQA model training performance, we adopt the E2EPref framework (Hu *et al.*, 2025), described in Section 3.5.

3.2 Merge-Rank

When we apply MR method on SQA task for scoring and ranking systems, it reduces comparisons on two distinct levels:

- (1) *System-pair selection level:* At the system-pair selection level, MR reduces the number of required comparisons by leveraging the strong stochastic transitivity assumption. Under strong stochastic transitivity, if the algorithm has already established that system i is preferred to system j , and system j is preferred to system k , each with an error rate bounded by δ , then the relative order between i and k can be inferred without performing a direct comparison while preserving the same δ guarantee. This transitive inference allows MR to prune a large number of system-pair comparisons, particularly for pairs with substantial quality differences, and therefore improves comparison efficiency. The complete MR procedure is provided in Appendix 1.
- (2) *Individual comparison level:* At the individual comparison level, MR determines the winner between two systems through repeated stochastic comparisons. To reduce the sampling cost while maintaining correctness, MR incorporates an adaptive early-stopping rule consisting of two conditions. The first condition evaluates an estimation bias term (illustrated in Figure 2). After r repeated comparisons between systems i and j , let $p_{i,j}^r = w/r$, and define the error bias as:

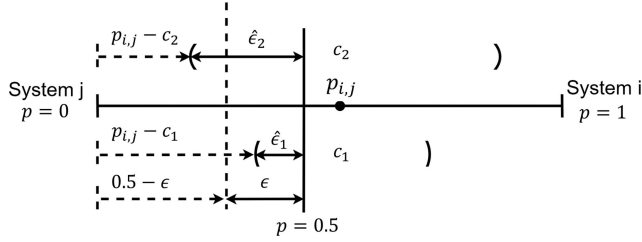


Figure 2. The figure illustrates a comparison between system i and system j based on the empirical preference probability $p_{i,j}$, where, without loss of generality, we assume $p_{i,j} \geq 0.5$. Two different confidence intervals, c_1 and c_2 , correspond to two possible cases of collected preference scores. In the first case, the lower bound of the preference confidence interval, $p_{i,j} - c_1$, exceeds the lower bound of the bias threshold, $0.5 - \epsilon$. Conversely, in the second case, the lower bound $p_{i,j} - c_2$ is below $0.5 - \epsilon$. The first case offers clearer evidence supporting the superiority of system j . In contrast, in the second case, the collected preference scores do not provide strong evidence to conclude that system j is superior to system i . As a result, early stopping can be applied in the first case, where there is clearer evidence that j is better, but not in the second case, where the evidence is insufficient

Source: Authors' own work

$$\hat{e} = c - \left| p_{i,j}^r - \frac{1}{2} \right|, \quad (1)$$

where the confidence term c is:

$$c = \sqrt{\frac{1}{2r} \log \left(\frac{4r^2}{\delta} \right)}. \quad (2)$$

MR halts the comparison early if $\hat{e} < \epsilon$.

The second stopping condition enforces a maximum sampling budget, independent of the estimation bias, to guarantee that the comparison does not continue indefinitely. The budget is given by:

$$m = \frac{1}{2\epsilon^2} \log \left(\frac{2}{\delta} \right), \quad (3)$$

and MR halts the comparison when $r = m$. Together, these two stopping rules ensure that each pairwise decision has an error probability bounded by δ . The full probabilistic guarantees and derivations are provided in [Appendix 2](#).

Original MR methods only output ranking results. In this paper, we not only examine the rankings produced by the original MR algorithm (i.e. the output of Algorithm 1 in [Appendix 1](#)), but also propose a new way to use MR by aggregating system-level scores directly from the collected pairwise comparison outcomes (i.e. all preference scores collected from line 9 in Algorithm 3 in [Appendix 2](#)).

We denote the original MR method as MR (S), which serves as our baseline and produces rankings through system-level transitivity. Since MR (S) only yields system rankings, we evaluate only its ranking correlation in our experiments. In contrast, our proposed method, denoted as MR (U), extends MR by aggregating the raw comparison outcomes using the

preference aggregation methods adopted in the E2EPref framework introduced in Section 3.5. Importantly, because MR avoids comparing system pairs with large score differences, its role in MR (U) is to guide the collection of preference scores specifically for systems with similar estimated scores, making the aggregated scoring both efficient and informative.

3.3 Pair selection

We investigate four pair selection methods: one trivial method and three based on the MR framework. Despite their differences, all methods follow the same iterative process. In each step, we use the pair selection algorithm to choose an informative system pair. From this chosen system pair, we simply randomly sample an available utterance pair and add it to our sub-dataset. This cycle repeats until the stopping criterion is met.

3.3.1 Pair balanced. This method represents a trivial but informative baseline that assumes all system pairs require equal evaluation effort. Given the total number of preference scores to be collected, we gather an equal number of comparison results for each system pair. As a result, this method guarantees full coverage of all system pairs and serves as a reference for evaluating the benefits of adaptive pair selection strategies.

3.3.2 Merge-Rank on MOS-sorted initialized array (MR sorted). This approach applies the MR method with the system array initialized in sorted order based on the MOS, representing a scenario where a coarse prior ranking is available. The goal is to examine how incorporating such prior information influences the selection of system pairs and the resulting preference data. Under this pair selection strategy, we denote the system rankings obtained directly from MR as MR Sorted (S), and the system scores obtained by aggregating the collected preference scores as MR Sorted (U).

3.3.3 Merge-Rank on randomly initialized array (MR random). This method applies the MR framework with the system array randomly initialized, without relying on any prior ranking information. It serves as an unbiased reference for evaluating the effectiveness of adaptive pair selection when no prior knowledge is available. Under this pair selection strategy, we denote the system rankings obtained directly from MR as MR Random (S), and the system scores obtained by aggregating the collected preference scores as MR Random (U).

3.3.4 Merge-Rank bias equalization (MR bias equalization). In addition to MR Sorted and MR Random, we also propose a MR-based pair selection method that focuses on equalizing the MR bias error across all system pairs. MR provides an explicit estimate of uncertainty for each system pair through its bias error, which reflects the reliability of the corresponding preference comparisons under noisy listening conditions. In this study, where preference data are further used for system-level aggregation and pseudo-labeling, maintaining comparable uncertainty across system pairs is important to avoid over-representing highly confident pairs while leaving others under-explored.

Based on this observation, by adopting this approach, we aim to gather comparison results from every system pair, using the bias error $\hat{\epsilon}$ introduced in the MR framework as a criterion to determine how many listening tests each pair requires to reach a similar confidence level. Specifically, during each iteration, we prioritize the system pair with the highest bias error and collect additional listening test data for it. To prevent excessive testing on pairs whose bias error decreases more slowly, we impose a maximum budget constraint that caps the number of tests allowed for any single system pair, ensuring a balanced and efficient evaluation process.

3.4 Pseudo-labeling approach

To compensate for the inherent reduction in labeled utterance pairs caused by pair selection, this approach enables the model to use the vast pool of unselected utterance pairs. Instead of relying solely on the sparse human-annotated subset, we propose a mechanism to extract supervision signals from the available but uncomparing data.

Specifically, we first aggregate the collected preference scores using the preference aggregation methods in the E2EPref framework (introduced in Section 3.5) to derive system-level quality scores. We then assign these derived system scores to every utterance produced by the respective system, designating them as utterance-level pseudo DA scores. This assignment is applied globally to form a “pseudo-labeled dataset” that encompasses all available utterances – including both the vast number of unselected utterances and those that were already involved in real preference comparisons. By establishing these pseudo DA scores as a universal numerical standard within this dataset, we enable the formation of training pairs from any combination of utterances, which are subsequently used in the model training process described in Section 3.5.

3.5 Model structure

In our experimental workflow, the proposed online pair selection method is first used to select a representative sub-dataset. This selected sub-dataset is then used to train and test the downstream model, E2EPref (Hu *et al.*, 2025). E2EPref serves as the backbone SQA framework and processes the input data through four components: Pair Generation, Preference Function, Threshold Selection, and Preference Aggregation, as illustrated in Figure 3.

It is crucial to distinguish our upstream pair selection method from the internal “Pair Generation” component of E2EPref. During training, this component adapts its behavior to support two experimental configurations: (1) training solely with the real preference dataset, and (2) training with a mixture of the real preference dataset and the pseudo-labeled dataset. For the real preference dataset (used in both configurations), the internal generation step acts as a deterministic retrieval mechanism. Since the comparison pairs are explicitly defined by the upstream selection process, the component simply fetches the existing utterance pair and assigns its annotated real preference score as the ground truth (s_{real}). For the pseudo-labeled dataset (used in the second configuration), the component operates in a dynamic construction mode. In this case, two (utterance, pseudo DA score) tuples are randomly sampled to form a training pair. The ground truth is then derived from the comparison of their pseudo DA scores (y_1, y_2). When training with the mixed configuration, pairs are sampled with equal probability from the two sources. Regardless of the source, the model’s predicted preference is defined as:

$$\text{pref}_{\text{dv}} = 2 \cdot \text{sigmoid}(x_1 - x_2) - 1,$$

where x_1 and x_2 represent the scores predicted by the model for the two utterance inputs. The ground truth is formally defined as:

$$\text{pref}_{\text{gt}} = \begin{cases} s_{\text{real}} & \text{if using real preference scores,} \\ \text{sgn}(y_1 - y_2) & \text{if using pseudo DA scores } y_1, y_2. \end{cases}$$

Here, $\text{sgn}(\cdot)$ represents the sign function. The overall training loss is given by: $L = \text{MSE}(\text{pref}_{\text{gt}} - \text{pref}_{\text{dv}})$.

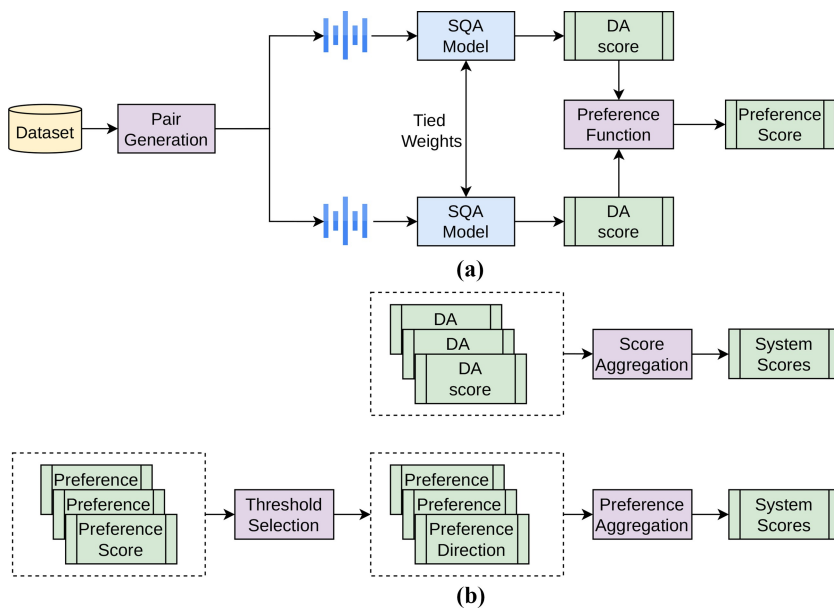


Figure 3. (a) E2EPref SQA framework. One pair of utterances is generated via pair generation. DA scores are predicted for each utterance, and preference scores are computed using a preference function. Training is driven by ground-truth and predicted preference scores. (b) Evaluation stage of E2EPref. After predicting DA and preference scores with the SQA model, DA scores are aggregated into system-level scores. Preference scores are processed via threshold selection and then aggregated using preference aggregation

Source: Authors' own work

During the inference phase, given a set of test systems to be evaluated, our objective is to derive their final system-level scores and rankings. To achieve this, we strictly follow the protocols defined in [Hu et al. \(2025\)](#). The process initiates with Pair Generation, where we construct a specific set of comparison pairs from these test systems to be processed by the subsequent modules. Specifically, we use the Balanced System Pair Selection method for inference. This strategy ensures that every system pair is assigned the same number of test utterance pairs for comparison. Subsequently, the model evaluates the DA scores of the utterances in each generated pair and applies the same Preference Function used during training to produce a continuous preference score for each pair. Regarding Threshold Selection, we apply no filtering threshold; the sign of the predicted score is directly used to determine the preference direction (win or loss) for the subsequent aggregation step. Finally, Preference Aggregation is performed to derive a single scalar score for each system from the collected pairwise outcomes. We use four standard methods: winning count (WC), which sums the total wins; differential count (DC), which calculates wins minus losses; and two generalized linear model (GLM) methods, namely, the Bradley–Terry–Luce (BTL) model and Thurstone’s Case V Model (THUR). As described in [Hu et al. \(2025\)](#), these GLM methods estimate latent system scores such that the distance between two scores correlates with the probability of one outperforming the other. This is achieved via a link function $g(\mu)$ mapping the score difference μ to a probability space $[0, 1]$. Specifically, the BTL model uses

the logit link $g(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$, while the THUR model uses the probit link $g(\mu) = \Phi(\mu)$, where Φ is the standard normal CDF.

4. Experimental evaluations

As introduced in Section 3, we evaluate the pair selection methods and the pseudo-labeling approach from two perspectives: ground-truth derivation and SQA model training performance. For ground-truth derivation, we use correlation metrics to assess pair selection methods by comparing system-level quality scores aggregated from extracted preference scores with those derived from the full set of preference scores. For SQA model training performance, we use correlation metrics to assess the proposed method by comparing system-level quality scores aggregated from predicted and ground-truth preference scores of test dataset.

In Section 4.1, we introduce the correlation metrics used for evaluation. The details of the dataset used in the experiment are provided in Section 4.2. We then present the experimental setup and results for the ground-truth derivation and the SQA model training performance in Sections 4.3 and 4.4, respectively.

4.1 Metrics

To compare system-level quality scores aggregated from extracted preference scores with those derived from the full set of preference scores, we use two widely used correlation metrics: linear correlation coefficient (LCC) and Spearman's rank correlation coefficient (SRCC).

4.1.1 Linear correlation coefficient. The LCC measures the strength and direction of the linear relationship between the predicted system-level scores and the ground-truth system-level scores. It is defined as:

$$\text{LCC} = \frac{\sum_{i=1}^N (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^N (X_i - \bar{X})^2} \cdot \sqrt{\sum_{i=1}^N (Y_i - \bar{Y})^2}} \quad (4)$$

where X_i and Y_i represent the predicted and ground-truth system-level scores for the i th system, respectively, $\bar{X}$ and $\bar{Y}$ are their respective means across all systems, and N is the total number of systems being evaluated.

LCC ranges from -1 to 1 , where values close to 1 indicate a strong positive correlation, values close to -1 indicate a strong negative correlation, and values near 0 suggest no linear relationship. A higher LCC value indicates better alignment between the predicted and ground-truth system-level quality scores.

4.1.2 Spearman's rank correlation coefficient. The SRCC is a rank-based metric that assesses the monotonic relationship between the predicted and ground-truth system-level scores. Unlike LCC, SRCC considers only the order of the values rather than their absolute differences, making it robust to nonlinear relationships. It is computed as:

$$\text{SRCC} = 1 - \frac{6 \sum D_i^2}{N(N^2 - 1)} \quad (5)$$

where D_i is the rank difference between the predicted and ground-truth system-level scores for the i th system, and N is the total number of systems.

SRCC ranges from -1 to 1 , with values near 1 indicating a strong positive monotonic relationship and values near -1 indicating an inverse monotonic relationship. A higher SRCC suggests that the predicted rankings of system-level speech quality closely match the ground-truth rankings.

4.2 Dataset

We used the Voice Conversion Challenge 2020 (VCC2020), which was the third edition of the Voice Conversion Challenge. It comprises two tasks: the intra-lingual semi-parallel VC task and the cross-lingual VC task. In this paper, we focus on the intra-lingual semi-parallel task for our experiments. The available datasets for this task include the VCC2020-DA dataset, which contains DA scores (Yi *et al.*, 2020), and the VCC2020-Pref dataset, which includes the preference scores collected later (Hu *et al.*, 2025).

In VCC2020, there are 32 voice conversion systems, each producing 80 utterances given the same set of input utterances and target speakers. For the VCC2020-DA dataset, each listener provided between 32 and 320 scores, totaling 604 participants. Each utterance was evaluated 12 times, resulting in a total of 30,720 scores ($32 \text{ systems} \times 80 \text{ utterances} \times 12 \text{ evaluations}$). The VCC2020-Pref dataset contains pairwise comparison results for all systems. Each utterance in every system pair was compared twice, leading to 158,720 preference scores ($32 \text{ systems} \times 31 \text{ system pairs} \times 80 \text{ utterances} \times 2 \text{ comparisons}$). Notably, due to the matching utterance design used in the listening test, each comparison involved two utterances generated by different voice conversion models but sharing the same input utterance and target speaker. Across 1,548 participants, each listener evaluated between 40 and 10,040 utterance pairs.

In our experiments, we used two criteria proposed in Hu *et al.* (2025) to split systems into training and testing sets based on the properties of system pairs in each set. The first, termed DACon, involves selecting system pairs for the test set with statistically significant differences in DA scores. This ensures a subset of systems with unambiguous rankings, allowing evaluation based solely on DA scores. The second criterion, PrefCon, aims to select a subset of systems that maximizes the number of conflicting preference directions of its internal system pairs between DA and preference scores. It prioritizes cases where preference scores capture subtle distinctions that DA scores possibly miss. To implement this on the VCC2020-Pref dataset, we first precalculated the pairwise preference directions based on DA scores and preference scores. Then, we brute-forced all $\binom{32}{10}$ combinations of

selecting 10 systems out of 32. For each combination, we evaluated its $\binom{10}{2}$ internal system pairs to count the conflicting directions, ultimately selecting the combination with the maximum conflicts. Thanks to the precalculated directions, the overall time complexity remains highly manageable. These criteria provide complementary perspectives for ranking evaluations.

Using these two criteria, ten systems were selected from the complete dataset for the test set [1], leaving the remaining 22 systems as the training set. Based on the two criteria (DACon and PrefCon), two different dataset splits of the VCC2020-DA dataset were created: one called VCC2020-DACon-DA, and the other called VCC2020-PrefCon-DA. Similarly, the VCC2020-Pref dataset was split according to the same two criteria, resulting in VCC2020-DACon-Pref and VCC2020-PrefCon-Pref. Notably, when splitting the VCC2020-Pref dataset, preference scores were used selectively: only scores between

training systems were included in the training set, while scores between test systems were assigned to the test set.

For the test set under the DACon criterion, the system MOS obtained from all preference aggregation methods for VCC2020-DACon-Pref showed perfect correlation ($SRCC = 1$) with the average system MOS obtained from VCC2020-DACon-DA. In contrast, for the test set under the PrefCon criterion, 25 out of 45 system pairs among the ten selected test systems exhibited differing tendencies in MOS compared to preference scores. The SRCC between the system scores obtained from the BTL method for VCC2020-PrefCon-Pref and the system MOS from VCC2020-PrefCon-DA was approximately 0.733. It is worth noting that only the VCC2020-Pref dataset was used in the experiments; the VCC2020-DA dataset was used to calculate the MOS for the initialization of the MR Sorted method.

4.3 Experiment of ground-truth score derivation

In this subsection, we evaluate the effectiveness of various pair selection methods through a simulation-based approach. Specifically, we extract a subset from the VCC2020-Pref training set using pair selection methods, as illustrated in Figure 1. Before the evaluation, we first investigate the relationship between the adjustment of ϵ and the number of collected scores. Our goal is to identify combinations of MR methods and their corresponding ϵ values that yield a comparable number of collected scores, ensuring a fair comparison between different MR methods. Next, we examine the number of collected scores and the final bias error achieved for each system pair using each proposed selection method. Finally, we evaluate the proposed pair selection methods using the system-level quality scores obtained from the full training set as ground truth, and assess the LCC and SRCC between the subset-derived scores and the ground truth.

4.3.1 Simulation details. We simulate the listening test process for the four pair selection methods using the training set of both VCC2020-PrefCon-Pref and VCC2020-DACon-Pref. The collected preference scores are then aggregated using four preference aggregation methods introduced in Section 3.5. The aggregated system scores or rankings are compared against the ground truth computed from the complete dataset using LCC and SRCC. We plot the LCC and SRCC versus the number of scores. This allows us to evaluate which method can more accurately estimate system-level scores or rankings with the same number of preference scores.

For the Pair Balanced and MR Bias Equalization methods, we consistently collect data in multiples of the total number of system combinations $C_2^{22} = 231$. Initially, we simulate a single multiple. Starting from $n = 1000$, we increase the dataset size in steps of 1,000 up to 20,000, collecting $\text{floor}(n/231) \times 231$ pairs per simulation. In addition, for the calculation of bias error in MR Bias Equalization method, we use $\epsilon = 0.0877$ and $\delta = 0.05$ from Yasuda and Toda (2024). Specifically, we set the confidence level to $\delta = 0.05$ to ensure a standard 95% statistical confidence. The tolerance bias was set to $\epsilon = 0.0877$, which is mathematically derived from Hoeffding's inequality ($m = \frac{1}{2\epsilon^2} \log \frac{2}{\delta}$) by bounding the maximum number of evaluations per system pair to $m = 240$. Structurally, this bound is explicitly tailored to our dataset capacity: it ensures that the algorithm samples at most 75% (240 out of 320) of the available evaluation scores per system pair in the worst-case scenario, preventing data exhaustion while preserving sufficient sampling space to avoid repetitive or deterministic outcomes. For MR methods, we fixed $\delta = 0.05$ and varied ϵ from 0.05 to 0.5 in steps of 0.01. A higher value of ϵ indicates greater tolerance for bias error, which reduces the difficulty in determining the winner in a system pair. Each ϵ value is simulated 100 times, collecting 100 subsets per simulation, resulting in 10,000 subsets in total. Specifically for the MR Random method, the system array initialization is entirely reshuffled for each of the 100 simulations,

rather than being held static. Notably, the number of scores collected in each subset, as well as the SRCC and LCC values compared to the ground truth, may vary.

4.3.2 Simulation results. **4.3.2.1 Effect of ϵ on the number of collected scores.** We first analyzed the relationship between ϵ and the number of collected scores on both VCC2020-DACon-Pref and VCC2020-PrefCon-Pref. In this analysis, we calculated the average number of scores for each ϵ and plotted them in Figure 4(a) and (b). It is important to clarify that we are currently analyzing the raw collected dataset. Since the distinction between MR (S) and MR (U) lies only in the final scoring derivation and not the data collection itself, we present our analysis solely in terms of MR Sorted versus MR Random.

Figure 4(a) shows the changes in the number of collected scores caused by adjusting ϵ in the MR Sorted and MR Random methods on VCC2020-DACon-Pref. We found that as ϵ increases, the number of collected scores starts to decrease exponentially. We believe this is due to the design of the maximum budget m : as ϵ increases, the number of collected scores is limited by a relationship that is inversely proportional to the square of ϵ as mentioned in Section 3.2. Moreover, we observe that the number of collected scores by the MR Random method is consistently greater than that by the MR Sorted method. This suggests that, by the MR method, using MOS for array initialization can significantly reduce the number of scores needed to be collected.

Similarly, we also analyzed the relationship between ϵ and the number of collected scores in VCC2020-PrefCon-Pref. As shown in Figure 4(b), the changes in the number of collected scores for the MR Sorted and MR Random methods when adjusting ϵ were similar to those observed in VCC2020-DACon-Pref. The number of collected scores also decreases exponentially for both MR methods. Under the same ϵ , the MR Sorted method consistently collected fewer scores than the MR Random method. This further supports the observation that, within the MR method, using MOS for array initialization can significantly reduce the number of scores needed to be collected.

4.3.2.2 The number of scores collected and the final bias error reached for each pair selection method. We analyze the number of scores collected and the resulting bias error for each system pair by each pair selection method. For each method, we collect 5,000 scores. Specifically, for MR Random and MR Sorted, we identified the ϵ that allows us to collect approximately 5,000 scores on both VCC2020-Pref splits separately through the analysis in the previous discussion, and used these ϵ to extract preference scores. We plot the relationship between the scores collected by each method and the bias in Figures 5 and 6. In these figures, the x -axis represents the systems sorted by MOS, from left to right, and the y -axis represents the system order from top to bottom after the MOS sorting.

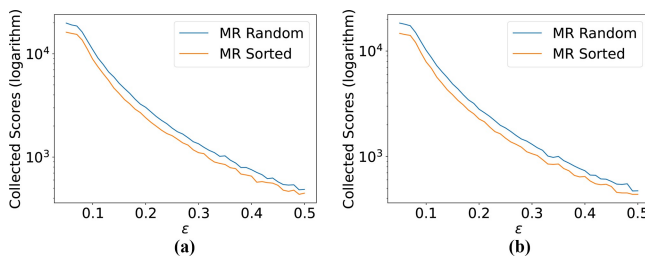


Figure 4. Relationship between ϵ and the number of collected scores for different pair selection methods on (a) VCC2020-DACon-Pref and (b) VCC2020-PrefCon-Pref

Source: Authors' own work

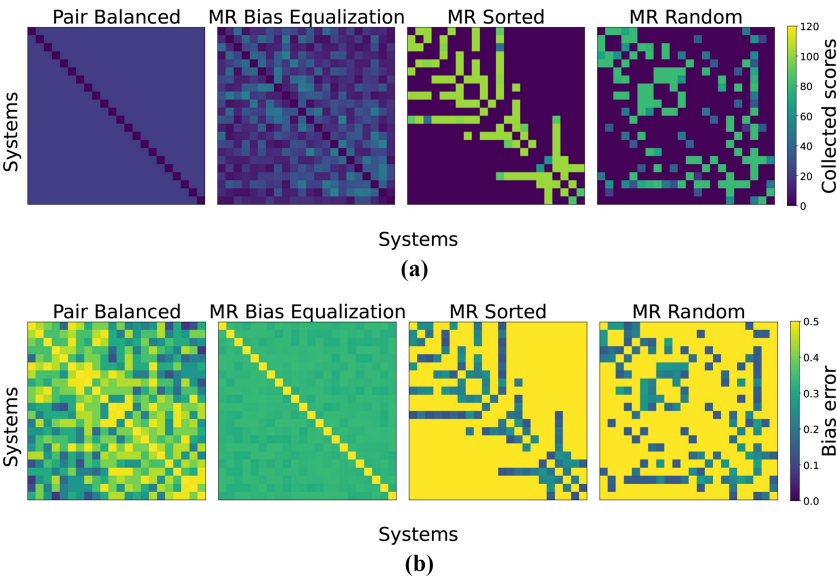


Figure 5. (a) The number of scores and (b) the bias error for each system pair under each pair selection method on VCC2020-DACon-Pref
Source: Authors’ own work

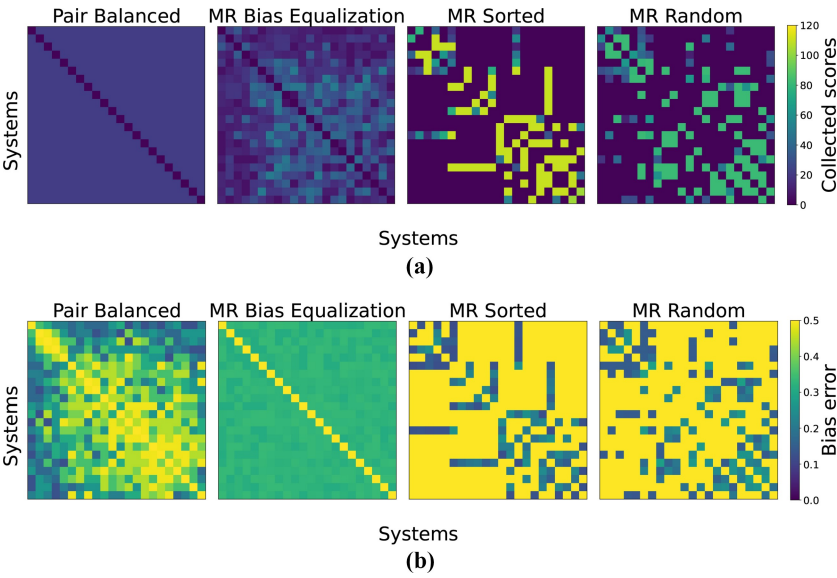


Figure 6. (a) The number of scores and (b) the bias error for each system pair under each pair selection method on VCC2020-PrefCon-Pref
Source: Authors’ own work

First, for both datasets, we observed that the Pair Balanced and MR Bias Equalization methods successfully achieved their objectives: the former extracted the same number of scores for each system pair, while the latter ensured that the bias for each system pair decreased equivalently. For the MR Sorted method, each iteration divided the systems into high-score and low-score groups based on the MOS scores. This approach effectively used the lowest-scoring system from the high-score group to compare with all systems in the low-score group, reducing comparisons between pairs with larger score differences. For example, comparisons in the lower-left and upper-right regions of the plot were entirely avoided in this method. In contrast, MR Random lacks this advantage, as the systems were initially randomly sorted. However, this method still effectively reduced the number of comparisons between system pairs that were farther from the main diagonal. We plotted the average number of scores collected per system pair versus the offset from the main diagonal on the y-axis and x-axis in Figure 7. From this, we can see that the majority of comparisons occurred between system pairs that were closer to the main diagonal.

4.3.2.3 The effectiveness of pair selection methods on correlation with ground truth. In this section, we evaluate the effectiveness of the pair selection methods by measuring the correlation (LCC and SRCC) between the system scores derived from the collected subset and the ground-truth scores aggregated from the full dataset. To ensure a systematic evaluation, we apply a consistent aggregation protocol: the same aggregation function is applied to both the partial collected scores and the full dataset to derive their respective system scores. This process is repeated for each aggregation method, with results presented in Figures 8 and 9. Unlike the previous data collection analysis, here we explicitly distinguish between the MR (U) and MR (S) variants due to their distinct output formats. The MR (U) variants produce continuous scores via the aforementioned aggregation protocol. In contrast, the MR (S) variants generate system rankings and are therefore independent of post hoc aggregation functions. Consequently, while we include MR (S) in the figures for comparison, we evaluate their performance solely using the SRCC metric, as they provide ordinal rankings rather than interval quality scores.

We first examined two mathematical model aggregation methods, THUR and BTL. On the VCC2020-DACon-Pref split, we observed that the Pair Balanced method and the MR Bias Equalization method exhibited similar performance in both SRCC and LCC. The MR Sorted (U) method achieved comparable LCC performance to these two methods when the number of collected scores was low, but its performance slightly dropped when the number of collected scores increased. In terms of SRCC, MR Sorted (U) achieved the best performance overall. MR Random (U) showed the opposite trend compared to MR Sorted

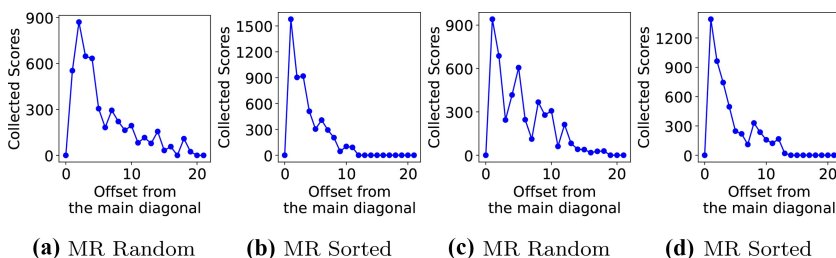


Figure 7. The average number of scores collected for each offset from the main diagonal using the MR random and MR sorted methods on (a, b) VCC2020-DACon-Pref and (c, d) VCC2020-PrefCon-Pref

Source: Authors' own work

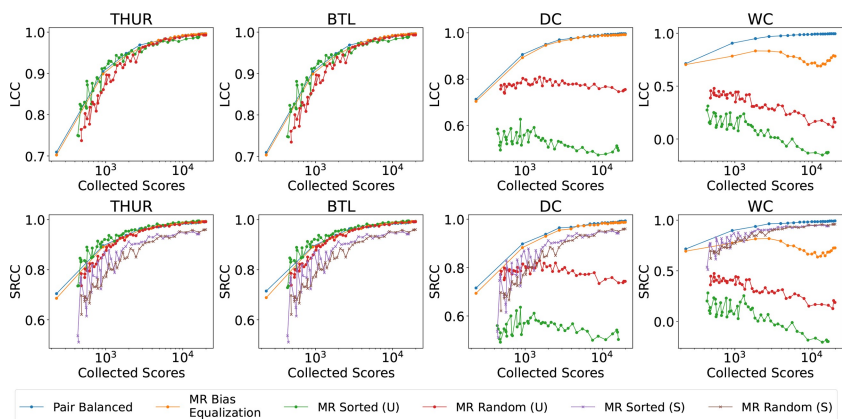


Figure 8. The correlation between collected and complete VCC2020-DACon-Pref results using different preference aggregation methods, evaluated in terms of LCC and SRCC

Source: Authors' own work

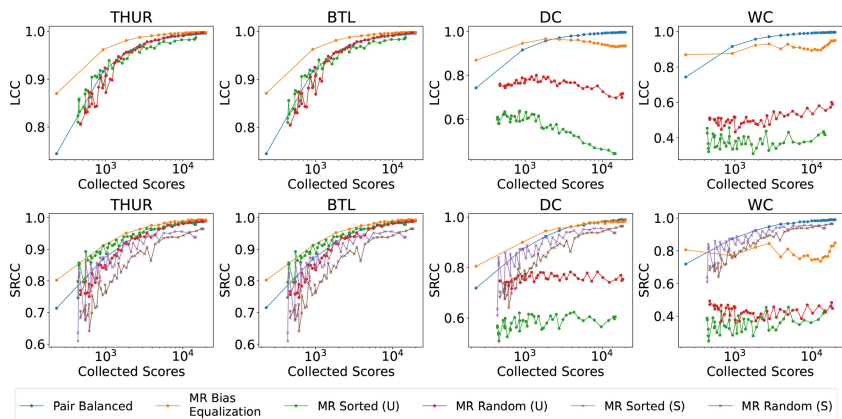


Figure 9. The correlation between collected and complete VCC2020-PrefCon-Pref results using different preference aggregation methods, evaluated in terms of LCC and SRCC

Source: Authors' own work

(U) in LCC. It performed worse than the other three methods when the number of collected scores was low but achieved similar performance to the Pair Balanced and MR Bias Equalization methods when the number of collected scores was high. In terms of SRCC, its performance was also comparable to the Pair Balanced and MR Bias Equalization methods. On the other hand, MR Random (S) and MR Sorted (S), which are based on the original MR output, consistently showed the worst performance among all methods.

On the VCC2020-PrefCon-Pref split, similar trends were observed, except that the MR Bias Equalization method achieved the best performance across all numbers of collected scores. Similarly, MR Sorted (U) performed relatively well with a small number of collected scores but degraded when the number of collected scores increased. MR Random (U) again

showed poor performance with few collected scores but improved as the number of collected scores increased. Meanwhile, MR Random (S) and MR Sorted (S) consistently showed the worst results.

In summary, when using BTL and THUR aggregation methods, we conclude that MR Bias Equalization yields the best performance. It effectively balances the bias across system pairs, leading to more reliable overall system score predictions. Although the MR Sorted (U) method benefits from utilizing MOS information and performs well with a small number of collected scores, it struggles when the number of collected scores increases. We believe this is due to the reduction in the number of system pairs compared, making further performance improvements more difficult. As for MR Random (U), although it is weaker with fewer collected scores, it can achieve performance similar to that of the Pair Balanced method and the MR Bias Balanced method when the number of collected scores is high. In contrast, MR Sorted (S) and MR Random (S) consistently underperform across all scenarios and fail to achieve results comparable to methods that aggregate preference scores.

For the DC and BC counting methods, which depend on the number of system comparisons, imbalanced comparison counts can affect the final aggregated scores. From the LCC and SRCC results, it is evident that the Pair Balanced method performed the best. Although the MR Bias Equalization method balanced system comparisons in some cases, it still underperformed compared to Pair Balanced. The MR methods, due to imbalances in system comparison counts, performed worse than both Pair Balanced and MR Bias Equalization. Among the MR methods, MR Random (U) and MR Random (S) outperformed MR Sorted (U) and MR Sorted (S), respectively. This is because the random initialization of the system list in MR Random methods helps balance the frequency of comparisons between higher- and lower-quality systems, which are compared less frequently in MR Sorted (U) and MR Sorted (S).

In conclusion, the combination of MR Bias Equalization and GLM aggregation methods emerges as the most robust configuration, demonstrating consistent effectiveness across varying numbers of collected scores and dataset splits. When compared with Pair Balanced, MR Bias Equalization shows clear advantages, especially when the number of preference scores is small. On VCC2020-DACon-Pref, the two methods achieve similar performance under low pair counts. However, on VCC2020-PrefCon-Pref, Pair Balanced is consistently limited to around 0.75 LCC and SRCC regardless of the aggregation method, whereas MR Bias Equalization combined with and without the GLM methods consistently achieves LCC and SRCC above 0.8. This indicates that MR Bias Equalization preserves system-level ranking quality more reliably when data are sparse. MR Random (S) and MR Sorted (S) perform significantly worse than all other methods and are not recommended. If an MR method must be used, our proposed MR (U), which aggregates all collected preference scores, is a better choice than MR (S), although it remains less robust than MR Bias Equalization.

4.4 Experiment of model training performance

4.4.1 Experiment details. In this section, we analyze the effectiveness of pair selection methods and the pseudo-labeling approach based on model performance. For each method, we collect approximately 5,000 scores for model training. In addition, we also evaluate models trained with and without the proposed pseudo-labeling method. For the MR methods, we use only our proposed MR (U) method to compute pseudo system scores. We do not use MR (S), as the simulation results in Section 4.3.2 show that MR (S) produces poor ranking accuracy and is therefore unsuitable for generating reliable pseudo labels. When applying MR (U), we pair it with the DC aggregation method, which provides stable system-level scores for the pseudo-labeling process. As a baseline, we also report models trained on the

full dataset. We evaluate both this baseline and the pair selection methods with and without the pseudo-labeling approach. When implementing the pseudo-labeling approach, specific aggregation methods are used to derive the requisite system-level quality scores: we use the DC aggregation method for the Full Dataset and the Pair Balanced method, while using the BTL aggregation method for the three MR variants [MR Sorted (U), MR Random (U), and MR Bias Equalization]. To ensure the reliability of the results, we repeated each experimental configuration 20 times independently. The final results, including the mean and statistical comparisons based on p -values derived from these 20 trials, are summarized and presented in [Tables 1 and 2](#).

Table 1. Model training result on VCC2020-DACon-Pref. An asterisk * indicates that the model uses pseudo scores for training and performs significantly better than models that do not use pseudo scores ($p < 0.05$). The italic cell indicates the best-performing model among those not trained on the full dataset

Pair selection	Pseudo-labeling approach	THUR		BTL		DC		WC	
		SRCC	LCC	SRCC	LCC	SRCC	LCC	SRCC	LCC
Pair balanced	✓	0.856	0.756	0.853	0.751	0.853	0.780	0.853	0.780
		<i>0.941*</i>	<i>0.952*</i>	<i>0.941*</i>	<i>0.951*</i>	<i>0.941*</i>	<i>0.942*</i>	<i>0.941*</i>	<i>0.942*</i>
MR bias		0.865	0.782	0.863	0.777	0.863	0.802	0.863	0.802
equalization	✓	0.939*	0.949*	0.937*	0.945*	0.937*	0.939*	0.937*	0.939*
MR sorted (U)	✓	0.855	0.766	0.855	0.761	0.855	0.785	0.855	0.785
		0.935*	0.951*	0.935*	0.947*	0.935*	0.942*	0.935*	0.942*
MR random (U)	✓	0.848	0.745	0.844	0.740	0.845	0.767	0.845	0.767
		0.934*	0.949*	0.934*	0.945*	0.934*	0.933*	0.934*	0.933*
Full dataset	✓	0.955	0.932	0.954	0.930	0.954	0.942	0.954	0.942
		0.940	0.955*	0.940	0.950*	0.934	0.948	0.934	0.948

Source(s): Authors' own work

Table 2. Model training result on VCC2020-PrefCon-Pref. An asterisk * indicates that the model uses pseudo scores for training and performs significantly better than models that do not use pseudo scores ($p < 0.05$). The italic cell indicates the best-performing model among those not trained on the full dataset

Pair selection	Pseudo-labeling approach	THUR		BTL		DC		WC	
		SRCC	LCC	SRCC	LCC	SRCC	LCC	SRCC	LCC
Pair balanced	✓	0.749	0.734	0.752	0.733	0.751	0.740	0.751	0.740
		0.731	0.776*	0.735	0.776*	0.735	0.782*	0.735	0.782*
MR bias		0.757	0.738	0.758	0.737	0.758	0.743	0.758	0.743
equalization	✓	<i>0.780</i>	<i>0.779*</i>	<i>0.779</i>	<i>0.778*</i>	<i>0.779</i>	<i>0.794*</i>	<i>0.779</i>	<i>0.794*</i>
MR sorted (U)	✓	0.757	0.749	0.760	0.747	0.760	0.759	0.760	0.759
		0.758	0.760	0.761	0.759	0.761	0.771	0.761	0.771
MR random (U)	✓	0.749	0.733	0.749	0.731	0.749	0.741	0.749	0.741
		0.766	0.782*	0.766	0.783*	0.766	0.784*	0.766	0.784*
Full dataset	✓	0.799	0.748	0.794	0.746	0.794	0.759	0.794	0.759
		0.681	0.788*	0.678	0.788*	0.678	0.784*	0.678	0.784*

Source(s): Authors' own work

4.4.2 Experiment results. In the following analysis, we systematically evaluate the pseudo-labeling approach and the pair selection methods considered in this work. We first conduct a detailed ablation study to assess the contributions of these components. We then broaden the analysis by comparing these configurations with a baseline trained on the full dataset. Finally, we summarize the key findings to provide practical insights.

4.4.2.1 Effect of pseudo-labeling. To assess the effect of pseudo-labeling, we conducted an ablation study by comparing models trained with and without pseudo-labeling across all pair selection methods. On the VCC2020-DACon-Pref split, the use of pseudo-labeling consistently led to statistically significant improvements regardless of the pair selection method. Specifically, we observed absolute increases of 0.07–0.09 in SRCC and 0.13–0.20 in LCC. On the VCC2020-PrefCon-Pref split, the impact was more varied: while changes in SRCC were marginal (below 0.03) and not statistically significant, LCC still showed a significant improvement of approximately 0.04 across all methods except for MR Sorted (U). These quantitative improvements highlight the core advantage of our pseudo-labeling strategy: it acts as an effective data augmentation mechanism. By leveraging the vast pool of unselected utterances, the model captures broader acoustic diversity and avoids overfitting to the sparse human-annotated subset, thereby achieving better generalization with fewer human labels.

To further examine the impact of potential noise introduced by pseudo-labeling, we analyzed the consistency between pseudo-label-induced pairwise decisions and human preferences. The results showed that pseudo labels achieve accuracy close to human consistency, indicating that the noise introduced by system-level assignment is limited. Detailed results are provided in [Appendix 3](#).

4.4.2.2 Effect of pair selection methods. To analyze the impact of different pair selection strategies, we compared the performance of MR Sorted, MR Random, Pair Balanced, and MR Bias Equalization. No statistically significant differences were observed across methods on either dataset split, regardless of whether pseudo-labeling was applied. On the VCC2020-DACon-Pref split without pseudo-labeling, the maximum performance gap between the best and worst models was small (0.02 in SRCC and 0.04 in LCC). With pseudo-labeling, these differences further decreased to 0.007 in SRCC and 0.009 in LCC. Similarly, on the VCC2020-PrefCon-Pref split, the maximum differences without pseudo-labeling remained small (0.011 in SRCC and 0.019 in LCC). Although the introduction of pseudo-labeling on this split appeared to increase the maximum gaps (up to 0.05 in SRCC and 0.024 in LCC), the relatively high variance in the results rendered these differences statistically insignificant.

While no statistically significant differences were found overall, the average performance across conditions revealed some noteworthy patterns. On the VCC2020-DACon-Pref split, without the pseudo-labeling approach, the MR Bias Equalization method yielded the best results, whereas with the pseudo-labeling approach, Pair Balanced performed best. Conversely, on the VCC2020-PrefCon-Pref split, MR Sorted (U) was the top performer without the pseudo-labeling approach, while MR Bias Equalization and MR Random (U) achieved the highest performance when the pseudo-labeling approach was used.

Although Section 4.3.2 discusses the relative effectiveness of each method in producing pseudo-derived system scores, we found no clear or consistent alignment between those earlier findings and the performance outcomes observed here, except in the case of MR Bias Equalization. While it did not significantly outperform other methods in any single setting, it consistently ranked among the top, frequently placing first or second across both dataset splits, both with and without the pseudo-labeling method. This recurring pattern suggests that MR Bias Equalization may offer greater robustness across different training scenarios.

Therefore, despite the overall lack of statistically significant differences, we consider MR Bias Equalization a generally reliable and effective choice for pair selection in practice.

4.4.2.3 Comparison with full dataset training. Finally, we compared the performance of models trained using pair selection methods combined with the pseudo-labeling approach against those trained on the full dataset. Since the pseudo-labeling approach generally improves performance, we focused on configurations that incorporate it. The model trained on the full dataset without the pseudo-labeling approach achieved the highest SRCC across both dataset splits when compared to models using pair selection methods. However, it consistently underperformed in terms of LCC relative to those models. In contrast, the model trained on the full dataset with the pseudo-labeling approach showed more competitive results. On the VCC2020-DACon-Pref split, models using pair selection methods achieved SRCC values comparable to, or even slightly better than, those of the full dataset model with the pseudo-labeling approach, although their LCC values remained slightly lower. On the VCC2020-PrefCon-Pref split, the SRCC of the full dataset model with the pseudo-labeling approach dropped significantly, allowing all pair selection methods to surpass it. In terms of LCC, some pair selection methods – particularly those using DC and WC aggregation – matched or slightly exceeded the performance of the full dataset model.

From this, we observed that although the pseudo-labeling approach benefited pair selection methods, its effectiveness was less consistent when applied to the full dataset. One possible explanation is that pseudo-labels convert inherently uncertain human preferences into deterministic pairwise comparisons. In human annotations, system pairs with similar quality often exhibit low agreement, reflecting ambiguity in relative preference, whereas pseudo-labels enforce a fixed ordering for such pairs.

Specifically, this deterministic supervision may encourage the model to learn more globally consistent ordering across systems, which in turn can lead to more stable and well-separated score assignments. This may strengthen the linear relationship between predicted and ground-truth scores, thereby benefiting LCC.

However, for ambiguous pairs, enforcing a fixed ordering may conflict with the underlying human preference distribution. Since SRCC is sensitive to local ranking errors near such decision boundaries, this mismatch could contribute to the observed degradation in SRCC. Further investigation is required to validate this hypothesis, which we consider an important direction for future work.

4.4.2.4 Summary of findings. In summary, we conclude two key points: (1) Robustness of pair selection: First, regarding the acquisition of preference scores, MR Bias Equalization proved to be the most robust strategy. Even with a limited budget of 5,000 scores, it consistently yielded stable system-level scores across varying data distributions. While it performed comparably to the strong Pair Balanced baseline on the standard VCC2020-DACon-Pref split, it demonstrated a clear advantage on the more challenging VCC2020-PrefCon-Pref split, outperforming Pair Balanced across all metrics. This confirms MR Bias Equalization as a reliable choice for selecting high-quality subsets of data. (2) Effectiveness of pseudo-labeling: Second, to effectively use these selected subsets, the pseudo-labeling approach is essential. This method directly compensates for the inherent scarcity of explicit preference scores resulting from the selection process. By deriving system-level scores to annotate individual utterances, pseudo-labeling generates dense supervision signals that enable the model to generalize effectively. Consequently, models trained with this pseudo-labeled data achieved performance comparable to those trained on the full dataset, demonstrating that high-performance learning is achievable with significantly reduced annotation costs.

5. Discussion and conclusion

5.1 Limitations

While the investigated strategies demonstrate robust performance on the Voice Conversion task using the VCC2020 dataset, their generalizability to other domains warrants further discussion.

First, the pseudo-labeling strategy assumes negligible intra-system variance by assigning a global system-level score to all its utterances. This assumption holds well for the Voice Conversion dataset used in our study (VCC2020), as it consists of clean speech recordings and strictly intra-lingual (English-to-English) conversions, which naturally constrain the intra-system variance. However, in domains with high intra-system variance, such as Text-to-Speech with complex prosodic variations or Speech Enhancement under varying signal-to-noise ratios, this method could introduce significant label noise. Second, the MR Bias Equalization algorithm inherently prioritizes the system pair with the highest bias error. While this mechanism is highly effective under normal conditions, a limitation arises in high-variance domains like TTS or SE. In such domains, certain system pairs may be inherently difficult to evaluate due to specific text prompts or extreme noise conditions. When encountering such a pair, the algorithm will repeatedly allocate listening tests to it until it hits the predefined maximum budget cap. If a dataset contains a large number of these highly variable pairs, they will act as a bottleneck. The algorithm will exhaust a significant portion of its total evaluation budget simply reaching the cap for these specific pairs, which consequently hinders and delays the overall reduction of bias across the rest of the systems.

5.2 Summary

In this study, we conducted a comprehensive investigation into online pair selection strategies for preference-based SQA. We systematically evaluated three pair selection strategies under a fixed preference budget: (i) the Pair Balanced strategy (a uniform baseline); (ii) the Merge-Rank, where we investigated both its original stochastic ranking output [MR (S)] and our proposed score aggregation method [MR (U)]; and (iii) the proposed MR Bias Equalization, which actively guides pair selection to equalize uncertainty across system pairs. Our findings can be summarized from two key perspectives: true score derivation and model training performance.

First, regarding true score derivation, we demonstrated that MR Bias Equalization consistently achieves the most robust performance. Under the same annotation budget, it exhibits greater stability compared to other methods, including the competitive Pair Balanced baseline. Furthermore, our analysis confirms that the proposed MR (U) significantly improves stability relative to the original MR (S). Conversely, the original MR (S) exhibits consistently poor correlation with ground truth, indicating that direct reliance on the raw stochastic rankings is insufficient for accurate system evaluation.

Second, concerning model training performance, we highlight the critical role of the pseudo-labeling approach. This method effectively compensates for the inherent scarcity of explicit preference scores caused by pair selection. By deriving reliable system-level scores to provide dense supervision, it enables models to generalize well even with limited data. However, its success depends strictly on the reliability of the source scores. Consequently, MR (S) variants were excluded from pseudo-label generation. Our experiments showed that in the absence of pseudo-labeling, MR Sorted (U) achieved the strongest performance. However, when pseudo-labels were introduced, MR Bias Equalization and MR Random (U) demonstrated superior stability across different dataset splits, whereas other methods exhibited larger performance fluctuations.

In conclusion, due to its consistent robustness in both accurate score derivation and downstream model training, we identify the combination of MR Bias Equalization (for selection) and GLM (for aggregation) as the most reliable strategy for preference-based SQA.

Note

- [1.] Following systems are selected for DACon as the test set: team14, team26, team18, team19, team12, team24, team20, team32, team11, and team10. Similarly, for PrefCon, following systems are selected as the test set: team01, team02, team03, team06, team08, team09, team12, team18, team19, and team28.

References

- Black, A.W. and Tokuda, K. (2005), "The blizzard challenge – 2005: evaluating corpus-based speech synthesis on common datasets", *Proc. Interspeech 2005*, pp. 722-752, doi: [10.21437/Interspeech.2005-72](https://doi.org/10.21437/Interspeech.2005-72).
- Bradley, R.A. and Terry, M.E. (1952), "Rank analysis of incomplete block designs: I. the method of paired comparisons", *Biometrika*, Vol. 39 Nos 3-4, pp. 324-345.
- Cooper, E., Huang, W.C., Tsao, Y., Wang, H.M., Toda, T. and Yamagishi, J. (2023), "The VoiceMOS challenge 2023: zero-shot subjective speech quality prediction for multiple domains", *Proc. ASRU*, pp. 1-7, doi: [10.1109/ASRU57964.2023.10389763](https://doi.org/10.1109/ASRU57964.2023.10389763).
- Delamou, M., Bazzi, A., Chafii, M. and Amhoud, E.M. (2023), "Deep learning-based estimation for multitarget radar detection", *Proc. IEEE Vehicular Technology Conference (VTC)*, pp. 1-5, doi: [10.1109/VTC2023-Spring57618.2023.10200157](https://doi.org/10.1109/VTC2023-Spring57618.2023.10200157).
- Falahatgar, M., Orlitsky, A., Pichapati, V. and Suresh, A.T. (2017), "Maximum selection and ranking under noisy comparisons", *Proc. ICML 2017*, p. 1088-96, 10, 2022, 69896-909, doi: [10.1109/ACCESS.2022.3187837](https://doi.org/10.1109/ACCESS.2022.3187837).
- Hu, Y. and Loizou, P.C. (2007), "Subjective comparison and evaluation of speech enhancement algorithms", *Speech Communication*, Vol. 49 No. 7, pp. 588-601, doi: [10.1016/j.specom.2006.12.006](https://doi.org/10.1016/j.specom.2006.12.006).
- Hu, C.H., Yasuda, Y. and Toda, T. (2023), "Preference-based training framework for automatic speech quality assessment using deep neural network", *Proc. Interspeech 2023*, pp. 546-550, doi: [10.21437/Interspeech.2023-589](https://doi.org/10.21437/Interspeech.2023-589).
- Hu, C.H., Yasuda, Y. and Toda, T. (2024), "Embedding learning for preference-based speech quality assessment", *Proc. Interspeech 2024*, pp. 2685-2689, doi: [10.21437/Interspeech.2024-1243](https://doi.org/10.21437/Interspeech.2024-1243).
- Hu, C.H., Yasuda, Y. and Toda, T. (2025), "E2EPref: an end-to-end preference-based framework for speech quality assessment to alleviate bias in direct assessment scores", *Computer Speech and Language*, Vol. 93, p. 101799, doi: [10.1016/j.csl.2025.101799](https://doi.org/10.1016/j.csl.2025.101799).
- Huang, W.C., Cooper, E., Tsao, Y., Wang, H.M., Toda, T. and Yamagishi, J. (2022), "The VoiceMOS challenge 2022", *Proc. Interspeech 2022*, pp. 4536-4540, doi: [10.21437/Interspeech.2022-970](https://doi.org/10.21437/Interspeech.2022-970).
- Huang, W.C., Cooper, E., Yamagishi, J. and Toda, T. (2022), "LDNet: unified listener dependent modeling in MOS prediction for synthetic speech", *Proc. ICASSP 2022*, pp. 896-900, doi: [10.1109/ICASSP43922.2022.9747222](https://doi.org/10.1109/ICASSP43922.2022.9747222).
- Huang, W.C., Fu, S.W., Cooper, E., Zezario, R.E., Toda, T., Wang, H.M., Yamagishi, J. and Tsao, Y. (2024), "The VoiceMOS challenge 2024: beyond speech quality prediction", *Proc. IEEE Spoken Language Technology Workshop (SLT)*, pp. 803-810, doi: [10.1109/SLT61566.2024.10832295](https://doi.org/10.1109/SLT61566.2024.10832295).
- Huang, W.C., Violeta, L.P., Liu, S., Shi, J. and Toda, T. (2023), "The singing voice conversion challenge 2023", *Proc. ASRU*, pp. 1-8, doi: [10.1109/ASRU57964.2023.10389671](https://doi.org/10.1109/ASRU57964.2023.10389671).

- Kahn, J., Lee, A. and Hannun, A. (2020), "Self-training for end-to-end speech recognition", *Proc. ICASSP 2020*, pp. 7084-7088.
- King, S. and Karaiskos, V. (2013), "The blizzard challenge", *The Blizzard Challenge 2013*, pp. 1-12, doi: [10.21437/Blizzard.2013-1](https://doi.org/10.21437/Blizzard.2013-1).
- Leng, Y., Tan, X., Zhao, S., Soong, F., Li, X.Y. and Qin, T. (2021), "MBNet: MOS prediction for synthesized speech with mean-bias network", *Proc. ICASSP 2021*, pp. 391-395, doi: [10.1109/ICASSP39728.2021.9413877](https://doi.org/10.1109/ICASSP39728.2021.9413877).
- Lo, C.C., Fu, S.W., Huang, W.C., Wang, X., Yamagishi, J., Tsao, Y. and Wang, H.M. (2019), "MOSNet: deep learning-based objective assessment for voice conversion", *Proc. Inter speech 2019*, pp. 1541-1545, doi: [10.21437/Interspeech.2019-2003](https://doi.org/10.21437/Interspeech.2019-2003).
- Lorenzo-Trueba, J., Yamagishi, J., Toda, T., Saito, D., Villavicencio, F., Kinnunen, T. and Ling, Z. (2018), "The voice conversion challenge 2018: promoting development of parallel and nonparallel methods", *Proc. The Speaker and Language Recognition Workshop (Odyssey 2018)*, pp. 195-202, doi: [10.21437/Odyssey.2018-28](https://doi.org/10.21437/Odyssey.2018-28).
- Maniati, G., Vioni, A., Ellinas, N., Nikitaras, K., Klapsas, K., Sung, J.S., Jho, G., Chalamandaris, A. and Tsiakoulis, P. (2022), "SOMOS: the Samsung open MOS dataset for the evaluation of neural text-to-speech synthesis", *Proc. Interspeech 2022*, pp. 2388-2392, doi: [10.21437/Interspeech.2022-10922](https://doi.org/10.21437/Interspeech.2022-10922).
- Manocha, P., Jin, Z. and Finkelstein, A. (2022), "SQAPP: no-reference speech quality assessment via pairwise preference", *Proc. ICASSP 2022*, pp. 891-895, doi: [10.1109/ICASSP43922.2022.9746615](https://doi.org/10.1109/ICASSP43922.2022.9746615).
- Manocha, P., Xu, B. and Kumar, A. (2021), "NORESQA: a framework for speech quality assessment using non-matching references", *Advances in Neural Information Processing Systems*, Vol. 34, pp. 22363-22378.
- Manocha, P., Jin, Z., Zhang, R. and Finkelstein, A. (2021), "CDPAM: contrastive learning for perceptual audio similarity", *Proc. ICASSP 2021*, pp. 196-200, doi: [10.1109/ICASSP39728.2021.9413711](https://doi.org/10.1109/ICASSP39728.2021.9413711).
- Manocha, P., Finkelstein, A., Zhang, R., Bryan, N.J., Mysore, G.J. and Jin, Z. (2020), "A differentiable perceptual audio metric learned from just noticeable differences", *Proc. Interspeech 2020*, pp. 2852-2856, doi: [10.21437/Interspeech.2020-1191](https://doi.org/10.21437/Interspeech.2020-1191).
- Marcus-Roberts, H.M. and Roberts, F.S. (1987), "Meaningless statistics", *Journal of Educational Statistics*, Vol. 12 No. 4, pp. 383-394, doi: [10.2307/1165056](https://doi.org/10.2307/1165056).
- Mittag, G., Naderi, B., Chehadi, A. and Möller, S. (2021), "NISQA: a deep CNN-self-attention model for multidimensional speech quality prediction with crowdsourced datasets", *Proc. Interspeech 2021*, pp. 2127-2131, doi: [10.21437/Interspeech.2021-299](https://doi.org/10.21437/Interspeech.2021-299).
- Naoumi, S., Bazzi, A., Bomfin, R. and Chafii, M. (2026), "High-resolution sensing in communication-centric ISAC: deep learning and parametric methods", *IEEE Journal on Selected Areas in Communications*, Vol. 44, pp. 2201-2216, doi: [10.1109/JSAC.2025.3639350](https://doi.org/10.1109/JSAC.2025.3639350).
- Njima, W., Bazzi, A. and Chafii, M. (2022), "DNN-based indoor localization under limited dataset using GANs and semi-supervised learning", *IEEE Access*, Vol. 10, pp. 69896-69909, doi: [10.1109/ACCESS.2022.3187837](https://doi.org/10.1109/ACCESS.2022.3187837).
- Perrotin, O., Stephenson, B., Gerber, S. and Bailly, G. (2023), "The blizzard challenge 2023", *Proc. 18th Blizzard Challenge Workshop*, pp. 1-27, doi: [10.21437/Blizzard.2023-1](https://doi.org/10.21437/Blizzard.2023-1).
- Pham, H., Dai, Z., Xie, Q. and Le, Q.V. (2021), "Meta pseudo labels", *Proc. CVPR 2021*, pp. 11557-11568.
- Shah, N.B., Balakrishnan, S., Bradley, J., Parekh, A., Ramchandran, K. and Wainwright, M. (2014), "When is it better to compare than to score?", *arXiv preprint arXiv:1406.6618*.
- Synnaeve, G., Xu, Q., Kahn, J., Likhomanenko, T., Grave, E., Pratap, V., Sriram, A., Liptchinsky, V. and Collobert, R. (2020), "End-to-end ASR: from supervised to semi-supervised learning with modern architectures", *Proc. ICML 2020 Workshop on Self-Supervision in Audio and Speech*.

- Tang, Y., Shi, J., Wu, Y. and Jin, Q. (2024), "SingMOS: an extensive opensource singing voice dataset for MOS prediction", *arXiv preprint arXiv:2406.10911*.
- Teodorescu, A.R., Moran, R. and Usher, M. (2016), "Absolutely relative or relatively absolute: violations of value invariance in human decision making", *Psychonomic Bulletin and Review*, Vol. 23 No. 1, pp. 22-38, doi: [10.3758/s13423-015-0858-8](https://doi.org/10.3758/s13423-015-0858-8).
- Thurstone, L. (1927), "A law of comparative judgement", *Psychological Review*, Vol. 34 No. 4, pp. 273-286, doi: [10.1037/h0070288](https://doi.org/10.1037/h0070288).
- Toda, T., Chen, L.H., Saito, D., Villavicencio, F., Wester, M., Wu, Z. and Yamagishi, J. (2016), "The voice conversion challenge 2016", *Proc. Interspeech 2016*, pp. 1632-1636, doi: [10.21437/Interspeech.2016-1066](https://doi.org/10.21437/Interspeech.2016-1066).
- Valentini-Botinhao, C., Ribeiro, M.S., Watts, O., Richmond, K. and Henter, G.E. (2022), "Predicting pairwise preferences between TTS audio stimuli using parallel ratings data and anti-symmetric twin neural networks", *Proc. Interspeech 2022*, pp. 471-475, doi: [10.21437/Interspeech.2022-10132](https://doi.org/10.21437/Interspeech.2022-10132).
- Venkateswaran, S., Böck, R., Kefler, T. and Krini, O. (2021), "Pseudolabelling and transfer learning based speech emotion recognition", *Studenttexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung*, pp. 19-26.
- Xie, Q., Luong, M.T., Hovy, E. and Le, Q.V. (2020), "Self-training with noisy student improves ImageNet classification", *Proc. CVPR 2020*, pp. 10687-10698, doi: [10.1109/CVPR42600.2020.01070](https://doi.org/10.1109/CVPR42600.2020.01070).
- Yasuda, Y. and Toda, T. (2024), "Automatic design optimization of preference based subjective evaluation with online learning in crowdsourcing environment", *arXiv preprint arXiv:2403.06100*.
- Yi, Z., Huang, W.C., Tian, X., Yamagishi, J., Das, R.K., Kinnunen, T., Ling, Z.H. and Toda, T. (2020), "Voice conversion challenge 2020: intra-lingual semi-parallel and cross-lingual voice conversion", *Proc. Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge*, pp. 80-98, doi: [10.21437/VCCBC.2020-14](https://doi.org/10.21437/VCCBC.2020-14).
- Zielinski, S., Hardisty, P., Hummersone, C. and Rumsey, F. (2007), "Potential biases in MUSHRA listening tests", *Proc. Audio Engineering Society Convention 123*.

Appendix 1. A Merge-Rank procedure

Algorithm 1. Merge-Rank

Require: Array A with n elements

Ensure: Approximately sorted array A

```

1: if  $n > 1$  then
2:   Split  $A$  into  $L = A[1 : \lfloor n/2 \rfloor]$ ,  $R = A[\lfloor n/2 \rfloor + 1 : n]$ 
3:   CALL Merge-Rank( $L$ )
4:   CALL Merge-Rank( $R$ )
5:   CALL Merge( $A, L, R$ )
6: end if

```

Algorithm 2. Merge

Require: Array A , subarrays L and R

Ensure: Merged and approximately sorted A

```

1:  $i, j, k \leftarrow 1, 1, 1$ 
2: while  $i \leq \text{length}(L)$  and  $j \leq \text{length}(R)$  do
3:   if COMPARE( $L[i], R[j]$ ) =  $L[i]$  then
4:      $A[k] \leftarrow L[i]$ ,  $i \leftarrow i + 1$ 
5:   else
6:      $A[k] \leftarrow R[j]$ ,  $j \leftarrow j + 1$ 
7:   end if
8:    $k \leftarrow k + 1$ 
9: end while
10: while  $i \leq \text{length}(L)$  do
11:    $A[k] \leftarrow L[i]$ ,  $i \leftarrow i + 1$ ,  $k \leftarrow k + 1$ 
12: end while
13: while  $j \leq \text{length}(R)$  do
14:    $A[k] \leftarrow R[j]$ ,  $j \leftarrow j + 1$ ,  $k \leftarrow k + 1$ 
15: end while

```

Algorithm 3. COMPARE

```

1: Initialize comparison parameters : error bias  $\epsilon$ ,
   confidence level  $\delta$ , and maximum budget  $m = \frac{1}{2\epsilon^2} \log \frac{2}{\delta}$ 

2: Initialize variables : comparison count  $r = 0$ , win count  $w = 0$ ,
   and probability  $p_{i,j}^r = \frac{1}{2}$ 

3: while comparison has not stopped do

4:   Compute confidence interval  $c = \sqrt{\frac{1}{2r} \log \frac{4r^2}{\delta}}$  if  $r > 0$ 

5:   Update error bias  $\hat{e} = c - |p_{i,j}^r - \frac{1}{2}|$  if  $r > 0$ 

6:   if  $\hat{e} < \epsilon$  or  $r \geq m$  then

7:     stop comparison

8:   endif

9:   Collect new comparison data

10:  if  $i$  wins the comparison then

11:     $w \leftarrow w + 1$ 

12:  endif

13:  Update probability  $p_{i,j}^r = \frac{w}{r}$  and increment comparison count
       $r \leftarrow r + 1$ 

14: endwhile

15: return  $i$  if  $p_{i,j}^r \geq \frac{1}{2}$  else  $j$ 

```

In this section, we detail the Merge-Rank (MR) procedure. MR adapts the classic Merge Sort algorithm to the stochastic nature of system evaluation. The procedure operates on two levels: system-pair selection (via the sorting logic) and individual pairwise comparison (via a sampling subroutine).

A.1 Merge-Rank algorithm

Similar to Merge Sort, MR uses a divide-and-conquer strategy. The algorithm recursively divides the set of systems into halves until individual systems remain, and then merges them back in sorted order. This approach avoids the exhaustive $O(N^2)$ comparisons required by a round-robin tournament. The pseudocode for the main procedure and the merge logic is outlined in Algorithms 1 and 2.

A.2 Strong stochastic transitivity

The efficiency of MR at the system-pair selection level relies on the assumption of strong stochastic transitivity. In the context of ranking, strong stochastic transitivity assumes that the winning probability propagates through intermediate comparisons. Specifically, for systems i, j, k , the probability satisfies:

$$p_{i,k} \geq \max(p_{i,j}, p_{j,k}). \quad (6)$$

This property implies that if the algorithm determines $i \succ j$ and $j \succ k$ (each with an error probability bounded by δ), the order $i \succ k$ can be inferred with the same error bound δ without performing a direct comparison between i and k . This transitive inference allows MR to prune a significant number of comparisons, particularly when systems exhibit distinct quality differences.

Appendix 2. B Probabilistic guarantees and derivations

In this section, we provide the probabilistic derivations for the adaptive stopping rules used in the COMPARE function described in Algorithm 3. The goal is to ensure that the error rate of determining the winner between any two systems i and j is bounded by δ . The function uses two stopping mechanisms: a bias-based criterion and a maximum budget constraint.

B.1 Stopping condition 1: bias-based criterion

Let r denote the number of comparisons performed so far, and let $p_{i,j}^r$ be the empirical probability that system i wins over system j . We define the confidence term c and the estimated bias $\hat{e}$ as:

$$c = \sqrt{\frac{1}{2r} \log \left(\frac{4r^2}{\delta} \right)}, \quad (7)$$

$$\hat{e} = c - \left| p_{i,j}^r - \frac{1}{2} \right|. \quad (8)$$

The algorithm terminates when $\hat{e} < \epsilon$. Substituting [equation \(8\)](#), we obtain:

$$c - \left| p_{i,j}^r - \frac{1}{2} \right| < \epsilon,$$

which is equivalent to:

$$\left| p_{i,j}^r - \frac{1}{2} \right| > c - \epsilon.$$

Suppose the empirical winner is system i , namely:

$$p_{i,j}^r > \frac{1}{2}. \quad (9)$$

Combining this with the stopping condition yields:

$$p_{i,j}^r > \frac{1}{2} + c - \epsilon. \quad (10)$$

An error occurs only if the true preference is actually in favor of system j , that is:

$$p_{i,j} \leq \frac{1}{2}. \quad (11)$$

Therefore:

$$p_{i,j}^r > \frac{1}{2} + c - \epsilon \geq p_{i,j} + c - \epsilon, \quad (12)$$

which implies:

$$p_{i,j}^r - p_{i,j} > c - \epsilon. \quad (13)$$

To apply Hoeffding's inequality, let $X_k \in \{0, 1\}$ denote the outcome of the k -th comparison, where $X_k = 1$ if system i wins over system j , and $X_k = 0$ otherwise. Then:

$$p_{i,j}^r = \frac{1}{r} \sum_{k=1}^r X_k, \quad (14)$$

with $\mathbb{E}[X_k] = p_{i,j}$. Since X_k are independent Bernoulli random variables, Hoeffding's inequality gives:

$$\Pr[p_{i,j}^r - p_{i,j} > t] \leq \exp(-2rt^2). \quad (15)$$

Setting $t = c - \epsilon$, we obtain:

$$\Pr[\text{error}] \leq \Pr[p_{i,j}^r - p_{i,j} > c - \epsilon] \leq \exp(-2r(c - \epsilon)^2). \quad (16)$$

Following the derivation provided in our methodology, we upper bound this term by $\exp(-2r(c^2))$:

$$\exp(-2r(c^2)) = \exp\left(-\log\left(\frac{4r^2}{\delta}\right)\right) = \frac{\delta}{4r^2}. \quad (17)$$

Since the stopping condition could be met at any step r , we apply Boole's inequality (union bound) over all possible values of r :

$$\Pr\left[\exists r, p_{i,j}^r < \frac{1}{2} + c - \epsilon\right] \leq \sum_{r=1}^{\infty} \frac{\delta}{4r^2} = \frac{\delta}{4} \sum_{r=1}^{\infty} \frac{1}{r^2}. \quad (18)$$

Using the Basel problem identity $\sum_{r=1}^{\infty} \frac{1}{r^2} = \frac{\pi^2}{6}$, we obtain:

$$\frac{\delta}{4} \cdot \frac{\pi^2}{6} = \frac{\pi^2}{24} \delta \leq \frac{\delta}{2}. \quad (19)$$

Thus, when the algorithm halts due to the bias-based criterion, the error rate is less than $\delta/2$.

B.2 Stopping condition 2: maximum budget constraint

To prevent infinite loops, the algorithm also enforces a maximum budget m , defined as:

$$m = \frac{1}{2\epsilon^2} \log\left(\frac{2}{\delta}\right). \quad (20)$$

If the comparison reaches $r = m$, the algorithm halts. By Chernoff's bound, the error probability at this limit is:

$$\Pr[p_{i,j}^m < 1/2] \leq \exp(-2m\epsilon^2). \quad (21)$$

Substituting m :

$$\exp\left(-2 \cdot \left(\frac{1}{2\epsilon^2} \log\frac{2}{\delta}\right) \cdot \epsilon^2\right) = \exp\left(-\log\frac{2}{\delta}\right) = \frac{\delta}{2}. \quad (22)$$

Thus, when meeting the budget constraint, the error rate is bounded by $\delta/2$.

B.3 Conclusion

Combining both mechanisms, whether the COMPARE function halts due to the bias threshold ($\hat{\epsilon} < \epsilon$) or the budget constraint ($r = m$), the probability of correctly identifying the winner is maintained such that the error rate is consistently less than $\delta/2$.

Appendix 3. C Pseudo-label quality analysis

C.1 Experimental setup and evaluation protocol

We conduct the analysis on the training sets of VCC2020-DACon and VCC2020-PrefCon. Pseudo labels are constructed following the same procedure as in the model training experiments described in Section 4.4.1. Specifically, we consider the same set of pair selection methods. For each method, we sample approximately 5,000 preference scores to derive system-level quality scores.

The aggregation strategies are consistent with those used in Section 4.4.1. In particular, we apply the DC aggregation method for the pair balanced method and the full dataset setting, while using the BTL aggregation method for the MR variants. These aggregated system-level scores are then assigned to all utterances of each system to construct pseudo labels.

C.2 Evaluation protocol and human consistency

To evaluate pseudo-label quality, we compare pseudo-label-induced pairwise decisions with ground-truth human preference labels. For each preference instance, the two utterances correspond to two systems, each associated with a pseudo-label (i.e. system-level score). By comparing these scores, we derive a pairwise preference decision in the pseudo-label space, which is then compared with the human annotation. The accuracy is computed as the proportion of agreements between the two.

To ensure a fair comparison, we aim to estimate human consistency under the same decision framework. Ideally, this would require identifying a globally optimal system-level ranking that maximizes agreement with all human preference labels. However, this is computationally intractable due to the factorial number of possible rankings (e.g. $22!$ for 22 systems).

Instead, we approximate this upper bound by computing system pair-level human consistency. Specifically, for each system pair, we aggregate all human preference annotations involving that pair and treat the majority vote as the ground-truth label. We then measure how often individual human annotations agree with this consensus.

This formulation aligns the evaluation of pseudo labels and human consistency in the same pairwise decision space, while providing a tractable and meaningful approximation of the upper bound imposed by human agreement.

C.3 Results and discussion

As shown in Table A1, pseudo-label-based decisions derived from different pair selection methods achieve accuracy close to human consistency, even when using only 5,000 preference scores (averaged over 100 simulations). Increasing the data size to the full dataset yields only marginal improvements, indicating that reliable pseudo labels can already be obtained from relatively small subsets.

Across all settings, the gap between pseudo-label accuracy and human consistency remains small. Notably, human consistency itself is relatively modest (0.6288 on VCC2020-DACon and 0.6455 on VCC2020-PrefCon), reflecting inherent variability in human preference annotations. This suggests that most of the residual disagreement is due to intrinsic human inconsistency rather than errors introduced by pseudo labeling.

Overall, these results indicate that human agreement imposes a practical upper bound on performance, which the proposed pseudo-labeling strategy already closely approaches, supporting its reliability for training.

Table A1. Comparison of pseudo-label-induced pairwise accuracy and human consistency on VCC2020-DACon and VCC2020-PrefCon

Dataset	Pair balanced	MR bias equalization	MR sorted	MR random	Full dataset	Human consistency
VCC2020-DACon	0.6212	0.6252	0.6261	0.6252	0.6277	0.6288
VCC2020-PrefCon	0.6419	0.6419	0.6426	0.6423	0.6449	0.6455

Source(s): Authors' own work

Corresponding author

ChengHung Hu can be contacted at: hu.chenghung@g.sp.m.is.nagoya-u.ac.jp