Probabilistic Models of Speech Quality

Michael Chinen

College of Advanced Science and Technology

Tokyo Denki University

A thesis presented for the degree of

Doctor of Philosophy

November 2021

Acknowledgements

I would like to express my appreciation and gratitude to Professor Naotoshi Osaka at Tokyo Denki University, for his constant encouragement, discussions, and advice. Professor Osaka introduced me to the field of research in sound synthesis in 2006 when I was a research student in his laboratory, and has provided me with guidance ever since.

I would also like to thank Jan Skoglund, my manager at Google for his support of speech quality research, as well as for providing various important discussions and advice on the matter.

I also would like to thank Professor Andrew Hines at University College Dublin for his collaboration and advice upon his foundational work that much of this thesis is built upon.

I would also like to thank my colleagues at Google for their valuable discussions and inspiration.

Finally, I would like to thank my friends and family for their support.

Abstract

Speech quality estimation is the process of estimating how human listeners respond to speech. This is often implemented by designing a relevant perceptual model and fitting the model to the datasets containing speech and quality ratings obtained through subjective tests. These quality models are used in various contexts, including speech codec evaluation and development, as well as regression testing for online systems such as Google Meet. The production environment has substantially different requirements from those of the research context, in terms of the speech data, as well as stability and efficiency. Production-driven adjustments are often needed. Changes to ViSQOL (an open source speech quality estimation framework) in this context are discussed.

Intrusive subjective speech quality estimation of mean opinion score (MOS) often involves mapping a raw similarity score extracted from differences between the clean and degraded utterance onto MOS with a fitted mapping function. A multi-dimensional mapping function using deep lattice networks (DLNs) to provide monotonic constraints with input features provided by ViSQOL is presented. The DLN improved the speech mapping to 0.24 mean-squared error (MSE) on a mixture of datasets that include voice over IP (VoIP) and codec degradations, outperforming the 1-D fitted functions, SVR, as well as PESQ and POLQA. Additionally, we show that the DLN can be used to learn a quantile function that is well calibrated and a useful measure of uncertainty.

Identifying the relevant predictors that can be used to estimate quality is a non-trivial task. Often, a quality model is fit to a dataset that contains predictors that are related only to the speech waveform. Other metadata such as the rating statistics of each individual rater, as well as the language of the rater may also be relevant. The majority of speech quality models use deterministic or frequentist approaches for estimation. We propose to model speech quality as a process with uncertainty that has certain priors in a

Bayesian context. The models that included language and/or rater obtained significantly lower errors (0.601 versus 0.684 root-mean-square error (RMSE)) and higher correlation than those that did not. Additionally, individual rater models matched or exceeded the performance of MOS models.

Speech quality estimation will continue to evolve due to the continuous development of speech synthesis models, as well as the cultural reception and trends of natural speech. The immediate future looks to combine un- and semi-supervised training to satisfy the data-hungry models of deep learning that have shown immense promise in virtually every other speech domain.

Keywords— Speech Quality Estimation - Bayesian - Mean Opinion Score - Language

Table of Contents

1 | Introduction

1.1 Motivation

Communication is the transfer of information between separate entities. Communication occurs over distances that are very large, such as the information about interstellar space sent back to us from the Voyager 2 space probe which is currently some 19.1 billion kilometers away from earth, to the microscopic scale, such as the signaling molecules that change the behaviors of cells. The Voyager example is clearly man-made communication that exists due to technological progress. The cell example has naturally arisen as a product of evolution. Speech communication is interesting in that it has aspects of it that are man-made and aspects that are natural, and others that are hard to classify in a binary fashion. For example, while speech is one of the oldest and primary forms of communication between people, the telephone, television, and computer has resulted in an increasingly large number of humans communicating with speech at vast distances in a tightly connected manner that has never been possible until recently.

Humans are not the only consumers of speech. In today's technological landscape with deep learning, machine listening consumes arguably more speech than humans due to the simple fact that machines can process speech at speeds much faster than humans can. For an increasing number of problems, like chess, go, image classification, and speech recognition, machine learning is able to reliably outperform humans. However, quality estimation is arguably in a different category from these because an objective physical target, subjective human behavior itself is the target that is being modeled. One interpretation of this is that quality models will never be able surpass human performance. A consequence of this is that the problem itself is less well-defined, and the term 'quality' itself is subject

to many different interpretations.

What exactly is meant by 'speech quality'? It does not refer to any standard objective signal metric such as signal-to-noise ratio (SNR), because there are many signals that have very low SNR but still will be perceived as 'sounding good'. For example, flipping the polarity of a signal will produce a signal with a very low SNR, but will sound identical to humans. Typically, it refers to how humans perceive and react to a speech utterance. If it sounds 'good', then it is usually easy to understand, without significant artifacts or noise. If it sounds 'bad' then it may be unintelligible. Intelligibility is only one factor of many with regards to quality. Perception is dependent on the complex physical and psychological hearing system that humans have. Science and research has developed some understanding of this system, but it is still unknown how exactly a human will react to a given speech utterance unless they are directly observed and surveyed.

The most common method of surveying humans on speech quality is known as opinion score testing, which asks raters to assign a numeric value indicating increasing levels of quality, usually given speech from some application, for example, from network transmission [1]. For a given speech signal, there is significant variation and uncertainty in the scores that can be observed from these surveys. Often, researchers are interested in the average of these scores, known as the mean opinion score (MOS). MOS is often compared between classes of signals to show how one class compares to another. For example, when designing a speech codec like Lyra or Opus, the MOS values for these two systems will be compared to show that one has equal or better performance overall compared to the other. This kind of application of MOS is useful for virtually all speech synthesis domains.

Designing a proper experiment and preparing the data for the test is a complicated matter. An even larger problem is that gathering human listening subjects

for these types of tests is expensive and time consuming. The proliferation of crowdsourcing-based testing has lowered the hurdle to this, but the costs of gathering the data still remain high compared to other domains such as speech or image recognition where the labels are often freely available.

What tools are available to speech and audio researchers and engineers to measure quality that cannot afford to conduct an expensive subjective test at each stage of research and development? There are numerous objective metrics available, i.e, metrics obtained by measurements on the audio signal, to assess the quality of recorded audio clips. Examples of physical measurements include signal-to-noise ratio (SNR), total harmonic distortion (THD), and spectral (magnitude) distortion. When estimating perceived quality, PESQ [2, 3] and POLQA [4, 5] have become standards for measuring speech quality. There are other notable examples, e.g., PEAQ [6] and PEMO-Q [7]. Most of these metrics require commercial licenses. ViSQOL [8] and ViSQOLAudio [9] (referred to collectively as ViSQOL below), are freely available alternatives for speech and audio. These metrics are continually being expanded to cover additional domains. For example the work on AMBIQUAL [10] extends the same principles used in ViSQOLAudio into the ambisonics domain.

Are these existing options good enough? There is always room for improvement. However, the target is always moving. Speech-related technologies have advanced significantly in the past few years. We have been able to record and resynthesize speech at a near-transparent level for quite some time. Speech coding, which compresses information for efficient transmission of speech, has produced increasingly efficient results due to advancements in deep learning such as LPCNet [11], Lyra [12, 13], SoundStream [14], and the HuBERT-based model [15] that can bring the bitrate down to just 365 bits per second, which is an order of magnitude less than what was possible just a few years ago. Another application

is speech enhancement [16], which processes audio to restore quality and intelligibility. Yet another is packet loss concealment [17], which attempts to recover gracefully from speech transmissions that are missing a piece of information due to network issues. All of these applications strive to obtain resynthesized speech of a higher quality. As such, it may be worthwhile to analyze the performance of these objective quality estimators for these new systems. For example, the generative models in particular are problematic for existing full reference speech quality metrics due tendency to produce small time and pitch differences.

1.2 Aim and Objectives

This thesis attempts to describe how speech quality estimation systems are designed, and proposes improvements that can be made. The author is the primary maintainer of the github repository for ViSQOL, a speech quality framework which will be described in detail and used as a reference point for these improvements.

In intrusive contexts, where a reference speech utterance is compared to a test utterance, speech quality and similarity have a monotonic relationship. This constraint is straightforward to preserve in the one-dimensional context, but the problem of maintaining monotonic relationships between the predicted MOS and high-dimensional similarity vectors is more difficult. Chapter 3 discusses this problem and proposes a solution that uses deep learning.

There is no such thing as a perfect metric, because the metric depends on the application and context. Depending on the context, there may be various problems with using MOS models. How can these problems be addressed? First, MOS models are often too reductive for certain applications. That is, it is not diagnostic - MOS models predict overall quality but do not inform the user on which specific

aspects of the quality are good or bad, which limits the user from being able to diagnose what the cause of the low quality is. Next, the mean opinion does not describe the distribution of scores, but rather discards various important information such as variance and individual rater information. Additionally, when conducting a large subjective test, MOS is often aggregated over results from test subjects of different cultures and languages, which each have their own biases that are not directly comparable. This problem is discussed extensively in Chapter 4.

1.3 Thesis Outline

The remainder of this report is organized as follows:

Chapter 2 — introduces ViSQOL, a popular speech quality estimation framework.

Chapter 3 — presents a probabilistic model of speech quality using deep lattice networks.

Chapter 4 — considers the relationship between the subjective test metadata (cultural, rater, and language information) and the quality scores reported.

Chapter 5 — summarizes and concludes the thesis, and describes future directions.

2 | ViSQOL v3: An Open Source Production Ready Objective Speech and Audio Metric[1]

2.1 Introduction

ViSQOL is a full-reference mean opinion score (MOS) estimator for speech and audio. ViSQOL uses traditional signal processing methods to extract a similarity score that is mapped to MOS. It was originally designed with a polynomial mapping of the neurogram similarity index measure (NSIM) [19] to MOS, and later, ViSQOLAudio was developed by extending ViSQOL to use a model trained for support vector regression on audio.

This chapter describes the C++ version of ViSQOL, v3 (conformance version 310), which is the most recent version on GitHub as of the time of publishing. Compared to previous Matlab v2 versions, ViSQOL v3 contains incremental improvements to the existing framework based on real-world feedback, rather than fundamental changes such as end-to-end DNN modeling. Since ViSQOL has been presented and benchmarked in a large number of experiments that have validated its application to a number of use cases [8, 20, 21, 9, 22, 23] we consider it relatively well analyzed for the known datasets, which tend to be smaller and relatively homogeneous. We instead turn our attention to the data and types of problems encountered "in the wild" at Google teams that were independent of ViSQOL development, and the iterative improvements that have come from this

[1]The content from this chapter was originally published and presented at the International Conference on Quality of Multimedia Experience (QoMEX) 2020 with co-authors Felicia S. C. Lim, Jan Skoglund, Nikita Gureev, Feargus O'Gorman, and Andrew Hines [18].

analysis. Adapting it to these cases has yielded various improvements to usability and performance, along with feedback and insights about the design of future systems for estimating perceptual quality. Since the nature of these improvements fill in the 'blind spots' of the datasets, they are not expected to improve its results on these datasets. Until there is the creation of more diverse subjective score datasets, real-world validation seems to be a reasonable compromise.

Alongside improving the quality of MOS estimation from real-world data, we are concerned with how to make ViSQOL more useful to the community from a practical tooling perspective. Even though ViSQOL was available through a MATLAB implementation, there were still unnecessary hurdles to use it in certain cases, e.g. production and continuous integration testing, (which may need to run on a server), or may not have MATLAB licenses available. As a result, we chose to re-implement it in C++ because it is a widely available and extensible language that can be wrapped in other languages, such as python. The code was released on GitHub for ease of access and to obtain feedback and contributions from the community.

This chapter is structured as follows: In section 2.2, we present the general design and algorithmic improvements that are in the new version. In section 2.3, a case study of the findings and challenges encountered when integrating ViSQOL into various Google projects. Then in section 2.4, the improvements with respect to the case studies are discussed. Finally, we summarize in a concluding section 2.5.

2.2 Design and improvements

This section summarizes the previous version and describes the changes made to the new version. Figure 2.1 shows the overall program flow and highlights the new components of the system that are referred to in the subsections.

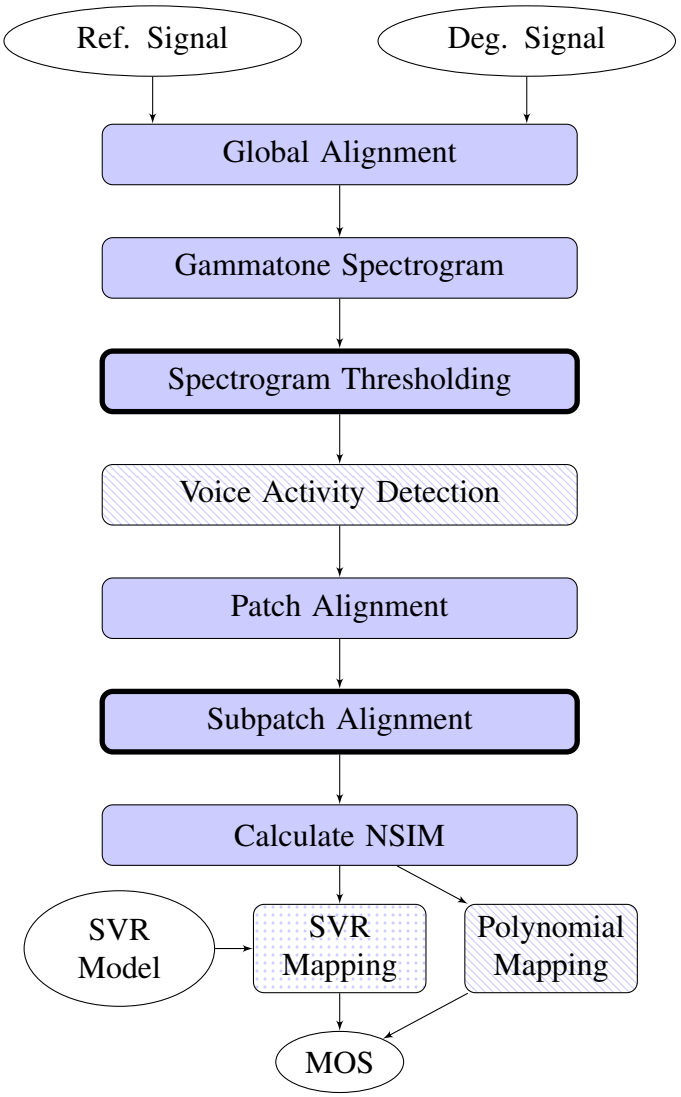


Figure 2.1: System Diagram. The inputs and outputs have white fill, and the processing components have blue fill. New components have thick edges. The dashed and dotted fill represent speech-only and audio-only components, respectively.

2.2.1 Signal processing modules

The ViSQOL algorithms described in [8] and [9] share many components by design with v3, such as the gammatone spectrogram and NSIM calculation. It then seems reasonable that the common components be shared and developed together. The differences between the two algorithms are related to differences in the characteristics of speech and music. For example, the use of voice activity detection (VAD) for speech, and analysis of the higher bands (up to 24 kHz) for general audio/music. The common components of both speech and audio systems include creating a gammatone spectrogram using equivalent rectangular bandwidth (ERB) filters, creation of patches on the order of a half-second, aligning them, computing the NSIM from the aligned patches, and then mapping the NSIM values to MOS.

There were minor changes to some of these components in ViSQOL v3 because of practical reasons, such as modifying dependencies, or fixing issues found in case studies or test failures. For example, the VAD implementation uses a simple energy-based VAD, which should be sufficient given the requirement of clean references. As another example, window sizes were updated to be 80 ms with a hop of 20 ms after discovering an issue with the windowing of previous versions.

The common individual processing modules for ViSQOL's speech mode are described here, in order. Additions in v3 are described later.

2.2.1.1 Global alignment and normalization

The first step normalizes the energy of the input reference and degraded signals. A global alignment is conducted next, using cross correlation to shift the degraded signal by the computed lag.

2.2.1.2 Gammatone spectrogram

A 21-band gammatone filterbank, which can be viewed as a coarse model of the human auditory system, is used to create a spectrogram for both of the aligned reference and degraded signals. The frame length is 80 milliseconds, with a hop length of 20 milliseconds.

2.2.1.3 Voice activity detection

A RMS energy-based voice activity detection algorithm is used on the reference signal to identify the frames of the spectrogram that are likely to have voice activity. The reference is assumed to be clean, which allows a relatively simple threshold and gate-based algorithm to identify active regions.

2.2.1.4 Patch alignment

The next step creates 400 millisecond 'patches' from the reference signal, wherever there is voice activity. A dynamic time warping algorithm is used on the degraded signal to create corresponding degraded patches for each reference patch. These patch pairs will be used to compute similarity. This is necessary because the global alignment step that was initially done does not handle local shifts in time, which can happen because of packet loss and jitter, or because of generative codecs like WaveNet that can shift the onset of each phoneme by some amount.

2.2.1.5 NSIM calculation

The neurogram similarity index measure (NSIM) is computed over each spectrogram. As the name suggests, NSIM was originally used on neurograms, but has been shown to be useful for indicating similarity for speech signals when applied on an audio spectrogram [24]. The output of the NSIM calculation is another

spectrogram that indicates similarity between the aligned patches. These spectrograms are aggregated over time by taking the mean, producing 21 NSIM values for the 21 frequency bands. Depending on the subsequent mapping function, they can also be aggregated over frequency bands to produce a single scalar.

2.2.1.6 NSIM to MOS mapping

Finally NSIM is mapped to MOS using a function that has been fit to subjective data. Previous versions of ViSQOL used a third order polynomial function, and v3 uses an exponential function.

2.2.2 Constraints and Usage Recommendations

ViSQOL was originally designed for speech codec and VoIP use cases, which has produced a system that has certain constraints. The general constraints are illustrated in table 2.1. Many of the constraints come from the historical use case of speech coding and VoIP, as well as the conventions for subjective tests (e.g. the recommendation of a 5 to 10 seconds length for input utterances). At the same time, a number of the recommendations are arbitrary or chosen based on design principles (e.g. a 0.4 second patch length was chosen to have patches contain at least a few phonemes). As the number of applications grows, ViSQOL is used outside of these recommendations as well, with varying success.

The parameters and recommendations were informed by a number of factors, including the ITU-T P.800 absolute categorical test procedure recommendations [1], as well as conventional and historical choices. For example, the P.800 recommendation for presenting two five second utterances means the MOS datasets that ViSQOL is trained on most frequently contain input that is 5 or 10 seconds in length (with some tests omitting the second utterance). The reference speech ut-

Table 2.1: Parameters and Recommendations for ViSQOL speech mode.

Parameter	Value
Recommended Utterance Duration	5-10 seconds
Clean Reference Required	Yes
Maximum Misalignment	2 seconds
Sample Rate	16000 Hz
Spectrogram Kernel	Gammatone Filter
Spectrogram Frame Length	80 milliseconds
Spectrogram Hop Length	20 milliseconds
Voice Activity Detection	RMS energy threshold
Patch Duration	0.4 seconds

terance should be relatively clean and correspond to the degraded utterance. The input should be 16000 Hz for both narrow and wideband speech use cases.

2.2.3 C++ Library and Binary

To make ViSQOL more available, we uncoupled the dependency on MATLAB by implementing a C++ version with only open source dependencies. The new version, v3, is available as a binary or as a library. The codebase was made available on GitHub because we wish for it to be easy to use by the public, and to invite external contributions.

The majority of users were binary users, but some had requirements for finer control. For this purpose we designed a library with protobuf support and error checking, which the binary depends on. This library would also be useful for a user that wishes to wrap the functions in a different language, such as with python bindings.

There were several changes to the input and output. Verbose output has also changed to include the average NSIM values per frequency band and mean NSIM per frame. Because ViSQOL is continuously changing to adapt to new problems,

a conformance version number is included in the output. Whenever the MOS changes for known files, the conformance number will be incremented. Lastly, batch processing via comma-separated value (csv) files are also supported.

A number of Google-related projects were used to build this version. The application binary was implemented using the Abseil C++ application framework [25]. The Google Test C++ testing framework [26] was integrated and various tests were implemented to ensure correctness, detect regressions, and increase stability for edge cases. 23 test classes with multiple tests were implemented. These include not only unit tests, but also a test to check the conformance of the current version to known scores. The Bazel framework [27] was used to handle building and dependency fetching, as well as test development.

2.2.4 Fine-scaled Time Alignment

Although the previous versions of ViSQOL did two levels of alignment (global and patch), there were still issues with the patch alignment due to the spectrogram frames being misaligned at a fine scale. To address this, we implemented an additional alignment step that offsets by the lag found in a cross correlation step on the time-domain regions that corresponds to the aligned patches as described in [28]. Next, the gammatone spectrogram is recomputed for sample-aligned patch audio and the NSIM score is taken.

2.2.5 Silence Thresholds

To deal with the problem of log-scale amplitudes discussed in 2.3.3, we introduce silence thresholds on the gammatone spectrogram. Because NSIM is calculated on log-amplitudes, we found that it was too sensitive to different levels of ambient noise. For example, a near-digital silence reference compared against a very low

level of ambient noise would still have a very low NSIM score, despite being perceptually transparent. The silence threshold introduces an absolute floor as well as a relative floor that may be higher for high amplitude frames.

The thresholded amplitude $y_{t,f}(x)$ for a time t and frequency band f given an input spectrogram x is subject to:

$$y_{t,f}(x) = \max(Y_{min}, Y_{fmin}(t), x(t, f)) \tag{2.1}$$

where:

$$Y_{fmin}(t) = \max(r_f(t), d_f(t)) - Y_{fmin} \tag{2.2}$$

given reference and degraded log amplitudes $r_{t,f}$ and $d_{t,f}$, and global absolute threshold Y_{min}, and relative per-frame threshold Y_{fmin}.

2.2.6 NSIM to MOS Model

The changes above ultimately affect the NSIM scores. This requires that a new SVR model is trained to map the frequency band NSIM to MOS using libsvm [29]. We conducted a grid search to minimize the 4-way cross validation loss on the same training set (TCDAudio14, CoreSV14, AACvOpus15). However, we observed in Section 2.3 that this model was too specific to the training data and would behave poorly on very low bitrate (6-18 kbps) audio. This appears to be related to the fact that there is no monotonicity constraint in the SVR model used by ViSQOL (a strictly higher NSIM for out of distribution data produced lower MOS). To address this issue for the default model, we relaxed the SVR parameters by lowering the cost and gamma parameters to have a slightly higher cross validation error while providing behavior that was closer to monotonic behavior.

Additionally, this version includes some tooling and documentation that allows

for users to train their own SVR model by the use of CSV input files if the user can provide subjective scores for degraded/reference pairs. By following the grid search methods described by libsvm authors, users should be able to tailor a model that is able to represent their data.

2.3 Case Studies and User Feedback

This version of ViSQOL is the result of the integration process of ViSQOL, using real production and integration testing cases at Google. The case studies described in this section were initiated by individual teams that were independent of prior ViSQOL development. They typically consulted with a ViSQOL developer to verify appropriate usage, or read the documentation and integrated ViSQOL on their own.

2.3.1 Hangouts Meet

The Meet team has been successfully using ViSQOL for assessing audio quality in Hangouts Meet. Hangouts Meet is a video communication service that uses WebRTC [30] for transmitting audio. Meet uses a testbed that is able to reliably replicate adverse network conditions to assess the quality of audio during the call. For this use case they have 48 kHz-sampled reference and degraded audio samples and use ViSQOLAudio for calculating the results.

In order to ensure that ViSQOL works reliably for this use case, it was compared to an internal no-reference audio quality metric that is based on technical metrics of a WebRTC-based receiver. The metric is on a scale from 0 to 1, with lower scores being better. ViSQOL's MOS is able to correlate to this metric, as seen in Figure 2.2.

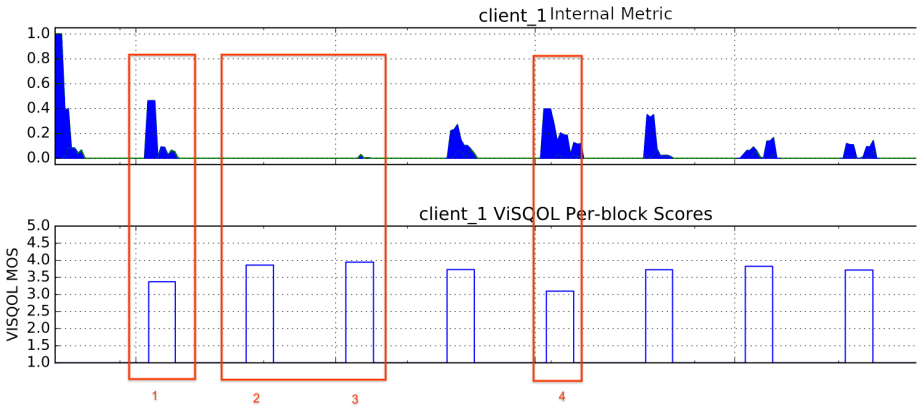


Figure 2.2: Hangouts Meet's internal no-reference metric has components to detect audio degradations. ViSQOL successfully detected these degradations in audio that contained them (blocks 1 and 4), while in the audio blocks that were not affected the scores from ViSQOL were higher (blocks 2 and 3).

In this use case, Meet developers were mostly interested in the sensitivity of ViSQOL to audio degradations from network impairments. In Figure 2.3 there is a comparison between mean ViSQOL scores during a call that shows that the metric is sensitive to how audio quality changed from a good network conditions scenario with scores ranging from 4.21 to 4.28, to a medium impaired scenario with scores ranging from 4.04 to 4.16, to finally an extremely challenging network scenario with scores from 3.72 to 3.94. Although the exact network conditions can not be shared, here good network conditions indicated that the connection should allow for both video and audio to be near perfect in the call, medium conditions indicate that the call might have issues, but the audio should continue to be good, while in extremely challenging conditions we expect to see both video and audio perceptually degraded, but the call would still go through.

In order to ensure that ViSQOL performs reliably, several hundreds of calls were collected from the testbed. The mean values obtained from ViSQOL and the internal metric from these calls were plotted in Figure 2.4. The results were reliably reproduced. Following the positive results from this investigation, ViSQOL is cur-

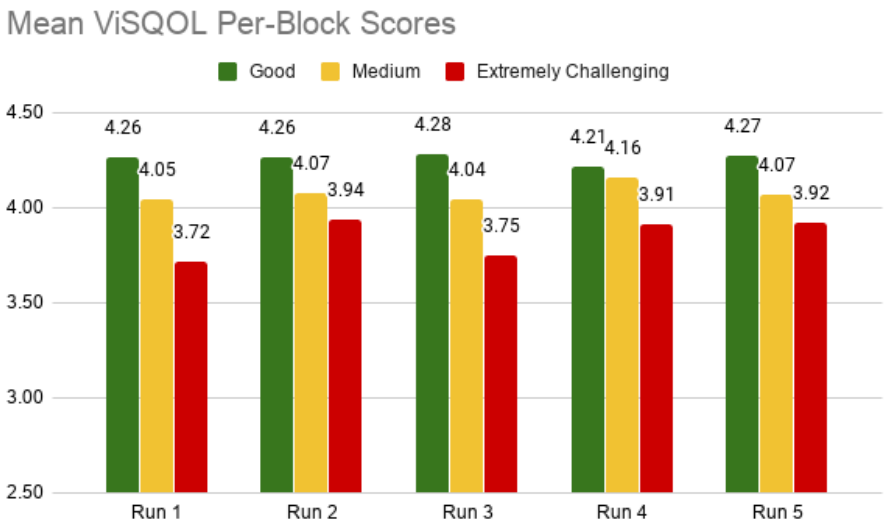


Figure 2.3: Comparison between mean ViSQOL MOS and network degradations. Some of the calls were run with good network conditions (green), some were simulating average network conditions, where the product should still perform well (yellow), while others were simulating extremely challenging network conditions, where it is expected to for issues to appear (red).

rently one of the main objective audio quality metrics deployed by the Hangouts Meet product team at Google.

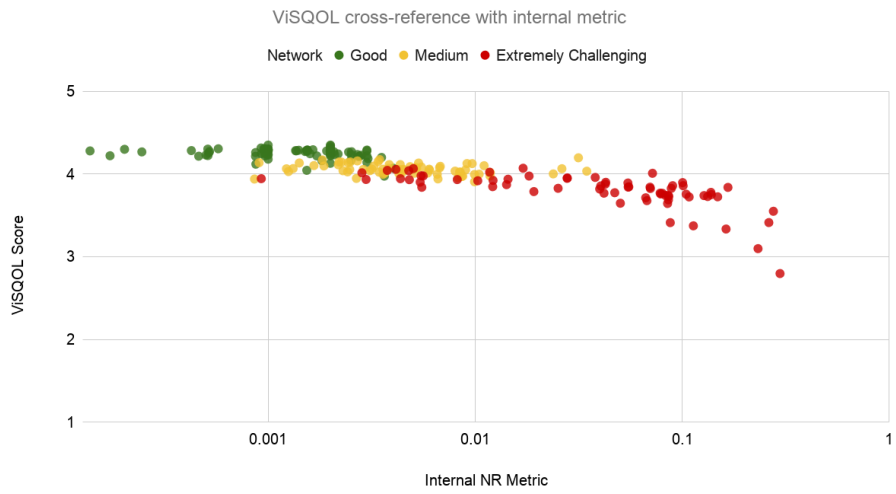


Figure 2.4: Scatter plot of ViSQOL MOS versus a no-reference internal metric. Each point represents a call.

2.3.2 Opus Codec

Google contributes to the development of the Opus codec. ViSQOL and POLQA were used to benchmark the quality of the Opus coder for both speech and music at various bitrates and computational complexities. In previous studies ViSQOLAudio has been shown to perform reasonably on low bitrate audio [9]. However, ViSQOL's speech mode did not specifically target the low bitrate case. Additionally, recent advancements in Opus have pushed the lower bound of the range of bitrates further downwards for a given bandwidth since the time ViSQOL was introduced. For example, Opus 1.3 can produce a wideband signal at 9 kbps, whereas the TCDAudio14 [31], CoreSV14 [32], and AACvOpus15 [23] datasets that ViSQOL's support vector regression was trained on have bitrates that only go as low as 24 kbps.

POLQA and the original version of ViSQOL in speech mode display similar trends that are consistent with expectations with respect to the bitrate and complexity settings. The differences in the *lower* bitrates are more pronounced according to POLQA. The differences in *higher* bitrates are more pronounced according to ViSQOL. Although subjective scores were not available, the developers expected that MOS should be less sensitive to changes in higher bitrates, giving POLQA a better match. After the improvements described in section 3, ViSQOL v3 MOS was a closer match to the expectation as can be seen in Figure 2.5.

For musical examples, the developers found that both metrics display similar trends with respect to bitrates. However, POLQA shows higher discrimination between 6-8 kbps, 10-12 kbps and 16-24 kbps. ViSQOL is able to discriminate between the different bitrates with monotonic behavior, but one point of concern is that this results in ViSQOLAudio being relatively insensitive to differences in complexity settings. In light of this, we would not recommend using ViSQOL for automated regression tests without retraining the model. The improvements made in section 3 slightly ameliorate these issues, as can be seen in Figure 2.6. On the other hand, ViSQOL identified a spurious bandwidth 'bump' at 12 kbps for the 5 and 6 complexity settings (which was perceived as higher quality in informal listening), where POLQA did not.

Lastly, ViSQOL was used to analyze the results for both clean and noisy references. This is not a case ViSQOL was designed for, as it presumes a clean reference, similar to PESQ [3] and POLQA [5]. However, it was found to perform in a similar fashion to the clean cases for both speech and audio in the noisy cases.

It was concluded that ViSQOL could be used for regression testing for speech. However, formal listening tests would be desirable for two reasons: to better interpret the differences between POLQA and ViSQOLAudio, and to allow training

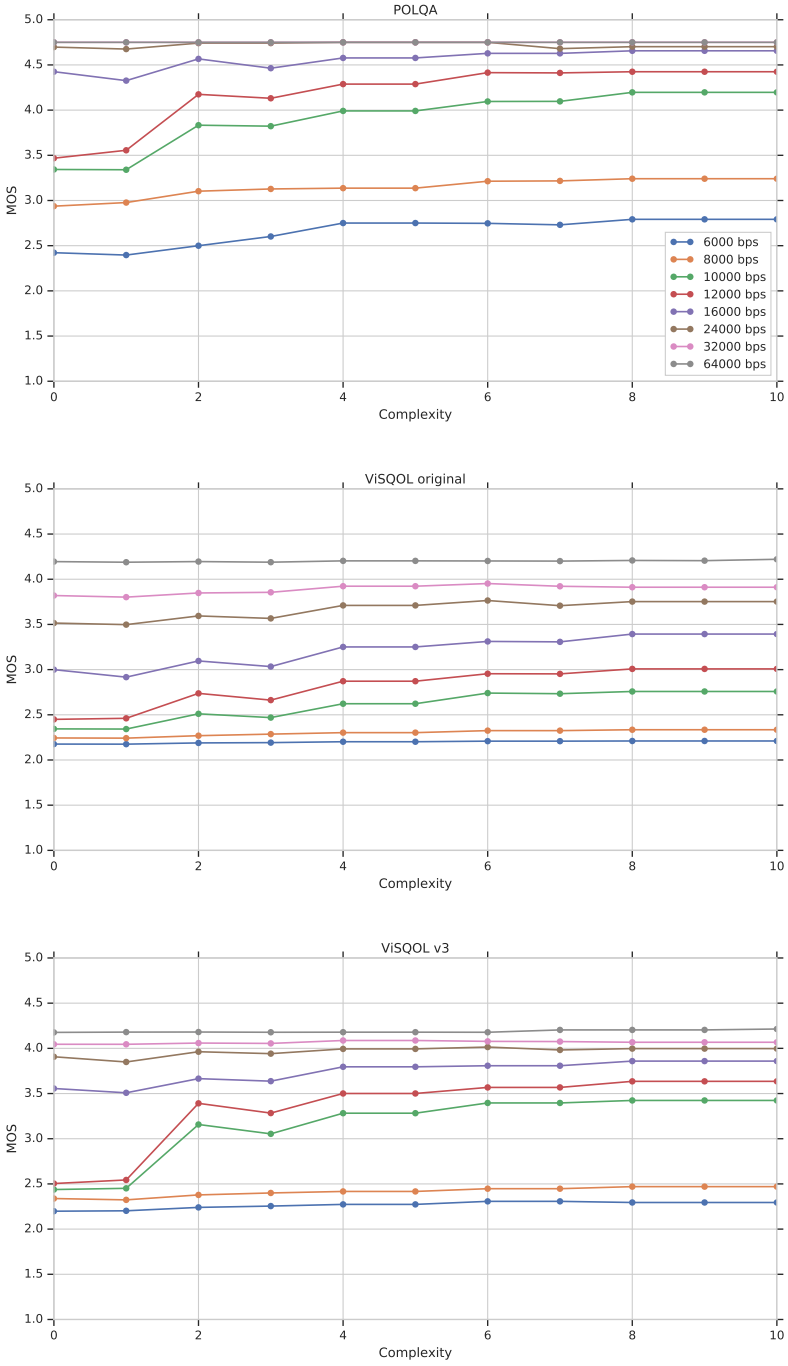


Figure 2.5: Estimated MOS for varying bitrates (6-64 kbps) and complexity settings on Opus-encoded speech.

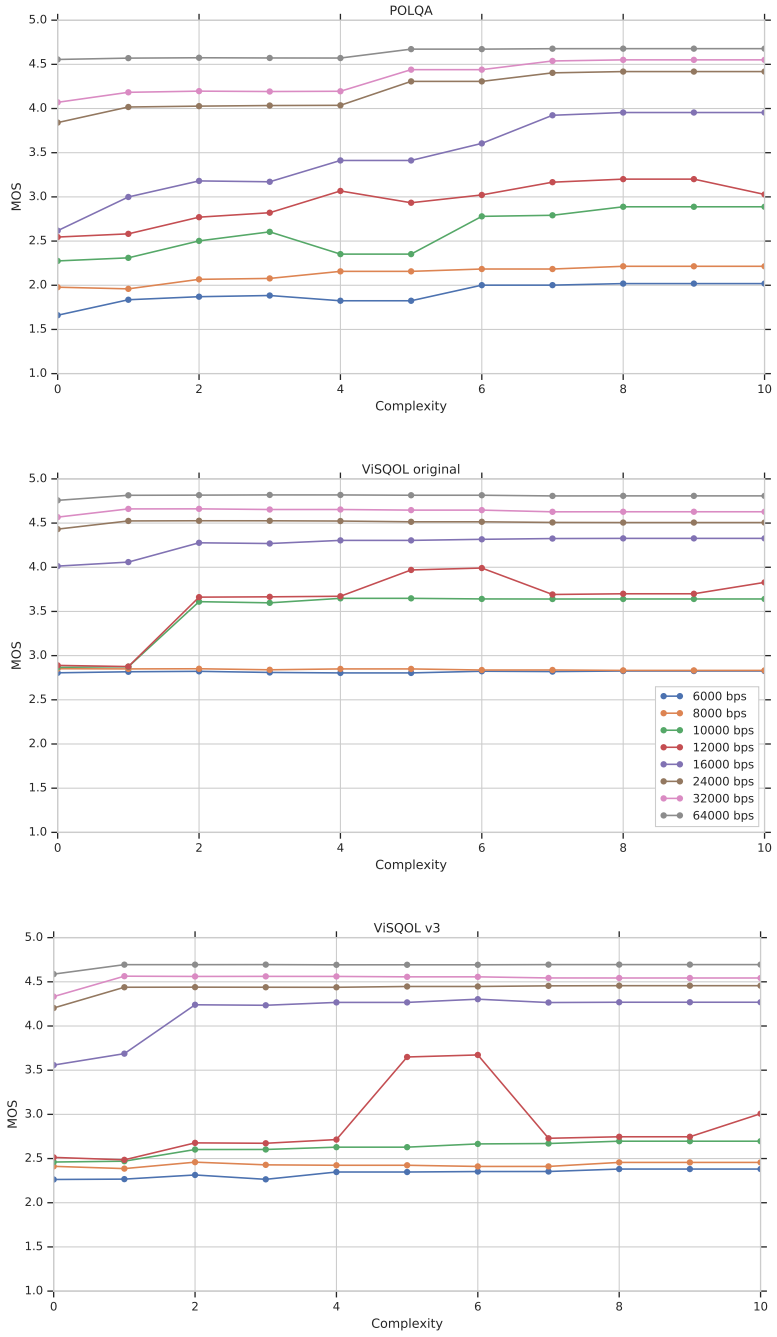


Figure 2.6: Estimated MOS for varying bitrates and complexity settings on Opus-encoded music (legend as per Fig. 2.5). The bitrates follow the same key as Figure 2.5. The bump at complexity 5, 6, and 10 for 12 kbps is related to Opus deciding to use a 12 kHz bandwidth for some fraction of the files instead of the 8 kHz bandwidth it used for complexities 2-4 and 7-9.

a model that represented the low bitrate ranges.

2.3.3 Other Findings

A number of other teams have also adapted ViSQOL for their products. In the majority of cases, their use case vaguely resembles the training data (e.g. wideband speech network degradations or music coding), but often has marked differences. For example, one team chose to analyze the network loop with a digital and analog interface, requiring a rig to be built for continuous automated testing. Typically these teams also had access to PESQ, POLQA or subjective scores for their cases and wanted to evaluate the accuracy of ViSQOL measurements as well as identify limitations. A frequent issue was related to the duration and segmentation of the audio that would be used with ViSQOL when used in an automated framework. While ViSQOL in speech mode has a voice activity detector, it was found that ViSQOLAudio would perform poorly for segments where the reference was silent, because of either the averaging effects, or because of the lack of log-scale thresholding which was overly sensitive to small absolute differences in ambient noise levels. To resolve the averaging effects, it was recommended to extract segments of audio of 3 to 10 seconds where there was known activity. A solution to the thresholding issues is discussed in the next section.

2.4 Discussion

Here we present a discussion of the use cases and feedback in light of the improvements. This is followed by reflection on trends and the areas that are promising as future work.

The case studies mentioned in Section 2.3 highlight the challenges with real world applications of ViSQOL. The findings are generally that ViSQOL can be used for

various applications, but careful investigation is required for any use case. The users of these tools are the very developers of new audio processing and coding techniques, and are often analyzing new types of audio that are "out of sample". In some cases, we can allow the user to retrain a model to match the new data.

We find that developers are reasonably skeptical about how well ViSQOL will apply to their problem, given that it almost always has unique characteristics. Although ViSQOL is not guaranteed to give a meaningful absolute MOS for cases that are significantly different from what it was originally designed with, the developers in our case studies found some correlation that was useful for their use case. However, this conclusion is often facilitated by the use of additional metrics that can be used to validate ViSQOL's application.

In other cases, for example, in the generative case, it is possible that it requires a redesign of the algorithm at a fundamental level, which could include different spectrogram representations or DNNs. Projects like LibriTTS [33] have curated large amounts of freely available speech data, which has been a boon to speech-related DNNs, there is yet no standard and widely available subjective score dataset that is of similar scale. A larger dataset would enable new development, but also require rethinking of existing tools, such as support vector regression, used by ViSQOLAudio, which is intended for use on smaller datasets on the order of hundreds of points.

2.5 Summary

This chapter introduced the history, fundamental properties, and design of ViSQOL. It described a new version of ViSQOL which is available for use on GitHub. The integration to real world problems by different teams at Google yielded a number of insights and improvements to the previous version. However, the described

version of ViSQOL relies on traditional signal processing modules. There are a number of promising avenues for improvements on top of ViSQOL that will be discussed in the next two chapters, including DNN based approaches, a more general model, and taking the new generative audio approaches into account.

3 | Speech Quality Estimation with Deep Lattice Networks[1]

3.1 Introduction

As seen in the previous chapter the estimation of speech quality is useful in a wide range of applications, such as channel transmission and speech coding. PESQ [2, 3], POLQA [4, 5], and ViSQOL [8, 9, 18] are popular tools for objectively estimating the mean opinion score from a subjective test intrusively, that is, by comparing the degraded signal to a clean reference signal. The input space for such estimators is constantly expanding due to the innovations in synthesis, coders, and hardware capabilities, which create new categories of audio data and their associated biases, artifacts, and perceptual exploits. Audio coding examples of this are the frequency masking in MPEG audio coding, and the recent advent of neural generative models for speech, such as WaveNet [35], which achieved a new state of the art in speech synthesis. WaveNet diaspora include low bitrate parametric vocoder conditioning [12] and LPCNet [11], both of which provide very low bitrates (as low as 1.6 kbps) while maintaining reasonable speech quality. These models add subtle shifts in time and pitch that are problematic for traditional speech quality estimators which typically have an alignment preprocessing stage that was only designed for shifts in VoIP artifacts. Recent work has investigated the design of a metric that is able to track the quality of these generative models [36]. Generative adversarial networks (GANs) [37] for speech such as MelGAN [38] produce realistic output of yet another nature because the adversarial loss encourages the output to be plausible and realistic. Voice over

[1]The content from this chapter was originally published in the Journal of the Acoustical Society of America in 2021, vol. 149, no.6, with co-authors Jan Skoglund and Andrew Hines [34].

IP (VoIP) applications produce new classes of artifacts when introducing novel methods for packet loss concealment [39], which prompt new speech quality models [40]. These are just a few examples; there are many other areas of research that contribute to the increasing diversity of computer generated audio, such as speech enhancement, physical modeling, text-to-speech, style transfer, and so on.

There are also non-technological and indirect natural processes that continuously affect subjective opinion and add additional sources of uncertainty. In contrast to the rapid development of speech synthesis and signal processing, the physical properties of hearing and human auditory anatomy are relatively constant, subject to a slow evolutionary time scale. On the other hand, geographical and technology-driven developments in society affect subjective opinion of these new technologies on a faster time scale. Language, culture, and nationality also are considered to affect subjective testing [41]. Additionally, while human hearing is relatively stable, natural speech production changes from generation to generation within a given region. The dialects and accents of regions and cultures develop slowly compared to technology, but fast enough that changes can be noticed within a region or across generations [42]. These types of societal changes happen over decades, in parallel with technology, and care must be taken to update and validate the models over time to reflect the current opinion. This is analogous to the changes in models that have occurred over the last decade for measuring the quality associated with page load time of web applications as network speeds have increased and evolving user expectations [43]. Speech quality can be assessed by using subjective listening tests to rate samples by a group of subjects scoring using a 5-point absolute category rating scale. The per subject results are averaged to produce a mean opinion score (MOS) to score the quality [1]. Besides social and cultural differences, it is inevitable to have non-zero measurement and sampling error in MOS between tests because of environmental differences, or

pre- and post-screening methods. Efforts such as the ITU-T listening test recommendations in ITU-T Rec. P.800 [1] try to limit the extent to which this happens, although various biases are present in all standard listening tests [44]. Rather than trying to eliminate or minimize the uncertainty and variation in the data due to biases, uncertainty from biases could be seen as a property that can be modeled [45, 46]. However, there are valid questions about whether MOS is the most useful property to model for speech quality [47, 48].

Machine learning has been applied to acoustics in a wide array of fields. Deep learning has become more prevalent as the collection of acoustic data becomes more convenient (e.g. with crowdsourcing and cheaper storage), and compute capabilities grow. Acoustics problems fundamentally involve physical constraints, and years of research has yielded interpretable models to solve these problems. Rather than an opaque, purely data-driven model, it may be preferable to have a hybrid model that is able to harness the power of deep learning while being aware of acoustic constraints and other known relationships between the input and the output [49].

Within quality estimation, traditional models typically rely on digital signal processing toolboxes, where each component has a well recognized function (e.g. low pass filters, signal alignment, and resampling modules). In the recent surge of progress in deep neural networks (DNNs), deep learning has also been applied to speech quality estimation with increasingly good results and flexibility. For example, [50] have developed a convolutional neural network (CNN) based approach to estimating MOS both intrusively and non-intrusively. One of the general criticisms of deep learning is that it tends to create a 'black box' model. That is to say the explainability and interpretation of the modular components of the model are traded for prediction accuracy. Additionally, traditional methods relied upon signal processing and other accumulated knowledge that effectively served as priors

are not, in general, included in DNN-based solutions. Bayesian models were also explored to handle complex models with greater transparency [51]. Recent work in shape constraints for DNNs in the form of deep lattice networks (DLNs) [52] provides methods of using prior information to constrain the output in meaningful ways. Previous work has also used constraints to map a factor to quality with linear and additive models. For example OPINE's [53, 54] influence on the E-model used additive psychological factors that enforced a monotonic constraint on the MOS with respect to the factors [55].

This chapter considers the usage of DLNs for MOS estimation from a measure of similarity to satisfy several desiderata and prior constraints. First, the model should be aware that MOS is only defined in the 1 to 5 range. This seems like a trivial constraint, yet it introduces problems for mapping functions that are unaware of this. Second, as similarity increases, the MOS should generally increase, i.e., the model should be aware of the monotonic relationship between similarity and MOS. The proposed experiment considers the effects of using a DLN with these constraints to learn a high dimensional mapping function that operates on input features provided by ViSQOL. The resulting system uses traditional signal processing components and has prior shape constraints and explainable mappings. Lastly, the DLN's ability to have monotonic output is used to produce an inverse cumulative distribution function, or quantile function, that is able to provide *calibrated* estimates for an arbitrary quantile. The estimates are calibrated in the sense that a given percentile estimate will have approximately that percent of the data below the estimate. Calibration allows for the model to express uncertainty, for example, in the form of quantile intervals which may be more meaningful to the user than a point estimate without any notion of uncertainty.

This chapter is presented as follows. First, a system overview of DLNs and ViSQOL is provided. Next, the design choices and training methods for the

DLN-based mapping function, as well as the probabilistic quantile function, are described in detail. The following section describes the experimental results of various models over several datasets. In the next section the results are discussed in a wider context, as well as the implications they have on other applications. Lastly, a concluding section summarizes the work.

3.2 System Design

A deep lattice network is used as a component of ViSQOL to estimate MOS. Several modules serially process the input waveform to compute the input features for the DLN. An overview of the system is available in figure 3.1.

At a high level, the system estimates mean opinion score by applying the following transformations in sequence. First, the reference and degraded signals are normalized on RMS energy and globally aligned via cross correlation lag. A gammatone filterbank [56] is chosen for its resemblance to the auditory design of the cochlea, and a spectrogram for both the reference and degraded is created with it. Next, a simple energy-based voice activity detection algorithm is used to detect the segments of the reference signal that contain active speech. These active regions are broken up into patches of hundreds of milliseconds and paired with equivalent patches in the degraded signal. The neurogram similarity index measure (NSIM) [24] is calculated on the difference of these paired patches. Simple thresholding is applied to limit the sensitivity of the logarithm to small differences in low-energy segments. Lastly, a mapping function is applied to obtain an estimate of the MOS. PESQ and POLQA have similar components, but notable differences as well. For example, they have normalization and alignment steps, but they use Bark spectrum and time-masking. Notably, PESQ and POLQA both use a 3rd order polynomial mapping function as the final step to calculate

MOS [57].

3.2.1 High-dimensional monotonic constraints

Here a description of the various mapping functions under consideration is provided. The cubic polynomial mapping function maps a one-dimensional similarity to MOS. A one-dimensional exponential mapping function may be chosen over the polynomial mapping because of saddle points and non-monotonic fittings that arose from the data. With either of these mapping functions, the one dimensional mean-reduction of the across frequency bands may introduce significant error due to the nature of the lossy reduction as the data becomes more dispersed. The issue being described can be seen in figure 3.5, and will be discussed further in section 3.3.

Higher dimensional mapping functions provide a solution to this problem. Support Vector Regression (SVR) [58] is able to obtain higher accuracy but can have significant regions of non-monotonicity, and a tendency to overfit. Other solutions such as piecewise linear functions and DNN-based approaches present interesting solutions to the high-dimensional monotonic problem, and a method that combines these techniques is considered.

3.2.2 Deep lattice networks

Deep lattice networks (DLNs) [52] provide a method for using functional shape constraints, such as monotonicity or convexity with deep learning. This is accomplished with high dimensional piecewise linear functions that interpolate, and is related to earlier work on monotonic calibrated interpolated look-up tables [59]. The tf.lattice framework [60] is used to realize the DLN.

Figure 3.2 illustrates the layers of the system. At the input layer of the lattice

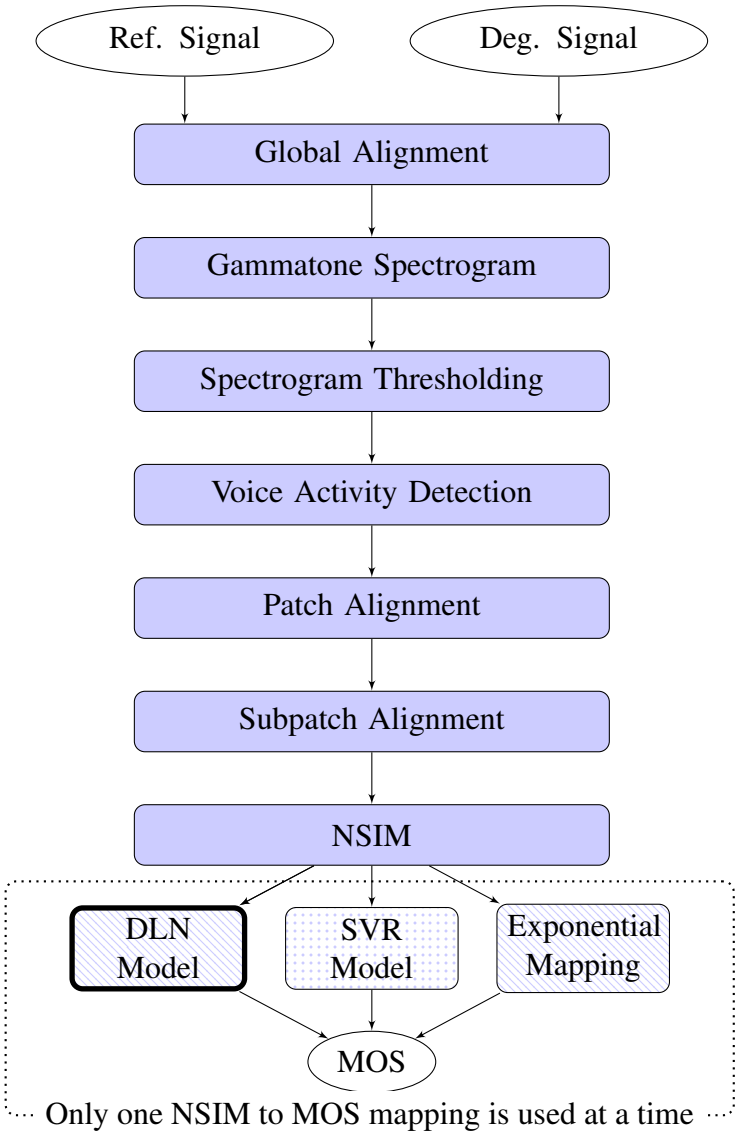


Figure 3.1: System Diagram. The inputs and outputs have white fill, and the processing components have blue fill. New components have thick edges. The dotted fill represents optional mapping functions.

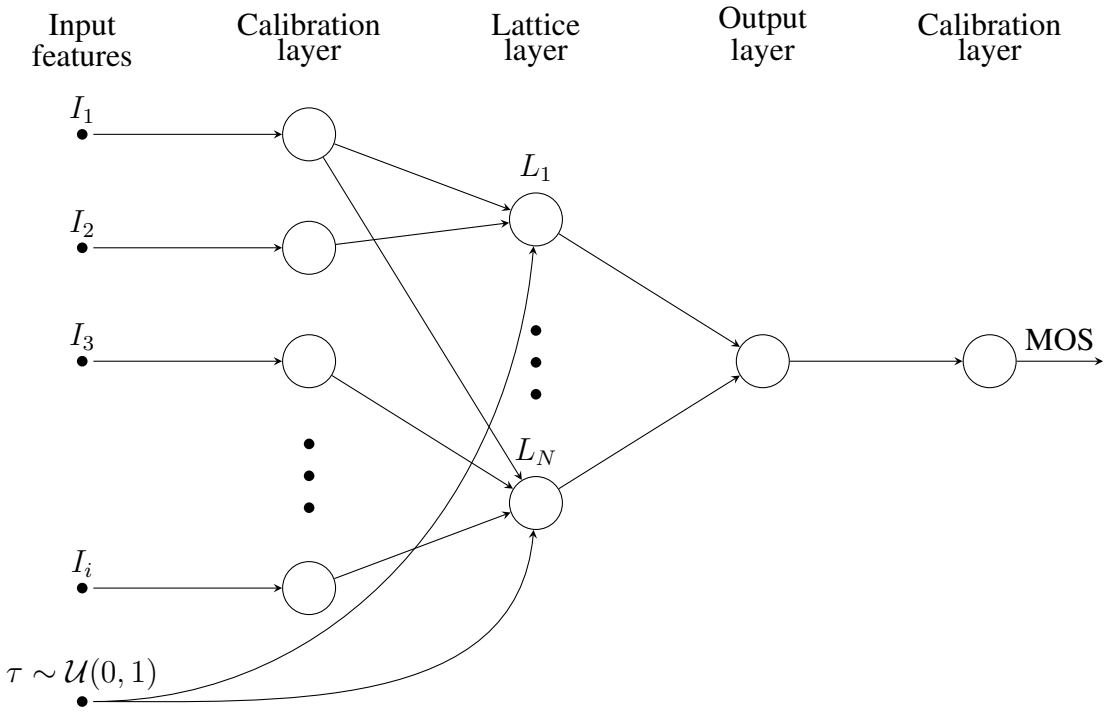


Figure 3.2: Layers of the deep lattice network. Input features I_1 to I_i are scalar values such as per-band NSIM or energy. τ is the quantile, which is generated during training by sampling the uniform distribution. Each lattice ensemble L uses τ and a random subset of input features.

network, each input feature (such as NSIM means and energy) is calibrated into monotonic piecewise linear functions with k pieces so that each piece is a bucket large enough to contain a uniform $\frac{1}{k}$ fraction of the training data (which provides a quantile-based normalization). These calibrated features are combined in a network of N lattices by randomly choosing M unique input features, with the guarantee that one of the features in each lattice will be the quantile parameter, τ, to allow for MOS to be monotonic on a global feature. Each lattice interpolates subsets of the calibrated features in more than one dimension, with some of these features having monotonic constraints. Finally, the lattice outputs are combined linearly and the output is optionally calibrated with the same method that the input was calibrated with.

The number of parameters per lattice is the product of the lattice sizes of each feature in that lattice. The maximum number of lattice parameters is K, but varies due to the random selection of features with differing lattice sizes. If the lattice exceeds that number of points, then features are removed until it is below that limit. This is important because the per-frequency band average NSIM (FVNSIM) feature has a different lattice size than the other features, which have the minimum lattice size of 2. Therefore, the total number of parameters in all lattices is less than NK.

Other prior constraints can be placed on the model based on the design of the subjective test, such as the requirement of output in the 1 to 5 interval. This prior can be modeled by the DNN directly, instead of naively clamping an unbounded output to the interval. This lets the network be aware of the sensible limits of the problem and has been shown to address issues with out of sample data producing nonsensical values, which can be an issue with traditional DNNs. Furthermore, if the model outputs a distribution, it allows the model's probability density function to have support for meaningful values.

The following subsections describe how the lattice network is adopted to the MOS estimation task and is illustrated in figure 3.3.

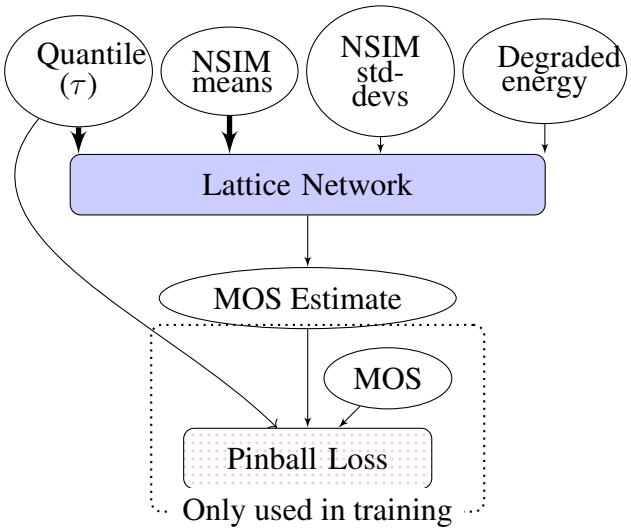


Figure 3.3: Deep Lattice Network Subsystem. The MOS estimate for a pair of reference and degraded audio depends on the quantile at which it is requested. Features with thick arrows indicate that the model will produce a MOS estimate that is monotonic on them.

3.2.3 Learning a quantile function

Mean squared error is perhaps the most popular criterion for regression methods. Section 7.3.2 of the ITU-T Rec. P. 1401 [61] describes minimizing root mean squared error (RMSE) for 1-D mapping functions. Mean absolute error (MAE) is also sometimes considered as a loss function. Using MSE, RMSE, or MAE to estimate the mean will invariably fail to represent the true distribution, because they use point predictions that discourage outliers, with MSE having the largest penalty for these. In contrast, estimators that predict a distribution instead of a single point typically minimize loss terms such as maximum likelihood or the pinball loss (described below), and will be penalized for failing to have support for outliers. Consider the scatter plot of the point-estimate based system for the expo-

nential mapping (which uses an MSE loss) in figure 3.5. The extreme predictions nearest to 1.0 and 5.0 are compressed by the squared error. Adding a standard deviation output partially resolves this problem, but introduces other issues, since the mean and variance alone would be able to specify a normal distribution, but are not sufficient statistics for arbitrary distributions. The definition of MOS imposes a constraint on the distribution by truncating the ends from 1 to 5, meaning that a normal distribution is not useful here without transformation. For example, by assuming a normal distribution and sampling means near the extremities, it is easy to obtain values beyond the allowed values (e.g. 5.2 or 0.5).

The Bayesian approach to solving the MOS constraints problem would use a prior that only has support from the 1 to 5 region, with some support near the extremities. The naive DNN approach of point estimates clamped to the 1 to 5 range would have the advantage of handling higher dimensional input, but does not solve the problem of mapping the output to the boundary or monotonic constraints without post-processing. Clamping a DNN's output limits the ability to model the distribution of point estimates, but it is also difficult to model a quantile function from the data alone without being aware of the bounds due to the fact that rare events are unlikely to be represented in a dataset of limited size (which is often the case for speech quality datasets). The approach of both using priors to capture intrinsic properties of the problem and the modeling of uncertainty is desirable for this reason. Although DLNs are not Bayesian models, they use the concept of priors to solve this problem. Estimating quantiles functions using regression [62] also provides these advantages, and this idea fits well into the DLN framework because of its support for monotonic features.

It has been shown that the quantile learning problem can be solved in a deep learning context with a deep lattice network [63]. The DLN learns a monotonic quantile function $f(x, \tau)$, which predicts a MOS given the features x (e.g. NSIM)

and an auxiliary quantile parameter τ. This function has monotonicity enforced on the quantile parameter τ by the DLN, which allows $f(x, \tau)$ to be used as an inverse cumulative distribution function, also known as a quantile function. During training, τ is randomly sampled from the uniform distribution $\mathcal{U}(0, 1)$, and trained with the so-called 'pinball loss' [62] which can be written as

$$\begin{aligned} \mathcal{L}_\tau(y, \hat{y}) &= \begin{cases} (y - \hat{y})\tau & \text{if } y \geq \hat{y}, \\ (\hat{y} - y)(1 - \tau) & \text{otherwise} \end{cases} \\ &= \max(\tau(y - \hat{y}),\ (\tau - 1)(y - \hat{y})). \end{aligned} \tag{3.1}$$

During inference, the user can provide the value of τ for the desired quantile along with the input audio. With this method, one can draw samples from the posterior distribution of the estimated MOS by sampling τ from a uniform quantile and should be able to recreate the input distribution if the model is working well, without the effects of compression of the output space due to regression on mean squared error. Since the quantile fully specifies the distribution and should be well-calibrated because of the pinball loss (as is verified in section 3.3.5), it may be more useful to output a quantile interval to provide some information about the uncertainty in the estimate. As the distribution is arbitrary, this interval is not necessarily symmetric around the mean. This should be an advantage over standard-deviation based confidence intervals for the MOS problem due to its value constraints (e.g. a very high median MOS of 4.9 should have an interval of at most 5.0 for its upper value, but it could have a value lower than 4.8 for its lower value). The user may also use the model to deterministically estimate the median MOS value for a pointwise estimate (or any other fixed quantile other than the mean). It is less convenient to compute the mean from the quantile function, although it is possible with integration. For most applications it may be more

convenient to use the median $\tau = .5$ and a compatibility interval.

3.2.4 Mapping function input features

The neurogram similarity index measure (NSIM) over the gammatone spectrogram for 21 frequency bands is used as the input features. NSIM is calculated for each aligned patch and frame, and is then aggregated by taking the mean over all frames. This input feature efficiently aggregates the time dimension, but discards information about the distribution of NSIM scores within the band. We propose that by including the variance or standard deviation of NSIM per frequency band (which we refer to as FSTDNSIM below), the mapping function may be able to make more informed choices. Additionally, with the introduction of a quantile function, the variance of the band's similarity measure can be used to help quantify the width of the distribution as well.

Additionally, we propose to add the per-frequency band energy of the degraded signal as an additional input feature for the mapping function. NSIM is a relative measure, so it is difficult for the model to infer how important a low or high NSIM score is without knowing the absolute energy of one of the signals. Including the degraded energy feature allows the model to distinguish the importance of each band in reference and degraded signals that have significantly different levels of energy. A frequent case where this happens is when comparing two signals that are narrowband or low-passed, and do not have significant energy in some upper frequency bands.

3.3 Experiments

To test the effectiveness of the deep lattice network-based mapping function, we compare the use of different mapping functions that use features provided by

ViSQOL, as well as SVR, POLQA and PESQ.

3.3.1 Datasets

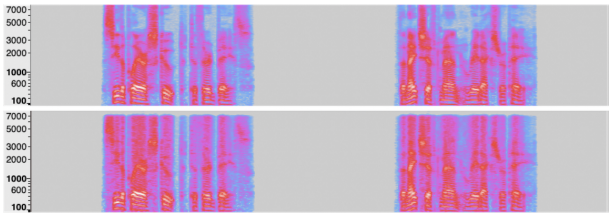
Table 3.1: Dataset properties

Dataset	Utterances	Conditions	Description
ITU-T P. Supplement 23 Exp. 1	528	44	Transmission standards and codecs from early 2000's (e.g. G.729)
ITU-T P. Supplement 23 Exp. 3	800	50	Channel degradations (e.g. packet loss, car noise, bit errors)
TCD-VOIP	384	84	VoIP network degradations (e.g. echo, packet loss, clipping)
Genspeech	192	10	Neural speech codecs (WaveNet, LPCNet, OpusNet)

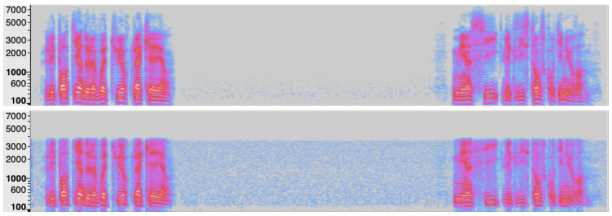
Three MOS datasets were chosen and aggregated for training purposes. The names and properties of the datasets are summarized in table 3.1. The first dataset is TCD-VOIP [64], which has a variety of VOIP conditions with wideband signals. ITU-T P. Supplement 23 [65] (Experiments 1 and 3), has a mixture of combinations of narrowband codecs and transmission artifacts. Lastly, we chose an internal dataset that evaluated generative speech codecs such as LPCNet and a WaveNet conditioned on vector quantized log-mel spectrograms, which was conducted as a MUSHRA test and linearly transformed to the MOS range. The total number of utterances in these combined datasets is 1904.

3.3.2 Lattice Configuration

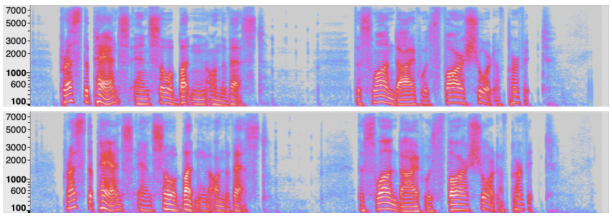
The deep lattice network was configured with hyperparameters in table 3.2 after a grid search. The model was trained for 140 epochs on a local CPU, which took approximately 5 minutes, with typical convergence. The lattice network was trained on 80 percent of the data, and 20 percent was held out for testing.



TCD-VOIP



P. Supp. 23



GenSpeech

Figure 3.4: Spectrograms from selected utterances for each dataset. The reference spectrogram is on top, and the degraded spectrogram is below. All examples are 8 seconds, with frequencies up to 8kHz in mel scale.

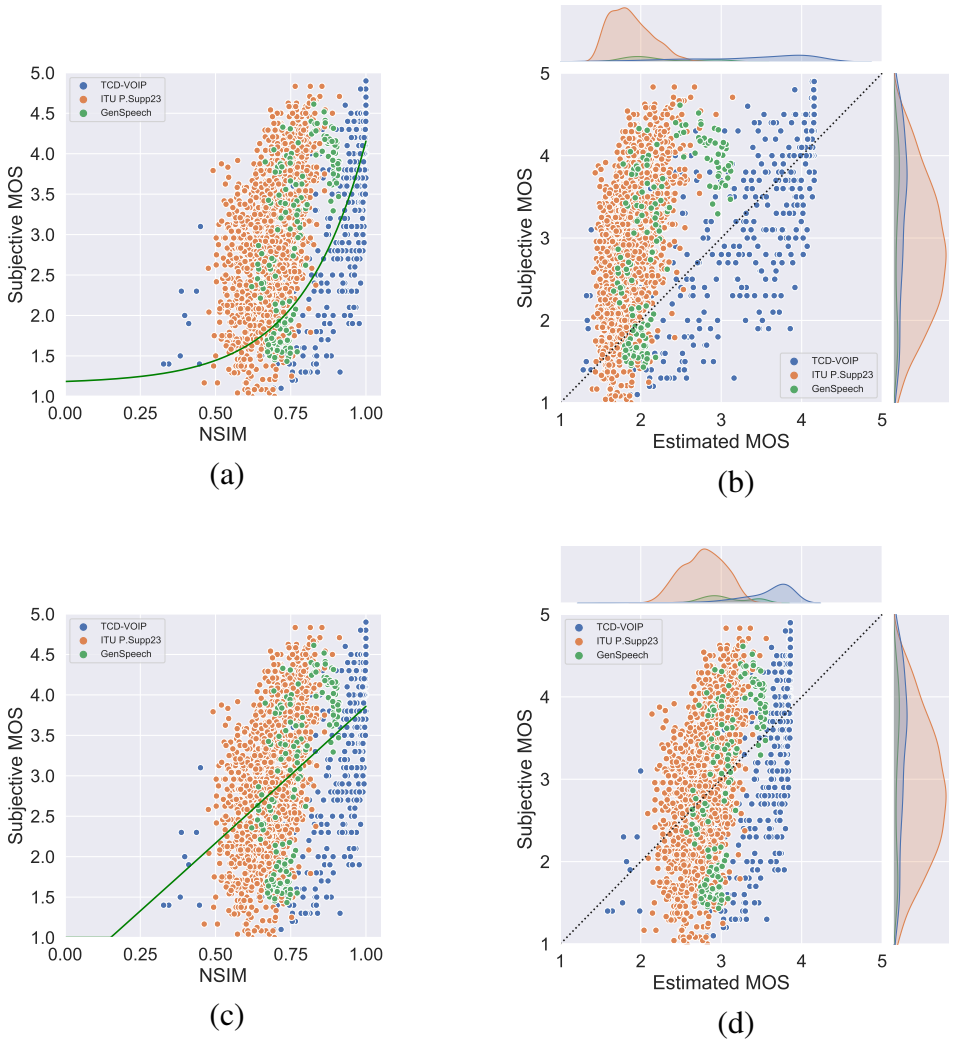


Figure 3.5: 1-D exponential mapping functions. (a) is a plot of NSIM on the x-axis to subjective MOS for all three datasets for the exponential mapping function fit to TCD-VOIP only, with a green line representing the 1-D mapping. (b) is the same model's estimated MOS on the x-axis to the subjective MOS fit to TCD-VOIP only. (c) and (d) are an exponential fit to all three datasets.

Table 3.2: Hyperparameters for deep lattice network. FVNSIM and FSTDNSIM are per-frequency band NSIM means and standard deviations, respectively.

Hyperparameter	Value
Learning Rate	0.005
Batch Size	100
Monotonic features	FVNSIM, τ
Non-monotonic features	FSTDNSIM, Degraded Energy
Max Lattice Rank	16
Number of Lattices	40
Maximum Lattice Parameters	10368
Lattice Construction	Custom
FVNSIM lattice size	3
Other features lattice size	2
Feature Calibration Keypoints	20
Output Calibration	True

To deal with overfitting, several measures are taken. First, the monotonic prior introduces a regularization effect. Next, the lattice size for the standard deviation and degraded energy features have the minimum value of 2, which strongly limits the amount of overfitting possible with those features. Next, the lattice is configured with wrinkle and Laplace regularization. The number of epochs used during training is experimentally determined by finding the point at which the test loss does not decrease. Figure 3.6 shows the difference between train and test using these techniques.

3.3.3 Comparison to other models methods

We compare the results from the deep lattice model to those from other mapping functions, as well as the popular PESQ and POLQA metrics. Table 3.3 shows a comparison of all of the discussed models.

Table 3.3: Comparison of mapping functions and metrics on all three datasets. Where applicable, train and evaluation values are separated by a slash.

Model	MSE	Pearson
Polynomial mapping	0.61/0.57	0.48/0.53
Exponential mapping	0.61/0.58	0.48/0.54
PESQ	0.61	0.76
POLQA	0.48	0.76
SVR	0.27/0.38	0.81/0.71
Deep Lattice Network (ours)	0.20/0.24	0.87/0.84

3.3.3.1 1-D exponential and polynomial mapping

A 1-D exponential mapping function was fit to TCD-VOIP. A 3rd-order polynomial mapping was also fit. The fits were found using scipy's 'curvefit' method, which minimizes the mean-squared error. As can be seen in figure 3.5, the exponential fit is acceptable for modeling TCD-VOIP in isolation, but adding additional datasets adds significant entropy at every level of MOS or NSIM. Due to the dispersion of the data, this is not related to the specific 1-D function selected. Whether exponential or polynomial, the single dimension mean reduction of NSIM is not able to resolve MOS for multiple datasets, and this uncertainty results in a very compressed range of MOS to minimize outliers due to the MSE criterion. Our conclusion is that if NSIM is the input feature for sufficiently diverse data, then additional dimensions or features are needed. The main appeal of the exponential fit (and certain polynomial fits) is that it is relatively simple and provides the monotonic constraint. Having just 3 or 4 parameters, these 1-D mappings have virtually no risk of overfitting, which explains why the test set performs marginally better than the train set, (which does not occur the higher dimensional mappings). In contrast, the lattice network outperforms the 1-D fitting in the accuracy metrics and maintains the monotonic constraint.

3.3.3.2 Support vector regression

Support vector regression (SVR) provides the ability to fit a high-dimensional mapping function over the data. SVR is not ideal for similarity mapping because it is subject to non-monotonic behavior that causes inconsistent results outside of the training data. However, for the sake of comparison we include it here to test raw accuracy performance. Using a grid search on SVR hyperparameters to search over thousands of models we found the best SVR model on these three datasets to have a mean-squared error of 0.27 for train and 0.38 for a five-fold cross validation, which is roughly equivalent to the 20 percent evaluation holdout used with the other mapping methods. The test performance is marginally worse than the lattice model, and does not provide a monotonic function. The difference between train and cross validation MSE is larger than the DLN which is an indicator of relatively larger amounts of overfitting.

3.3.3.3 PESQ and POLQA

PESQ and POLQA were evaluated over all three datasets, and achieved an MSE of 0.99 and 0.48, respectively. This is significantly higher than the lattice network or SVR, but better than the exponential and polynomial mapping. PESQ was used in narrowband mode for the narrowband cases, and wideband mode for all other cases due to the unexpected performance mentioned in P.682.3 recommendations [66].

It is also worth mentioning that both POLQA and PESQ were not designed around or trained on the generative speech data, which came into existence after their development. Because PESQ and POLQA are not adopted on the two non ITU-T datasets, it is perhaps not surprising that it does not perform well on them. Still, both POLQA and PESQ have correlation coefficients that are significantly higher

than the 1-D fitted exponential and polynomial mappings, but perform worse than SVR and the DLN.

3.3.3.4 Deep Lattice Network Mapping Function

Figure 3.6 visualizes the dispersion over the entire dataset for the DLN model. The original exponential fit has a larger error, and is compressed in range. The deep lattice network trained on the mean NSIM per frequency band features improves significantly over the original exponential and polynomial mapping. The predicted MOS is no longer heavily compressed towards the middle scores, with the points on the entire diagonal. The network is able to predict in the 1.0 to 4.75 range unlike the exponential fit (i.e., the compression due to MSE penalizing the output for the exponential fit limited the maximum MOS to approximately 4.0, as can be seen in figure 3.5c).

It is also typical to report the correlation coefficients over each individual dataset, and over the aggregated conditions. Figure 3.7 shows the conditions aggregated by mean for the same deep lattice network.

3.3.4 Predictive model evaluation

We analyze the results of the quantile lattice models, which produce a posterior distribution conditional on the input. First, we consider the calibration of the learned quantile MOS function $f(x, \tau)$. That is, for a given a quantile τ, we expect the proportion of the dataset that has subjective MOS values below the estimated MOS quantile $f(x, \tau)$ to be approximately equal to τ.

Next, we consider the forecasting power of the quantile lattice model. Mean-squared error penalizes outliers, and as a result, a model trained with an MSE criterion will produce estimates with a compressed distribution that does not match

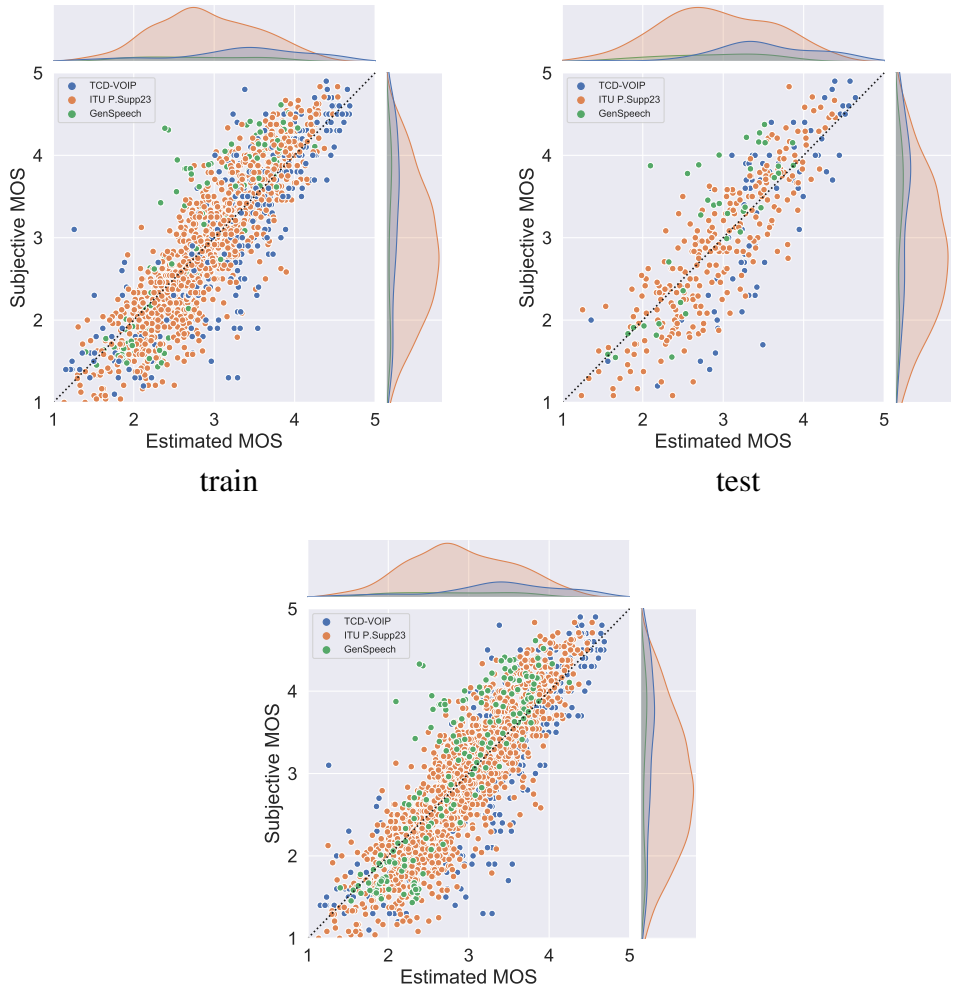


Figure 3.6: Scatter plot and marginals for the median quantile for each datapoint over three datasets using the deep lattice network model with ViSQOL. The x-axis is the predicted median MOS given by the deep lattice network model, and the Y axis is the subjective (ground truth) MOS.

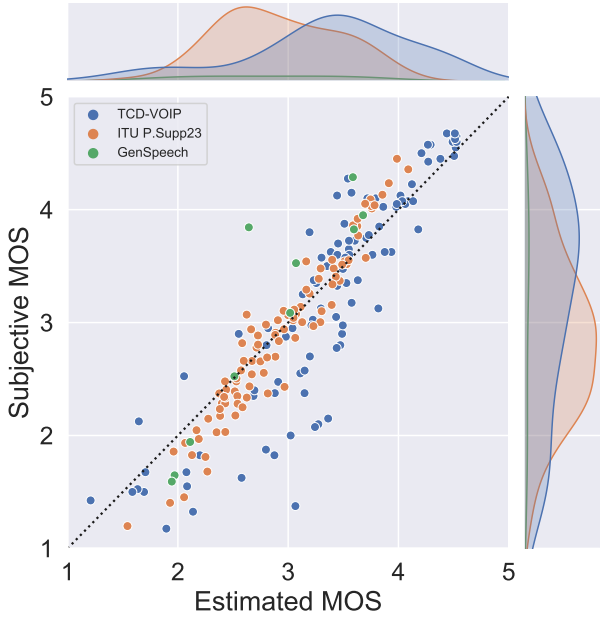


Figure 3.7: Median quantile of the deep lattice network for aggregated conditions per dataset

the shape of the input distribution if there is a non-zero error. Furthermore, the previously used point-estimate model does not describe the uncertainty of the prediction, which can be useful information for diagnostics or sampling. The quantile model produces an arbitrary distribution that can be used to draw samples from, or to provide the user with more information, for example in a quantile interval or histogram. When there is uncertainty about the prediction, the model should reflect this by producing a marginal distribution that is similar to the target distribution.

In Figure 3.8 we see that sampling the predictive distribution with a random quantile $\tau \sim \mathcal{U}(0, 1)$ leads to a marginal distribution that is more similar to the ground truth distribution, without the compression seen in other models. Since the compressed models have areas of literally no support, and the quantile models generally have support anywhere the input lies, meaning that the KL divergence is lower

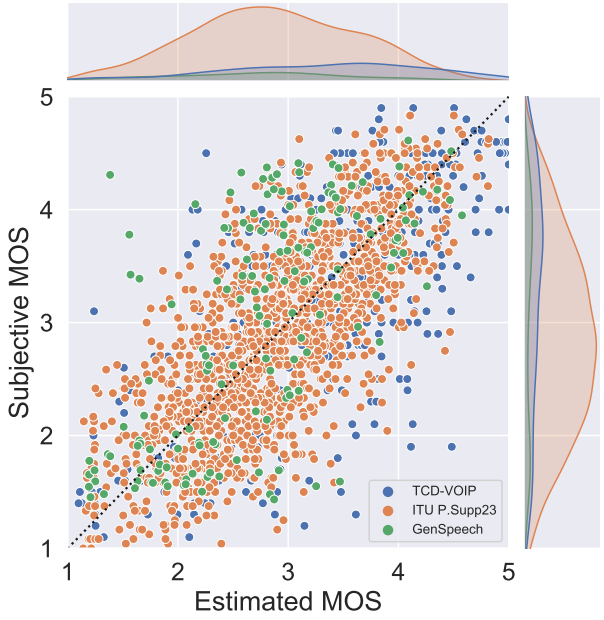


Figure 3.8: Samples drawn from a random quantile for each datapoint. Compared to the median quantile, the accuracy decreases but the marginals become more similar.

for the posterior models. In other words, for a certain level of quantization (which is required due to only having samples of the distributions), the posterior models have a finite KL-divergence against the ground truth, whereas the predictive distribution would have an infinite KL-divergence.

The quantile model can make point predictions using the median, mode, or mean, although the median is the most straightforward. As can be seen in figure 3.6, the median point samples reduce the expected error in a similar fashion to how mean point samples would, although the MSE should be optimal for mean point samples. However, mean and median are both insufficient statistics to describe the conditional distribution of MOS given a signal. Both types of point samples provide less information about the prediction than using the entire estimated distribution or a summary such as a quantile interval. To determine what a useful

quantile interval is, we consider the calibration of the quantile function next.

3.3.5 Calibration of quantile function

Table 3.4: The calibration, error, and correlation coefficients of the learned quantile function for the selected model. The values before the slash are for train, and the values after the slash are for test.

Quantile	.0	.1	.25	.5	.75	.9	1.0
Ratio under prediction	.039/.043	.078/.11	.23/.27	.53/.49	.77/.75	.92/.92	.99/.99
Mean squared error	0.89/0.94	0.58/0.61	0.29/0.33	0.20/0.24	0.29/0.34	0.57/0.64	1.03/1.13
Pearson	.86/.83	.87/.83	.87/.84	.87/.84	.87/.84	.86/.83	.84/.82
Spearman	.86/.85	.87/.83	.87/.84	.87/.84	.87/.84	.86/.83	.85/.82

The calibration of the quantile function refers to how accurately the model describes its uncertainty. For example, at the .5 quantile, half of the predictions should be above and half below, and at a quantile of .9, the model should overestimate all but 10 percent of the data.

Figure 3.9 visually shows that the quantile function of the model is reasonably calibrated for various quantiles. As expected, as the quantile becomes closer to 1.0, the predictions shift to towards a larger estimated MOS. However, it is important to note that the quantiles are not a linear transformation, but are conditional on the data and subject to uncertainty, which is modeled by the pinball loss. Thus it can be seen that the relative order of the data points changes to some extent, since the inputs with larger uncertainty have a larger change.

Table 3.4 shows the calibration accuracy for the selected model for various quantiles. As the percentage of underestimates are aligned with the quantile, we conclude that the result is generally well-calibrated, although there is some skew and bias observed in the extremes as can be seen in figure 3.9. As such, we consider that a median prediction along with a .1 to .9 quantile interval should be useful information to report to users (e.g. 'Estimated MOS median: 4.3, 10% quantile:

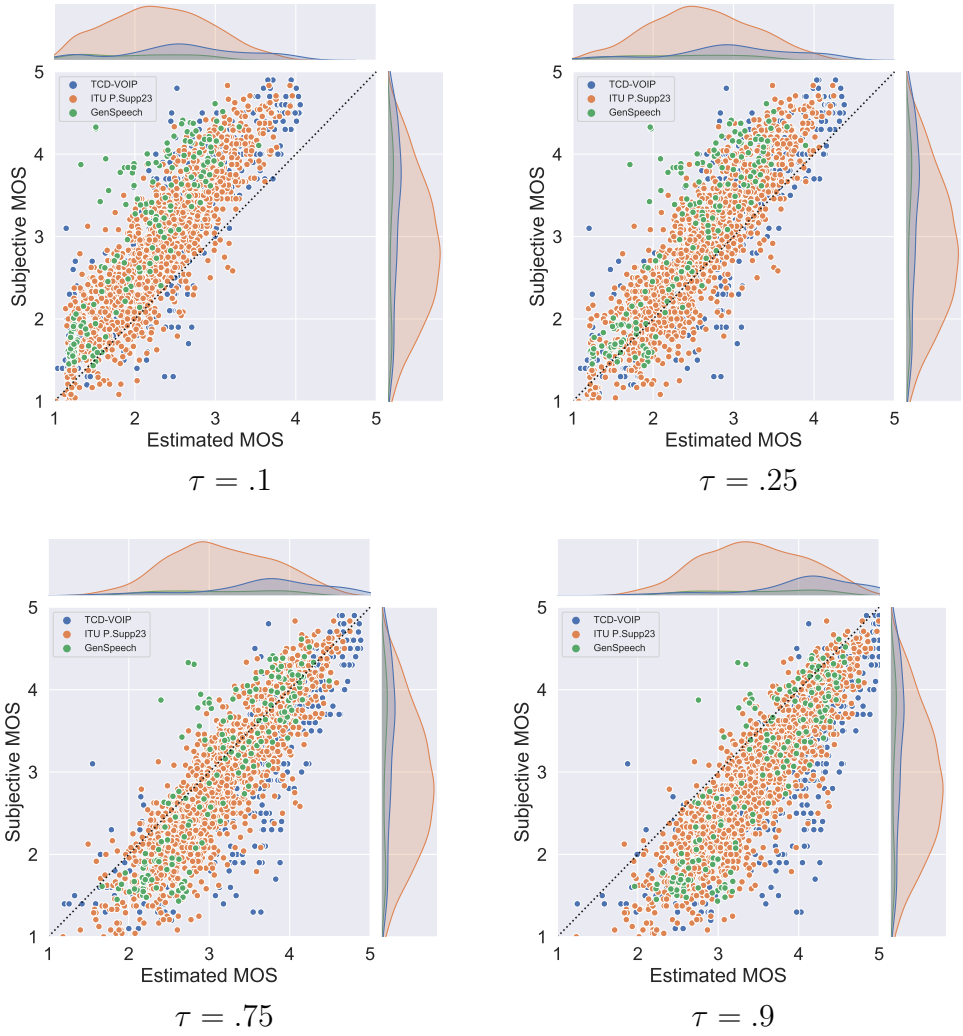


Figure 3.9: Calibration of estimated to actual MOS scores for various τ quantiles over the same data. For the median $\tau = .5$, see fig. 3.6

3.8, 90% quantile: 4.6').

3.4 Discussion

The experimental findings presented above suggest that using deep lattice networks as a mapping function should be beneficial in several ways. First, the general accuracy of the predictions is significantly higher than other methods we tested. Second, the ability to model calibrated uncertainty via a quantile function should provide the user with valuable information. Third, the model is more explainable than other high-dimensional mappings including SVR and typical DNNs, because of the monotonic and shape constraining priors, as well as the regularizing effect of small lattice sizes. In this section we discuss the findings in a general context not specific to ViSQOL and speculate on interesting applications in the quality estimation domain.

The finding that constraining MOS to be monotonic on NSIM performs well suggests that there may be other metrics that would benefit from a monotonic, convex, or concave model. This could be applied to a speech quality model where mapping an internal representation to a human readable scale occurs, such as in POLQA or PESQ. The findings are also interesting for other applications such as intelligibility. For example, STOI [67] could be mapped to an intelligibility scale. Another example where this could be useful are non-intrusive metrics which use automatic speech recognition (ASR) and word error rate (WER) as proxy input features that could be mapped to intelligibility and quality predictions.

The internal representation must still be chosen carefully, because it may contain regions that are non-monotonic. For example, SNR of unaligned or subtly warped signals will be lower than the SNR of aligned signals but the quality may be equivalent, so a monotonic prior may not be able to model quality without sig-

nificant uncertainty. It should therefore help the model to use an alignment step before calculating SNR. The importance of feature design is critical in any quality estimation framework, but adding prior shape constraints means that the system designer must take extra care that the features actually do have the specified relationship with quality.

So far, only features related to the audio signals have been considered. However, the ability to handle high dimensional inputs can also be extended to metadata features. For example, the bias of culture, language, and test environment discussed in the introduction could be integrated as additional features of the model. The user can then learn a model that can simulate and explain the effects and bias of different subjective tests. This requires the model to be trained on this kind of metadata. There are not many publicly available speech quality datasets compared to general speech datasets, but fortunately there are a number of datasets (such as the ITU-T P. Supplement 23 dataset) that test the same conditions in several countries and languages that should be sufficient to test this idea. There has been previous work that has explored the P. Supplement 23 dataset using Bayesian models [45] to model the heterogeneity in the data, showing that this approach is promising.

Adopting a more complicated model that has better performance often comes with a cost in terms of size and complexity, which often affect explainability. Deep learning methods typically produce 'black box' models that are difficult to interpret because of the many layers and non-linear transforms that make it complicated to explain the model's predictions. In contrast, the DLN approach uses deep learning, but it is designed with explainability in mind (for example, the DLN authors use it for applications in ML fairness [68], where explainability is essential). The constraints that the system designer places on the model are strict constraints, which the model is required to satisfy, providing immediate insight into how the

model maps the input to the output. The network is also relatively shallow at four layers compared to typical DNN models, all of which have an interpretable function. The only computationally intensive layer is the lattice ensemble layer, with each lattice having a very small number of points (typically two or three) along each feature dimension to interpolate over in piecewise linear fashion. The input calibration 'layers' can be calibrated in a preprocessing step before training occurs. Although it can have a high dimensionality that may be difficult to visualize, the lattice model is relatively more explainable than an RNN stack because it does a straightforward interpolation lookup instead of a series of arbitrary transforms.

3.5 Summary

This chapter has presented the results of using Deep Lattice Networks to maintain a monotonic constraint for high-dimensional similarity. The deep lattice network outperforms the other mapping functions studied, and extends the functionality to probabilistic quantile intervals that provide more information on the uncertainty of the estimation. The metric provides robust predictions of speech quality for several types of distortions. Namely, the three datasets studied contain VoIP, additive noise, and coding degradations. Importantly, the quantile function from the DLN has been shown to be reasonably calibrated and accurate.

This work has shown that DLNs can take advantage of NSIM as a monotonic prior for speech quality. It is plausible that other speech or acoustic features will benefit from the application of a DLN. For example, metrics such as speech intelligibility, word error rate, or direct-to-reverberant ratio (DRR) may serve as monotonic priors for quality estimation or other applications. The calibrated uncertainty may also be useful for other acoustic applications, for example, when it is critical to provide bounds for an estimated value. Lastly, the DLN provides the ability to

have domain-aware priors that may be useful for any bounded problem that can help the problem of nonsensical predictions when the input is out of sample. This frequently occurs in training with acoustic features due to environmental differences in training data and deployment.

Quality estimation from speech signals from subjective rating data tends to aggregate multiple listeners and environments to a single MOS. Using other metrics, such as modelling the individual listener response, and taking other factors such as language and speaker gender into account may yield more accurate predictions. The DLN structure should be compatible with such a model. This would provide an interesting experiment for future work.

4 | Marginal Effects of Language and Individual Raters on Speech Quality Models[1]

4.1 Introduction

Measuring and estimating speech quality is an important task for many fields. For example, in speech synthesis and coding [13], subjective measurements of quality can be used to validate novel designs, and may be especially useful when traditional objective metrics like SNR diverge from human perception. The absolute categorical ranking (ACR) test asks raters to measure the quality of speech utterances under various test conditions by assigning a score from from 1 (bad) to 5 (excellent), with recommendations for conducting the test in ITU P. 800 [1]. Typically, each utterance has multiple listeners. The mean opinion score (MOS) can be calculated by aggregating the scores over each utterance or all the utterances within a given condition. MOS is a standard measurement that is used in research and development of many speech applications such as codecs and speech enhancement [70].

Because it is expensive and logistically challenging to conduct a subjective experiment, researchers and developers often use estimates of MOS that do not require running a subjective test. Some MOS estimation techniques estimate MOS using a model of the effects of different psychologically pertinent factors (e.g. echo, delay, SNR, language, gender), without looking at the actual signal [53, 45]. Tools such as ViSQOL [8, 18, 34], POLQA [5, 4], PESQ [3, 2], and the E-model [55, 71]

[1]The content from this chapter was originally published in the IEEE Access Journal, 2021, volume 9 [69].

provide immediate and objective estimates of MOS using an intrusive method that looks mainly at the signal as opposed to these factors (by considering both the reference and degraded signal). These models obtain a MOS by fitting a mapping function using features from the signals and MOS extracted from datasets such as ITU-T P Supplement 23[65] by aggregating all listeners over each utterance.

Additionally, there are deep learning methods for estimating MOS non-intrusively, which learn latent features that are mapped to MOS [50] by looking even further into the lower level aspects of the signal. Deep learning requires significant data, and some work has been done to bootstrap by augmenting the data [72] or by clustering [73]. Another approach may be to use data more efficiently, by looking at the causes of measurement and sampling error in individual scores. The trend in research is to use machine learning to extract increasingly useful information from the signal. The vast majority of MOS estimation tools use only-signal level predictors, which effectively treats all populations of human raters as identical, which is clearly not the case.

Identifying and employing interesting information in the data is important to be able to design a good model. When the listeners are aggregated for fitting a mapping function to MOS, (e.g. by taking the arithmetic mean of all listeners), information about the variance of the scores with respect to the utterances is discarded, as well as information about the individuals. The resulting models are therefore unable to describe the effect of the presence of individual listeners. As an extreme example, if a very optimistic rater rated everything a '5', such a model is not able to capture the fact that the presence of the rater causes a slightly higher mean score and will instead have a higher error in predictions. The result is that the individuals that are outliers have an oversized effect on the mapping.

Proper handling outliers properly is increasingly important with crowd-sourced

data, e.g. listening tests that are conducted over a web service such as Amazon Mechanical Turk. In these cases the quality of ratings can be worse than traditional lab tests and may require additional pre-screening and restrictions [74], (e.g. raters that always give the highest or lowest score due to any number of reasons including bad headphones or malicious ratings), and outliers will have an undesirably large influence on the mapping. The ITU provides post filtering recommendations that filter based on deviation from normal behavior, which generally works well. However, the filtering process is sensitive and can have undesirable consequences. Such filtering has the potential to remove genuine scores and leave undesirable ratings in the data.

A model of the distribution of individual rater scores that is aware of the bias that a particular rater has over multiple utterances should provide the ability to exclude undesirable raters in a more systematic fashion, as well as extracting more information in raters that have significant bias. Post filtering will exclude some raters that simply have a positive or negative bias, although their relative ratings match the overall trends. Modeling the individual listener score will allow for the model to be able to take into account this rater data, accounting for the rater bias. Additionally, many researchers have pointed out issues with using MOS as the primary quality metric and have proposed alternatives [47, 48, 46]. Modeling the individual rater score allows using the model for other metrics that are alternatives or complements to MOS.

There are numerous other biases at work in a subjective audio quality test, including many biases that are related to how the test is conducted [44]. Besides the audio signal, test environment, and individual listener bias, the effect of language and culture may have a causal relationship with opinion scores. Some languages and cultures rate the same set of test conditions higher or lower than others. For example, Japanese listeners tend to rate the quality of speech lower on aver-

age [41], and have less variance in their ratings than listeners with other native languages. As another example where Japanese means are lower, Figure 4.3 depicts Japanese and French ratings in the ITU-T P Supplement 23 dataset with the same test conditions. This effect appears to be cultural, or at least it is not exclusively attributable to phonetic differences in languages, as other quality of experience (QoE) ratings from other non-speech domains (e.g. restaurant ratings) for the Japanese seem to follow this trend [75]. To further confound this issue, the quality labels recommended in [1] (such as 'excellent'/'good'/'fair'/'poor'/'bad') may not imply a linear progression of quality within a language, and furthermore, it is not likely that each level has equivalence between two different languages [44].

All of these factors suggest that for the same set of conditions, a certain difference in scores is expected for different languages. This difference means that a model that does not have language as one of the model's predictors will produce a larger error on the predicted MOS when compared to a similar model that has the language predictor. Furthermore, for users of a MOS estimation tool, it may be more desirable to estimate the MOS for the language that is being tested, as opposed to the global MOS for all listeners from any language. Typically MOS tests require that the listeners and utterances use the same language, so it should not be expected of the model to perform well across languages unless extra consideration or data is provided. Lastly, it should be considered that language, culture, technology and perception are not frozen objects, but are constantly evolving and interacting. This hints that a model should consider more than the signal alone, such as attributes of the raters.

Opinion score data is scarce for quality assessment problems compared to other fields such as speech synthesis, because human rating data does not occur as naturally as raw speech and usually must be collected manually by conducting expensive tests. Bayesian models provide a solution to deal with limited and out-

lying data by using prior distributions to provide a baseline that can be updated instead of fully depending only on the observations. For example, there are often 24 listeners per utterance. If all of them happen to rate the score as a '5', it is not reasonable to assume there is zero variance and all future raters will rate it a '5' as well with total confidence. To handle these problems, we propose to use a Bayesian hierarchical model of an ordered categorical distribution to model individual opinion scores based on speech, listener, and language features.

The experiments for this chapter use a Bayesian model on individual rater score instead of a MOS-based model with no loss of generality. That is, such a model can be used to compute anything that a MOS-based model can. In addition, to provide models that are closer to real-world MOS estimators, we propose to fit models that include signal level predictors, to provide an indication of the marginal benefit of adding language and rater predictors.

The main contributions of this work are as follows:

- We fit a Bayesian model with ordered categorical distributions and truncated normal distributions to model individual scores, and compare these to mean opinion score models.
- We measure the effects of language and listener in the observations and the posterior samples.
- We compare various models and show that a language and listener-aware model have significantly lower error than the model without it, even after adding signal features.

An explicit non-goal should be stated: this chapter is not concerned with finding the best model and predictors that produce the lowest error. Instead, it uses relatively simple models that can be easily compared. The findings from this paper

should be useful for model design and input feature selection in other frameworks including existing MOS estimation tools and deep learning-based models that attempt to find the most accurate model.

This paper is presented as follows: first, related work is described. The next section describes the model, including the choice of individual scores, a causal model, and the specific Bayesian models that are considered. The next section describes the experiments, dataset, and results. A concluding section summarizes the paper.

4.2 Related Work

There has been work on using hierarchical Bayesian models of MOS that consider the heterogeneity of the data, evaluating the effect of speaker gender across multiple languages [45] and loudness patterns [51]. This work showed the effectiveness of Bayesian modeling given no signal level predictors. Signal predictors have allowed existing frameworks like ViSQOL or POLQA to predict MOS with an even higher accuracy. The next question to ask is whether there is a marginal benefit to adding language predictors.

Previous work has explored using multinomial models to model the distribution of all rater scores [47]. The multinomial distribution has advantages over modeling MOS directly and is suitable for certain applications such as using the distribution directly as a quality measure instead of MOS (and is able to generalize to a MOS-model). However, it does not capture information about the individual raters.

As mentioned in previous work [34, 47], point estimates are strictly less useful than distributions of the scores, which provide a measure of uncertainty. The uncertainty is useful for both the end user, who may wish to know the range of

scores to expect, but also to conduct statistical tests or compute probabilistic metrics. This is a separate question from whether or not to use individual scores or per-utterance MOS data in the model. Bayesian models are always probabilistic, so they always have a notion of uncertainty.

4.3 Model Design

In this section, the properties of a desirable model for opinion scores are discussed. The design of the model includes not only considering the type of model (e.g. Bayesian, deep learning, or linear regression), but also the options for what quantity to model (e.g. individual score or mean score), and what predictors to use (e.g. signal-level features and language metadata) given a hypothetical causal relationship between both the predictors and the outcomes.

4.3.1 Individual opinion score vs. mean opinion score

A model of individual opinion score is strictly more useful than a model of mean opinion score, because a mean opinion score can be calculated by grouping the posterior samples of the individual score model. The inverse, transforming a mean opinion model into an individual score model is not possible without a very lossy pseudoinverse.

The drawbacks of modeling an individual opinion score are twofold. The first one is that it requires and uses more data and parameters, since the model will be trained on many raters per utterance instead of a single mean score. For more interesting models, individual raters might be modeled to learn their biases, which increases the number of parameters in the model. The other drawback is that the user must aggregate the individual scores to compute a MOS or median score, which requires some additional design and computation.

4.3.2 Bayesian models

4.3.2.1 Parameter uncertainty

In some frameworks, the model parameters θ, such as mean $\hat{\mu}$ and variance $\hat{\sigma^2}$ are fit without a notion of uncertainty about each parameter. In other words, a point estimate $\hat{\mu}$ of a MOS implies that the model does not indicate a degree of uncertainty or expected error between $\hat{\mu}$ and the true μ. This missing uncertainty is not to be confused with the observed sample variance $\hat{\sigma}^2$ which is also a point estimate with undefined uncertainty for the true variance σ^2. Other frameworks consider the standard error of the sample mean and variance by assuming a given distribution that may or may not match the data. Point estimation of MOS is popular in existing frameworks (e.g. POLQA, and PESQ). Well-calibrated uncertainty is a useful quantity for the user. For example, the user may want to know the bounds of MOS, that is, what MOS value is unlikely to be exceeded for a given utterance for a specified quantile. Additionally, these point estimate models often fit a mapping function by minimizing mean squared error, which means that outliers are penalized. This causes the distribution of the predicted MOS to be under-dispersed when compared to the observed distribution of MOS. That is, extreme MOS values closer to 1 or 5 will be seen less often in the predictions than in the real data. This compression effect of the true distribution into the narrower predicted distribution will increase as the prediction error increases for point estimate models. Alternatives to models fit with MSE include maximum likelihood or quantile regression [62], which are useful and practical solutions, but have issues with the observed variance problem described below.

When subjective tests are performed, it is common to see the results reported with the sample mean $\hat{\mu}$ specified along with a 95% or 99% symmetric confidence interval $[\hat{\mu} - c\hat{\sigma}, \hat{\mu} + c\hat{\sigma}]$ where $\hat{\sigma}$ is the sample standard deviation of the opinion

score and c is a constant. This provides a basic uncertainty estimate about the MOS, but again, this uncertainty assumes that $\hat{\mu}$ and $\hat{\sigma}$ have no error. Additionally, for the discrete 5-point scale used in subjective testing there are problematic properties of this formulation of uncertainty. For example, the symmetry of the confidence interval is not appropriate near the extremes of '1' or '5'. It can also be seen that the confidence interval is difficult to use with a smaller number of ratings, because the variance becomes less reliable. For example, if the study has only 5 raters and they all rate a '5', the variance will be zero, and the confidence interval will also be zero. This problem of variance estimation still exists if the raters give different scores with non-zero variance, although it is less obvious.

Bayesian models resolve the issues seen in the point estimate models and confidence interval analysis. In the Bayesian framework, the appropriate model will provide uncertainty estimates that are tailored to the 5-point outcome space by looking at the distribution of posterior samples. This means they will produce asymmetric credible intervals if the data requires it. The model will also start with a prior selected by the system designer that contains a reasonable distribution for each parameter (with non-zero variance), which handles the problem of sampling limited data where the sample variance does not capture the true variance.

4.3.2.2 Entropic rationale for language predictors

Next, we consider the basis in Bayesian models that suggests that adding language as a predictor will produce more accurate estimates. Given observed scores and any collection of observed features and latent (e.g. learnable) parameters, Bayes' formula provides a description of the uncertainty of each opinion score value as

$$P(score|features) = \frac{P(features|score)P(score)}{P(features)}. \tag{4.1}$$

Note that this can also be written using the joint distribution on the right hand side as

$$P(score|features) = \frac{P(features, score)}{P(features)}. \tag{4.2}$$

By modeling the joint distribution of features and scores the probability of each score can be inferred. A Bayesian model is able to model a joint distribution of the features (including features that are parameters and hyperparameters), and is therefore able to give a probability estimate that is precise to the extent that the joint distribution of features and scores has a low entropy.

Previous studies have shown that there is an effect of language on scores [41, 45], so it seems reasonable to infer that score and language are not unconditionally independent, and that the mutual information $I(S; L)$ between score and language is positive. However, score and language may be conditionally independent given the input features (which may contain language information). It follows then that models that only use features with signal-level information, without information about the language will have a strictly higher entropy unless the score is conditionally independent of them given the signal-level features, since for variables S as score, F as some chosen set of features, and L as language,

$$H(S|F) = H(S|F, L) + I(S; L) \tag{4.3}$$

from which it follows that

$$H(S|F, L) \leq H(S|F), \tag{4.4}$$

with equality when the features are chosen such that scores are conditionally independent of language information given these features. A higher entropy in score

conditional on features that do not contain language information means that the accuracy of the predictions should be worse, because the lowest error a model conditional on these features can achieve will be higher due to the increased uncertainty.

The language information may be present in fine-grained signal features, but typically for opinion score estimation, the signal features are coarse (e.g. a one-dimensional scalar representation of similarity between the degraded and reference signals), and furthermore, the signal only contains the language of the utterance and not necessarily that of the rater (although for the purposes of this work the main focus is on native language testing). The simplest way to obtain conditional independence between score and language is to add information about the language to the feature set. Alternatively, depending on the causal assumption (described in subsection 4.3.4) that other variables fully contain the language information, such as rater identifiers, it can be sufficient to simply include information about each individual rater without pooling them by language, although including both may be desirable for other reasons.

4.3.3 Model outcomes

An individual rater score is represented by a discrete value from 1 to 5. These values are ordered by perceived quality. Other Bayesian models have used normal distributions to model MOS [45]. A normal distribution does not model the boundaries at 1 and 5, and further, is continuous, while individual rater scores are discrete. Another work uses multinomial distributions to model histograms of scores for each utterance [47]. The multinomial model has the advantage of capturing a distribution of each of the five scores discretely per utterance, but is not able to model an individual rater's bias over multiple utterances, which is necessary for the experiments involving individual raters. There are several options

for the model that are reasonable given the individual rater problem. Here we consider three different types of models based on their outcomes.

4.3.3.1 Ordered categorical outcomes

The ordered categorical distribution, also known as the ordered logit or ordered logistic distribution, describes an ordered categorical (discrete) variable (such as an opinion score that has categories such as 'bad', 'poor', 'fair', 'good', and 'excellent' with a notion of order). Given N categories, there will be $N-1$ log-cumulative odds $\kappa_1, \kappa_2, ..., \kappa_{N-1}$, from which the linear categorical probabilities $p_1, p_2, ..., p_N$ can be derived via softmax. Each individual logit is made cumulative by using the probability that the outcome is less than a given cutpoint n, and is given as

$$\begin{aligned} \kappa_k &= log \frac{Pr(y_i \leq k)}{1 - Pr(y_i \leq k)} \\ &= \alpha_k - \phi_i, \end{aligned} \tag{4.5}$$

where k is the category index and i is the observation, and ϕ_i is any model based on predictors x, such as a simple linear model with one parameter β:

$$\phi_i = \beta x_i. \tag{4.6}$$

Additional (possibly non-linear) terms will result from adding more predictors. For the opinion score dataset, the features and model of ϕ depends on language, individual rater, and signal similarity and is discussed in section 4.3.5.

4.3.3.2 Truncated normal outcomes

The normal distribution is not bounded, which can be a problem for opinion scores. Rounding the outcome of a normal distribution to the opinion score range

can produce undesirable results. An alternative to the ordered logistic distribution that is appropriate for the opinion score problem is the truncated normal distribution, which specifies a PDF that has support entirely within a desired range. This is similar to the normal distribution, but has a lower and upper bound (which would be 1 and 5 for the MOS problem) that asymmetrically truncates the outcome. The probabilities after truncation are normalized by the truncated density so that the resulting PDF sums to 1.0. The main advantage of the truncated normal distribution over the ordered logistic is that it is difficult to formulate an ordered logistic distribution that allows for regression on more than the ϕ parameter. That is, the cutpoint parameters that control the thresholds of each category are typically not indexed per group (e.g. all raters share the same cutpoints, and only differ by their individual ϕ offset). The truncated normal should provide a more flexible model for opinion score estimation, since some raters may have more or less variance but the same mean. The disadvantage is that the truncated normal distribution outputs continuous scalars, which will not match the discrete nature of the opinion scores. As a result, the truncated normal is more useful for applications involving prediction as opposed to simulation.

4.3.3.3 MOS outcomes

A third option is to model the mean of the opinion score per utterance directly. Here, the data is aggregated before the model is fit, so there is no concept of individual rater. However it is still possible to use language predictors with this type of model. The formulation of this model is very similar to the truncated normal outcome model of individual scores, using the utterance MOS instead of individual scores.

4.3.4 Causal model of opinion score

For pure prediction problems, it is not strictly necessary to have a thorough understanding of the causal model when designing a regression model. That is, the system designer will generally achieve a more accurate prediction by adding as many predictors as possible, without needing to worry about causation versus correlation, or confounders. Deep learning presents many useful examples of this by using as many input features as possible. However, many popular MOS estimation frameworks use only signal level features, which ignore information about the rater and their environment.

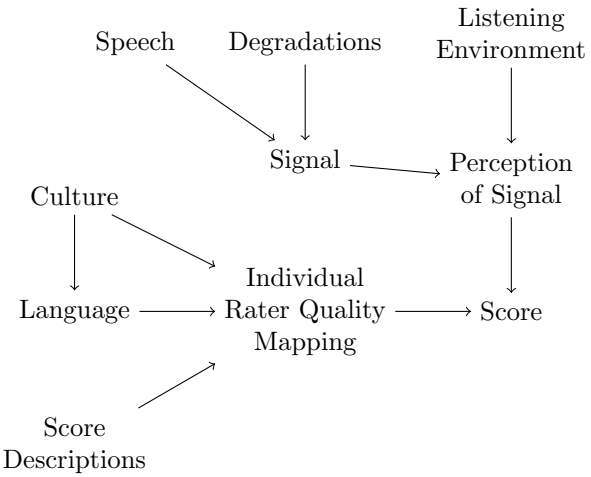


Figure 4.1: A plausible causal DAG for individual ratings. The acoustic properties of the test signal are only a part of the process that determines the score.

It may be useful to consider a plausible causal DAG such as the one in Figure 4.1 to entertain potential non-signal predictors. In the DAG, several variables independent of the speech signal are considered. 'Culture' contains many attributes, and is an unobserved variable, but since culture usually determines (native) language, language may serve as a proxy for culture, which may include rating tendencies. The 'Environment' variable pertains to the listening environment that

the rater conducts the test in, and contains many potentially unobserved attributes, such as the listening equipment and presentation order of the signal (which may cause listening fatigue), and even the weather during the test. The score description labels, which are text in the native language presenting along with each rating level (e.g. 'poor' in English, 'warui' in Japanese, or 'mÃ©diocre' in French for the score '2'), also affects the likelihood of a certain score, because the text may imply different qualities in different languages. Furthermore, the choice of labels may imply a nonlinear scale within a single language if the labels are not perceived to be equidistant. The culture and language problem has been studied to a considerable degree, for example, in [44], with some solutions using models specific to a certain language [76, 77].

If unlimited computing capacity is available, it would be desirable to add as many of these predictors to the model to obtain the highest accuracy. But in practice, compute and data are scarce resources. Furthermore, the causal process is noisy, and it is unclear to what extent each feature is actually useful in a predictive model without experimentation. For example, neither culture nor region uniquely determines language. An experiment is needed to see to what extent information about language improves the accuracy of the model, and so on for the other variables.

4.3.5 Features and parameters

MOS estimators typically will include features that are extracted from the speech waveform. For this problem it is appropriate to use the neurogram similarity index measure (NSIM), which is a 1-dimensional indicator of similarity over all frequency bands and time between the reference and degraded signals. NSIM has shown to be useful for early versions of ViSQOL [24], and the one-dimensional property allows for a relatively simple model with a single parameter related to the signal. Signal predictors which provide more modeling power and less aggrega-

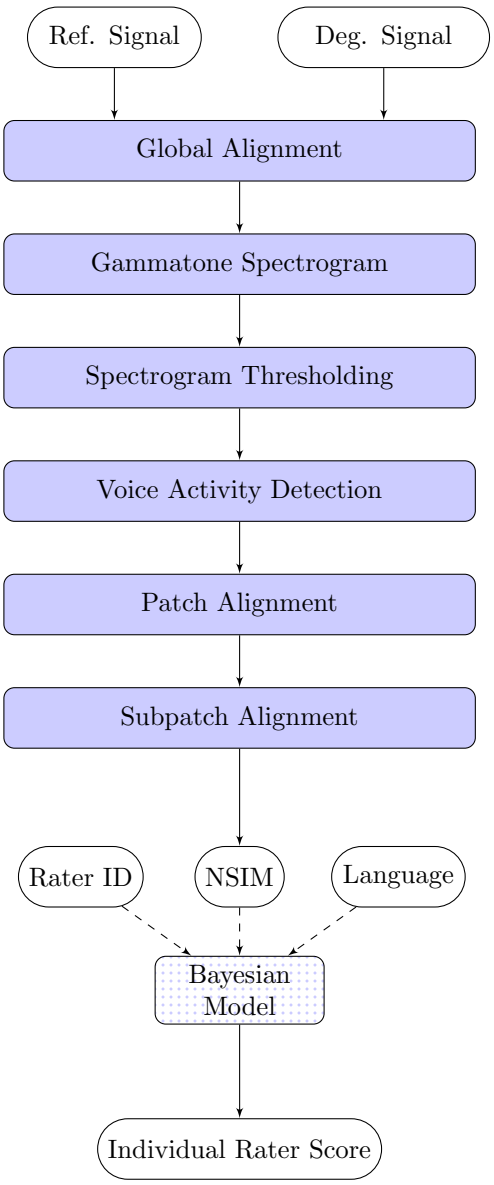


Figure 4.2: A system diagram showing the features that are used as predictors in the model. The dashed arrows represent optional predictors.

tion certainly exist (e.g. multiple frequency band NSIM [9], mel-spectrogram, or WARP-Q [36]). However, the purpose of this study is not to find the most useful signal-level descriptors, but instead to find the effects of features that are external to the signal, such as the individual rater and language bias. To this purpose,

it is desirable to keep the signal level features minimally complex. Figure 4.2 illustrates the features that are used by the Bayesian model as predictors.

As previously mentioned, individual raters have a bias and variance that differs from other raters. A rater identifier feature that is unique for each rater is added to the model to allow it to be aware of this bias. Similarly, language identifiers can be a feature that uniquely identifies the language. Because the raters of a given language forms a group, it is sensible to apply a hierarchical model to share information between individual raters. For the purposes of this study, this 'language' feature will be a laboratory identifier where the native language is used to test, and also encompasses other factors in the entire test environment such as the culture of the laboratory, the listening equipment, and so on. Each rater and language identifier is used as an index variable with normal priors that linearly influence the ϕ offset for the ordered logit model, and an exponential model for NSIM, as was found to be useful in [18]. The prior for ϕ_i can be described for individual observation i, rater j, and language k, and NSIM observation x_i as

$$\begin{aligned} \phi_i &= \alpha_j + \gamma \\ \alpha_j &= \text{Normal}(\mu_k, z) \\ \mu_k &= \text{Normal}(a, b) \\ \gamma &= e^{\beta(x_i - \theta)} \\ \beta &= \text{Normal}(c, d) \\ \theta &= \text{Normal}(e, f), \end{aligned} \tag{4.7}$$

with user-defined constants a, b, c, d, e, f (to be found with prior predictive checking). Note that the priors with subscripts denote that there is one prior for each item in the group, e.g. the μ_k indicates multiple Normal priors, one for each of the k languages, so the model will fit each item of the group separately. The trun-

cated normal model is formulated similarly, with ϕ_i being used as the mean for the truncated normal distribution, with an additional prior for variance.

4.3.6 Computing MOS from individual score models

An individual score model as described above outputs a discrete score for each utterance and rater. MOS is often used as the mean aggregated over utterances, or a collection of utterances within a certain test condition. Given an individual rater's score probability $p_r(s|x)$, the true MOS over a set of utterances X and raters R can be computed as

$$\begin{aligned} \mathrm{MOS}(X, R) &= \mathbb{E}[s|X, R] \\ &= \frac{1}{|X||R|} \sum_{x \in X} \sum_{r \in R} \sum_{s \in S} p_r(s|x) s \end{aligned} \tag{4.8}$$

where s is the score in the set $S = \{1, 2, 3, 4, 5\}$. For example, the utterance MOS uses X with a single element (a single utterance), and the condition MOS uses X with all the utterances in the condition.

$p_r(s|x)$ is an unknown quantity that must be estimated. In a Bayesian model each posterior sample is a sample from the joint distribution of all the parameters, so multiple samples will produce different likelihoods for the same utterance and rater pair. So in practice estimating $p_r(s|x)$ requires multiple samples. One way to do this is to sample the model's joint distribution many times to obtain the probability by converting the histogram into a probability. However, since we are interested in the expectation over X and R, the process can be further reduced by simply taking the mean of the posterior scores as the estimated MOS. In other words, although the observed data ratings have each listener rate each utterance once, in the posterior samples each listener 'rates' each utterance hundreds or thousands of times, and the MOS can be estimated by taking the average of all

samples.

4.4 Experiments

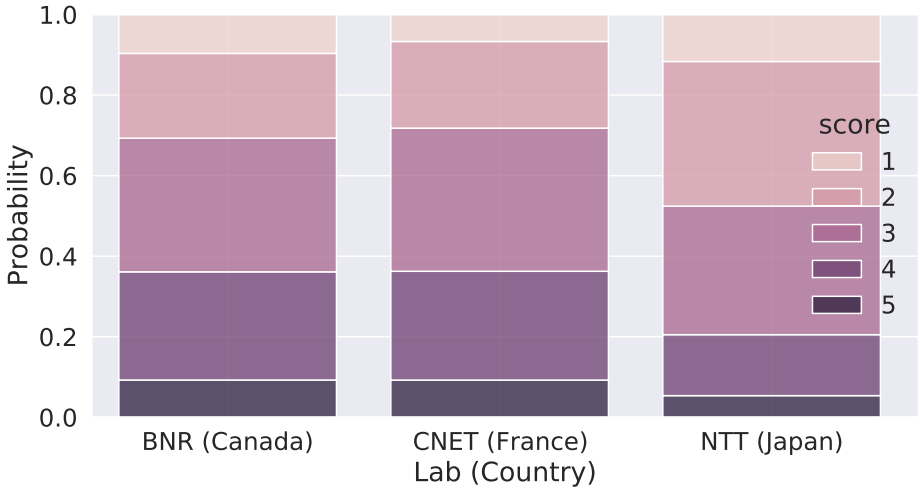
In this section, an appropriate dataset is analyzed, and experiments that use proposed models are presented.

4.4.1 Dataset

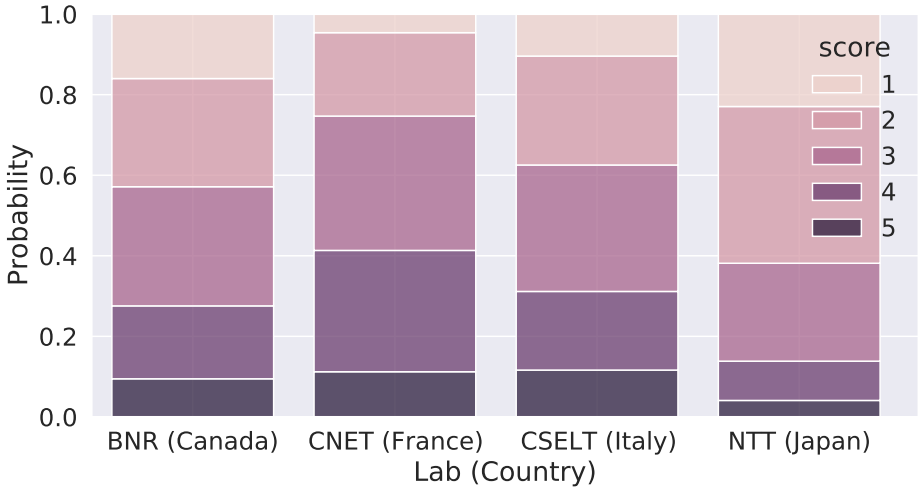
The ITU-T P. Supplement 23 dataset (experiments 1 and 3) is well-suited for this experiment. It is conducted across four different languages and laboratories with all listeners being native speakers of the language used in the utterance. Additionally, the recording conditions, and the signal processing chain including the pre- and post-processing of the signals are well-documented, and each lab conforms to the shared procedures. Lastly, all of the labs in a given experiment tested the same conditions (e.g. street noise at 6dB SNR), so the results between the labs should be comparable. These properties of the dataset enable this experiment to measure the effect of language. There are 24 listeners in each experiment and laboratory combination, and each listener rates many utterances, with the order of presentation randomized according to 4 different randomization patterns. All of the data for individual raters is recorded in the dataset (i.e., the data is not aggregated into MOS).

4.4.1.1 Analysis of distributions by language

In the P. Supplement 23 dataset, experiment 1 tests the performance of low bitrate codecs with transmission standards. Experiment 3 tests the effects of channel degradations. It may be useful to consider the distribution of the observed data for both of these experiments.



(a) Experiment 1



(b) Experiment 3

Figure 4.3: Distribution of individual rater scores in the ITU P Supp. 23 dataset for different languages. Each laboratory conducted the experiment under the same conditions in the native language. 'BNR' is Bell Northern Research in Ottawa and uses English.

Figure 4.3 compares the observed distribution of each rating between labs and experiments. There are remarkable differences between the different languages. Japanese raters tend to rate with very few 'fives', and many 'ones'. French raters are the opposite, being the least likely to rate as 'ones', and the most likely to rate

'fives'. It is expected that the content of the experiment affects the distribution of scores. So while it is expected that the score distributions within a language change in different experiments, it is interesting to note that the distribution preserves some properties, such as the relative biases of the languages.

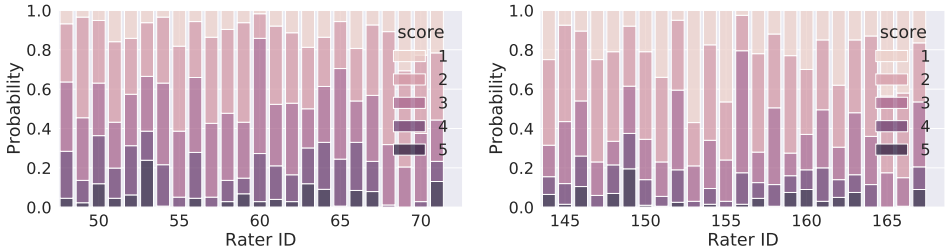
4.4.1.2 Analysis of individual rater distributions

Looking at the individual rater distributions within each language naturally contains all of the information to describe the distribution of the language, since the language data is simply the set of all raters of the language. But it also contains some additional information pertaining to each individual rater's tendencies that is not in the language information alone. In figure 4.4 it can be seen that there are language-level biases, and within a language that there are individual rater biases. So it appears to be reasonable to construct a model that captures both language and rater information.

4.4.2 Model specification

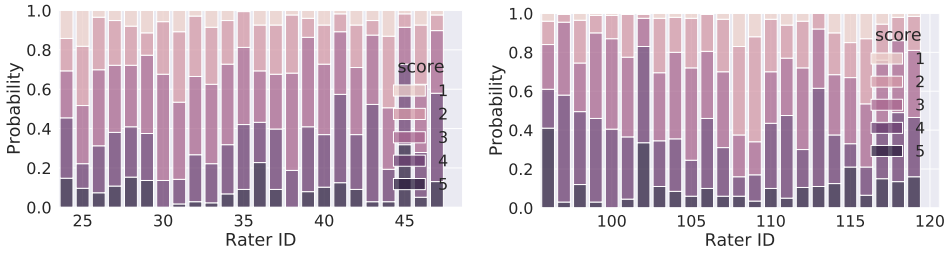
Several models are considered to show the effects of different predictors. These models and the features they use are described in table 4.1. The most basic model 'Baseline', uses no predictors (i.e. no input features) to predict a score. This model will obviously not be able to predict the scores of individual listeners or utterances accurately, but it should be able to model the overall distribution of the input. It serves both as a control that other models can be compared against, as well as to test whether the distribution of observed ratings has a dispersion that is captured by an ordered categorical model with the provided priors.

The most complicated model 'LangRaterNSIM' is the one that uses the features described in 4.3.5 that we expect to contain information about the outcome score.



(a) Experiment 1, Japanese

(b) Experiment 3, Japanese



(c) Experiment 1, French

(d) Experiment 3, French

Figure 4.4: Distribution of scores for each of the 24 raters in the ITU P Supp. 23 dataset for Japanese and French amongst the two different experiments. Japanese and French raters are the most different from each other in that they tend to rate low and high respectively. The distributions are different enough to be visually apparent, but have enough variance that there is overlap - the highest rating Japanese rater tends to rate higher than the lowest rating French rater.

More specifically, the predictors are listener and language identifiers (indices) along with a scalar NSIM predictor that indicates signal similarity between the original reference and the degraded utterance that the rater has scored. Additionally, a model called 'Order' with a single predictor uses the logarithm of the presentation order of the utterance to predict the score.

Additionally, we fit two models (NSIMMOS and LangNSIMMOS) that have the same predictors as NIM and LangNSIM, but are fit to pre-aggregated utterance MOS data directly instead of using individual rater score, and a truncated normal model (LangRaterNSIMTrunc) that has the same predictors as LangRaterNSIM. All models that do not end with either 'MOS' or 'Trunc' are ordered logistic

Table 4.1: Models and predictors

	Predictors			
Name	Language	Rater	NSIM	Order
Baseline				
Order				✓
Lang	✓			
LangRater	✓	✓		
NSIM			✓	
LangNSIM	✓		✓	
LangRaterNSIM	✓	✓	✓	
RaterNSIM		✓	✓	

models.

4.4.3 MOS results

We use the method described in section 4.3.6 to estimate MOS for each utterance, condition, and lab-specific condition. Table 4.2 shows the error and correlation coefficients for each model for three types of aggregation. The most common of these in the literature is aggregation by condition, which generally produces the lowest error and highest correlation, followed by aggregation by utterance, which produces a higher error due to the smaller number of samples. To better understand the effects of language, an aggregation by condition within each language is also presented. The models that add language predictors have a relatively large improvement for the aggregation by language and condition (0.562 vs 0.464 RMSE for NSIM vs LangNSIM), and the differentiation of language is evident in figure 4.7. Additionally, figure 4.6 visualizes the predictions in joint plots with the ground truth MOS at the utterance level, where the language effect is also evident.

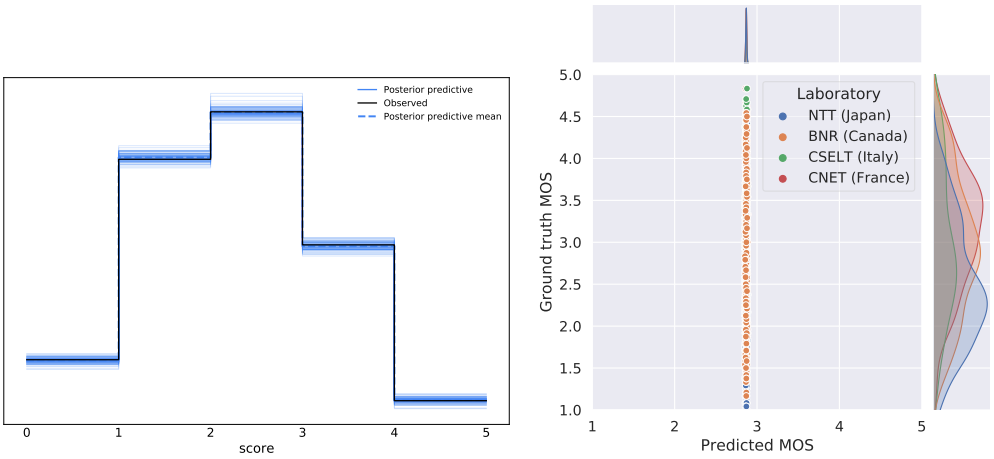
The 'Baseline' model has no predictors at all and its MOS predictions naturally converge on the MOS over all utterances, which is a value just below 3.0. The distribution of the unaggregated individual score predictions matches the input

distribution well. This can be seen in figure 4.5.

For comparison purposes ViSQOL exponential is added a an anchor that should align with the Bayesian NSIM model, as it is a model that maps a single dimension of physical similarity (e.g. mean similarity across all frequency bands and time) to quality that is analogous to the NSIM predictor used by the Bayesian models. PESQ and POLQA are added as standard models that are known to perform especially well on this dataset. The Bayesian NSIM model performs on par or slightly better for RMSE than ViSQOL exponential [18], indicating that it makes reasonable usage of the information in the signal predictor. As mentioned earlier, the purpose of this study is not to compare a simple Bayesian model with state of the art models that have complex predictors (e.g. multi-dimensional predictors with similarity for multiple frequency bands), but to exploit the simplicity of the Bayesian model to test the marginal benefit of adding language features on top of signal features. For this reason the POLQA scores are not directly comparable, and it is not surprising to see that POLQA has lower RMSE across the board.

Table 4.2: Model Comparison

	Utterance			Language+Condition			Condition		
Name	RMSE	Pearson	Spearman	RMSE	Pearson	Spearman	RMSE	Pearson	Spearman
Baseline	0.829	0.043	0.035	0.778	0.106	0.090	0.705	0.151	0.123
Order	0.829	-.0326	-0.033	0.779	-0.029	-0.035	0.706	-0.039	-0.033
Lang	0.790	0.303	0.295	0.734	0.333	0.321	0.707	-0.050	-0.023
LangRater	0.785	0.321	0.319	0.729	0.362	0.351	0.704	0.105	0.136
NSIM	0.684	0.568	0.551	0.562	0.717	0.700	0.431	0.858	0.856
NSIMMOS	0.681	0.570	0.554	0.560	0.712	0.699	0.427	0.858	0.856
LangNSIMMOS	0.613	0.674	0.664	0.457	0.821	0.812	0.404	0.850	0.840
LangNSIM	0.612	0.670	0.659	0.464	0.822	0.813	0.415	0.851	0.843
RaterNSIM	0.603	0.687	0.678	0.446	0.840	0.839	0.407	0.861	0.853
LangRaterNSIMTrunc	0.603	0.686	0.677	0.444	0.839	0.836	0.405	0.858	0.859
LangRaterNSIM	0.601	0.689	0.679	0.446	0.839	0.837	0.409	0.859	0.852
ViSQOL Exponential	0.740	0.498	0.488	0.652	0.685	0.676	0.562	0.838	0.836
PESQ NB	0.494	0.806	0.786	0.422	0.842	0.822	0.256	0.935	0.922
POLQA SWB	0.524	0.783	0.756	0.430	0.837	0.808	0.182	0.970	0.972

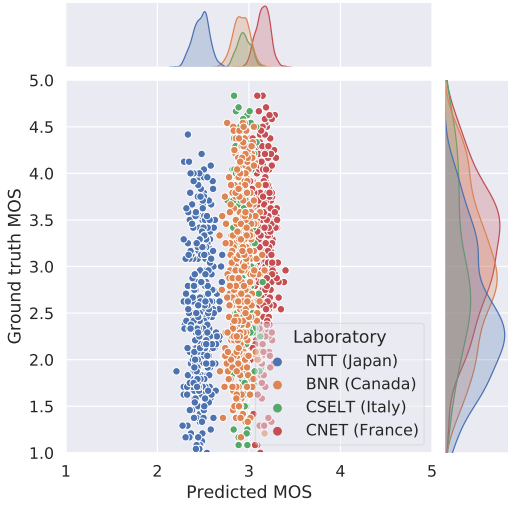


(a) (b)

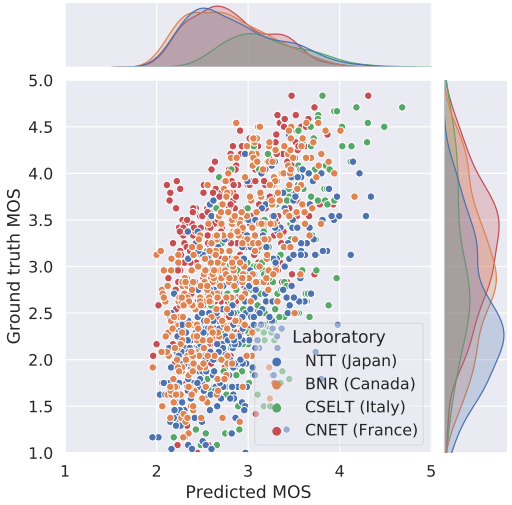
Figure 4.5: The baseline model is able to model the distribution of scores in the posterior samples (a). The individual predictions have no specific information, so although individual predictions span the full score range, the MOS of any individual utterance converges on the global MOS as shown in the joint plot (b).

4.4.4 Model validation and comparison

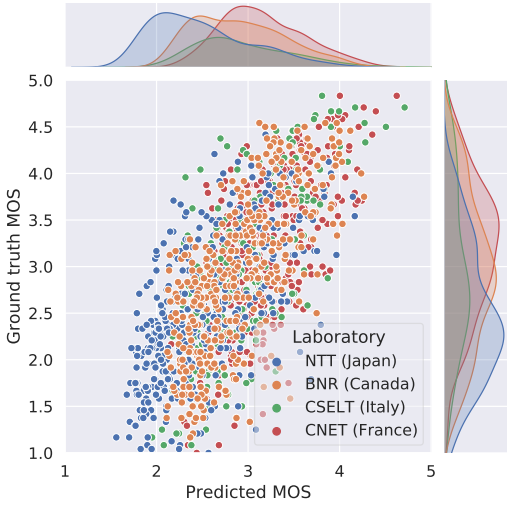
Model design, validation, and comparison of Bayesian models goes beyond looking at error and correlation. It involves an interactive process that is facilitated with prior predictive checks, posterior predictive checks, and verification that the posterior samples are useful. For example, inspecting the posterior samples against the observed appears to be reasonable at the global outcome level in figure 4.8a. These posterior samples can also be used to create a measure of uncertainty if the models are reasonably calibrated by confirming that for a certain quantile, approximately that many samples are underestimates (e.g. the median quantile should have half of the posterior estimates above the observed median.) The estimates at the global and language level appear to be reasonably calibrated, but this is not so for all individual raters, so the uncertainty should only be useful when aggregating over utterances, as is the practice with MOS.



(a) LangRater



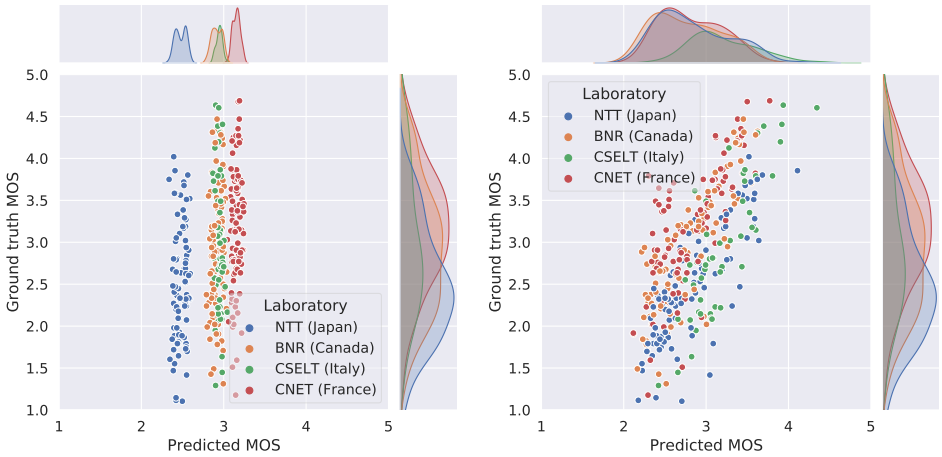
(b) NSIM



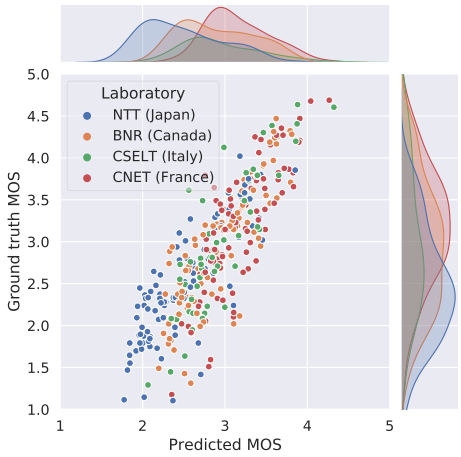
(c) LangRaterNSIM

Figure 4.6: Joint plots of per-utterance mean opinion scores for select ordered logistic models. Because the LangRater model has no signal predictor, it can only estimate language and rater means. The NSIM model has only a signal-level predictor and is not able to capture the differences in languages. LangRaterNSIM with signal, language and rater predictors improve on the NSIM model differing modes of each lab, showing the marginal improvement over NSIM.

Bayesian models are typically fit on the entire dataset, unlike deep learning models



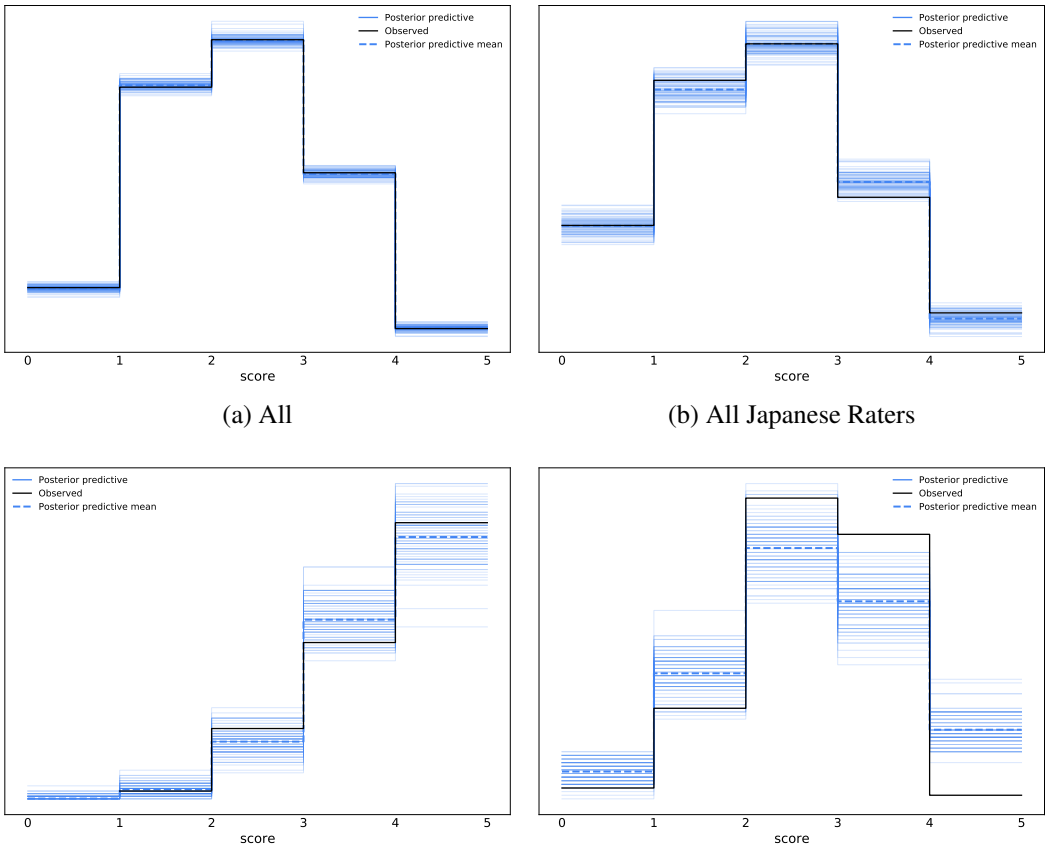
(a) LangRater (b) NSIM



(c) LangRaterNSIM

Figure 4.7: Joint plots of per-language-and-condition mean opinion scores for select ordered logistic models. Aggregating over conditions, which are a larger group than utterances, reduces the error and increases the correlation. The effect of language and rater can still be seen in the marginal distributions as in figure 4.6.

that split the data into a train and a test set. There are several reasons for this. Bayesian models are always probabilistic models, and other metrics that rely on this property can be used to check that the model is not invalid and that it is able to predict out of sample data, such LPPD, PSIS, WAIC, $\hat{r}$, and visual inspection of the chains and the posterior predictive distribution. It is common to compute



(a) All

(b) All Japanese Raters

(c) Individual Japanese Rater A

(d) Individual Japanese Rater B

Figure 4.8: Posterior samples from the LangRaterNSIM ordered logistic model compared to the observed scores for the overall data and different subgroups. The posterior overlaps well for the global (a) and language group (b), and reasonably for (c). The individual rater B (d) is an imperfect fit, presumably because the model uses shared cutpoints for all raters and is not able to increase the mean without increasing the likelihood of the '5' score.

.

estimates of 'leave one out' cross-validation (LOOCV) to check for out of sample predictive accuracy without dividing the dataset into train and test splits. PSIS and WAIC are popular estimates for this purpose. These metrics also measure predictive power, i.e. how accurately it will predict an out of sample observation.

Since some of the models in this experiment have different outcomes (i.e. MOS

models versus individual score models), not all models are suitable for relative comparisons of WAIC and PSIS. For select models where the comparison is sensible, table 4.3 shows the values for WAIC and PSIS, which converge on the same values within a decimal point. Here, higher values indicate more accurate out of sample predictions. The ranking matches the original models RMSE rankings. If the two models have the same predictive power the model with a language predictor is preferable to allow for predicting language effects.

The statistical significance of these results should be discussed. ITU-T Rec. P. 1401 [78] provides statistical tests for comparing the significance of model differences in RMSE. Under these tests, the RMSE difference is significant between 'Baseline' and 'Lang' (p = .0395), between 'Lang' and 'NSIM'(p = 8.01e-8), and between 'NSIM' and 'LangNSIM'(p = 2.58e-5), but not between 'LangNSIM', and 'LangRaterNSIM' (p = 0.254). This is more evidence in favor of language being a useful predictor on top of a signal predictor like NSIM, and that rater information may be useful, but not significant under this test. However, the significance of predictive power can also be looked at on the outcome scale of individual rater scores instead of aggregating the raters into a single value. For this purpose, the PSIS/WAIC analysis also provides the function of a statistical test for the significance of the out of sample predictive performance. It is also important to point out the relatively small standard errors in table 4.3 due to the large amount of data. This shows that most of the models do not overlap within three standard errors, and that there is a real benefit to predictive power from including each of the features in this order. The exception is the overlapping LangRaterNSIM and LangRaterNSIM, which is expected due to the rater data fully containing the language data.

Another difference from deep learning is that in Bayesian models, there are fewer parameters which make overfitting less of a risk, and models that have too many

Table 4.3: Predictive power and significance

Name	WAIC/PSIS(LOO-CV)	Standard Error
Baseline	-48109	74.30
Lang	-47217	82.29
NSIM	-45085	90.51
LangNSIM	-43717	97.56
RaterNSIM	-40487	110.07
LangRaterNSIM	-40485	110.49

parameters are often non-identifiable (which would fail the $\hat{r}$ test, or would be so specific as to not be useful (e.g. a separate model for each observation). The hyperparameters of a Bayesian model are the distribution parameters of the root level priors, and these are typically set at model creation time, or interactively, by looking at the prior predictive distribution (which does not involve the data). Some of the more accurate models in this experiment have very few parameters (LangNSIM only has 10 parameters: 4 for the cutpoints, 4 for the language offsets, and 2 for NSIM slope and intercept).

The models were fitted using Hamiltonian Monte Carlo (HMC) which efficiently samples the posterior distribution with a No-U-Turn sampler [79] using the TensorFlow Probability framework [80]. Multiple chains are used in HMC, and the typical check of involves visual inspection of the chains as well as verifying that $\hat{r}$ values are reasonable (i.e. near 1.0) to check that the chains converge and do not get stuck on some values, which would indicate that the posterior was not well-explored. For the models studied in this experiment these checks have passed, and the posterior distributions of the parameters and their chains appear healthy as shown for the LangRaterNSIM model in figure 4.9.

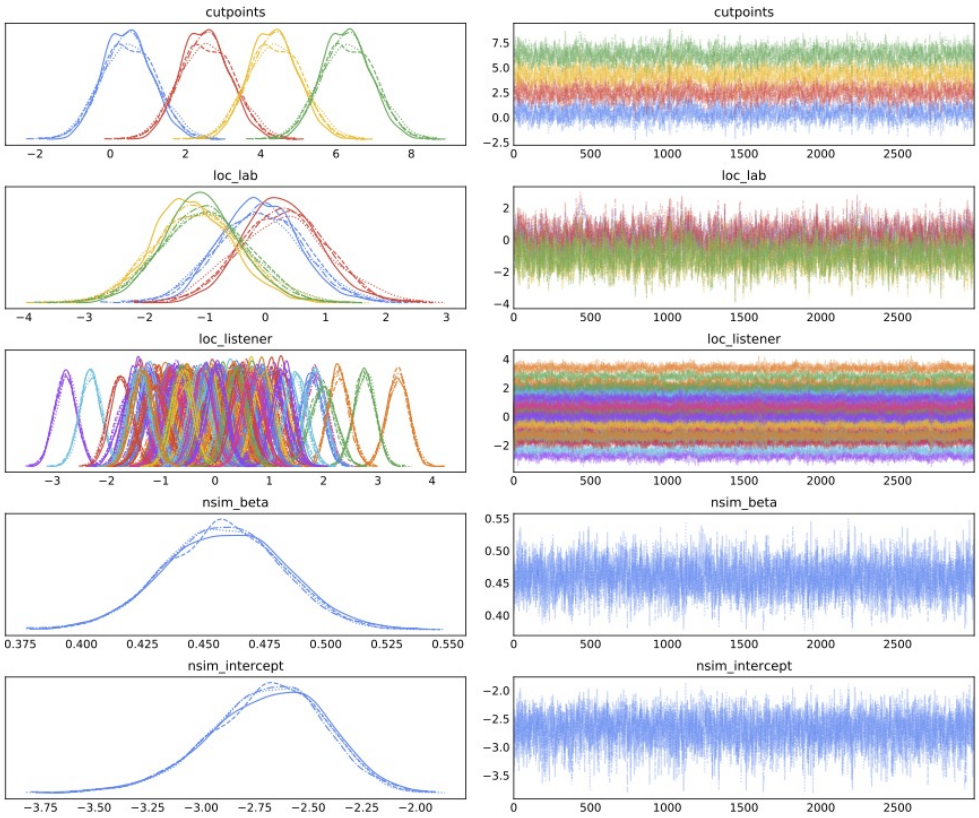


Figure 4.9: Traceplot of parameters for LangRaterNSIM model. The smoothed posterior samples form distributions for each indexed parameter, shown on the left in different colors (e.g. each language/laboratory is a different color) with each line-style (e.g. dashed, solid) representing the chain index. The Gaussian noise-like patterns on the right indicate the posterior for each chain and index is being explored in a healthy manner. The overlapping chains of the same color show that the model is identifiable.

4.5 Discussion

4.5.1 Effect of language and raters on opinion scores

In table 4.2 it can be seen that adding language predictors improved RMSE over a baseline model with no predictors (from 0.829 to 0.790) and correlation (from 0.043 to 0.303). Adding rater information improves it further. Furthermore, the

improvement when adding language and rater information is still significant on models that have signal level predictors (from 0.684 to 0.601 RMSE, 0.568 to 0.689 Pearson), which indicates that coarse signal predictors (of the type used in typical MOS estimation tools) cannot tell the full story about opinion scores. This is consistent with the observed opinion scores grouped by language in the data that compare the same degradations in figure 4.3.

The findings of this study show that a correlation between language and score exists, which agrees with previous work, but also shows the increase in modeling power when including language as a predictor. It has not been concluded whether the differences in scores between languages are cultural responses or related to perceptual quality. The causes behind the bias are related to general item response theory and psychology, and require more complicated experiments such as cross language studies with bilingual listeners to resolve, as well as studies that consider ratings that are completely unrelated to speech quality (e.g. restaurant ratings).

For example, the finding of a lower bias in Japanese scores does not reveal whether the 'true' quality of Japanese speech is lower, or that Japanese raters tend to rate a given quality lower than other raters. That question remains unanswerable with the current study. It is plausible that the distribution of phonemes in the language influences the score, just as it is plausible that there is a culture to rate things lower. In other words, the subjective test only measures the categorical ratings from 1 to 5, and how a different quality level is mapped to these numbers is unobserved. However, leaving the elusive unobserved 'quality' aside, it is possible to infer equivalencies between scores for different languages and signals. For example, for a certain kind of degradation, the expected Italian score is X, and for Canadians it is Y.

This does suggest that opinion scores should not be compared absolutely be-

tween different languages. The ITU-T specifications for P.800 listening tests [1] also gives strict recommendations for comparing MOS that excludes the cross-language case. It should be noted that while the model that was fit in this study has the ability to answer cross-language counterfactual questions such as 'what score would an Italian rater assign to this English utterance?'. However, there are zero occurrences of cross-language ratings in the particular dataset used in our experiments, so it should only be used with the due amount of apprehension.

Both individual rater models and MOS models can be made language aware, as has been done in the experiment. The findings with improved accuracy for language based models suggest that language predictors should be added to a scoring model, unless there is only a single language present in the data, and the model will never be used to predict scores for other languages. Since language is being used here as a proxy for culture, it may be useful to include other cultural covariates such as region and year if there is variation within languages in the data. To test the hypotheses of this experiment it was sufficient to use one laboratory per language at a single point in time. This also means that the resultant models may not generalize to other populations within the same language. As mentioned previously, the purpose of this work is not to obtain the most generally useful and most accurate model, but to answer the questions about the effect of language and raters in opinion score data.

The RaterNSIM model performed similarly to the LangRaterNSIM, which has additional parameters for language. This is consistent with the causal model DAG given in figure 4.1, because the individual rater blocks the path between language and score, so score and language are conditionally independent given the rater information. However, the advantage of the model with language parameters is that it is straightforward to simulate or answer counterfactual questions about new raters in the same language or to 'translate' raters into other languages.

4.5.2 Individual score versus MOS models

For the purposes of this work, we are only concerned with identifying the distribution of each rater's scores with a model that is given information about each rater, by also fitting a distribution of raters. Having fit such a model, it is then possible to predict the behavior of each rater with a higher accuracy. Two types of individual rater models were considered: the ordered logistic model and the truncated normal model. The truncated normal model achieved a lower error, presumably because in the truncated normal, the variance and mean are modeled as parameters for each rater, but in the ordered logistic model this is not the case because of the cutpoint formulation (only a logit offset, or 'location' parameter is modeled at the individual level). The ordered logistic distribution may still be useful, depending on the application. For example, unlike the truncated normal distribution, the ordered logistic distribution captures the discrete nature of categorical opinion scores, which may be desirable if the goal is to generate simulated data for individual raters.

The experiment compared models fit to MOS data as well as individual rater data. The experiments found that individual rater models are able to match and exceed the accuracy of MOS models because of the extra information they have, although the difference is not very large (.613 vs .601 utterance RMSE). The variance of individual raters is an important piece of information that is discarded in most MOS models, but is accounted for in individual rater models. Additionally, the bias or expected rating for individual raters is discarded by MOS models. This means that the MOS model will spuriously attribute individual rater bias to other predictors, increasing the error, although for this experiment it was not significant (the MOS models performed similarly to the individual rater models). For example, suppose there are two utterances of equivalent quality, and a rater that

consistently gives low scores. If the rater rates the first utterance, but does not rate the other, it should be expected that the MOS for the first utterance will be lower than the second. The MOS model, which does not have access to individual rater information, will not be able to recognize that the first utterance's lower score is due to the pessimistic rater's presence and will instead will attribute it to the signal predictors, or produce a wider uncertainty on all utterances of this quality. In contrast, the individual score model will handle this case by attributing the difference in scores to the pessimistic rater.

If a distribution of the individual rater is modeled, the model can answer questions that a MOS model cannot, such as 'what would the median rater score this utterance?'. Depending on the application, the median rater's expected score may be desirable over the MOS because the overall variance will be reduced. The individual rater model can be used to post-screen outliers even after being fit on them by sampling raters from a truncated distribution that excludes the extreme values (e.g. raters within the 5 to 95 percentile).

The drawbacks of the individual score model must be considered. To compute a mean score with an individual score model, multiple samples must be taken and aggregated. This aggregation has computational cost, but the time it takes is relatively quick compared to the time it takes to fit a model.

Lastly, this experiment was concerned with speech data, but the findings may be relevant to non-speech audio as well. The causal DAG in figure 4.1 proposes that culture and language of the rater, which is independent of the content of the audio signal, affect scores. In this case the individual rater model has the potential to become more valuable because a given test signal may be more readily listened to by people from different languages and cultures, especially with online testing. An experiment to verify this would be prudent.

4.6 Summary

This chapter has shown that language-aware models provide a significant improvement over models that do not consider language, and that models of individual scores are able to match and exceed MOS models (in all aspects other than computational cost) while providing additional functionality. The experimental results validate the theory for these arguments.

The findings were over a single dataset, ITU-T P. Supplement 23, which was chosen because of the breadth of data it contains and the thorough process used to create it. However, the dataset is over 20 years old, and enough time has passed that subjective quality may have changed. Additionally, subjective tests are now conducted in a wider variety of environments, such as crowd-sourced tests at home conducted over the internet. It would be interesting future work to re-evaluate the same data in new tests in the same regions, to see how raters have changed since the creation of the P. Supp 23 data was created.

Real world subjective test data does not often have equally balanced experimental conditions over multiple languages as P. Supp. 23 does. However, one of the strengths of individual score Bayesian models is that they are able to handle unbalanced data (e.g. different numbers of listeners for each utterance). In this case, since the utterance quality might be different between languages, score comparisons between the languages may not be useful to look at, but the improvement in accuracy due to the language and rater metadata predictors can be measured. Based on the current results, further studies and applications for this type of data seem like a good next step.

Lastly, the findings about language and individual scores imply a causal model that should apply to non-Bayesian models. For example, deep learning models of

speech quality could consider taking into account individual ratings and language metadata in the feature set. Bayesian models are relatively difficult to fit as the data or parameter size grows to very large sizes (because typically the probabilities are computed over all the data), while deep learning can use mini-batches to handle virtually unlimited amounts of data. Given the recent advances in deep learning, it may be interesting to evaluate a model with many input features including language and individual rater score.

5 | Conclusion

This thesis has presented several aspects of speech quality and methods for improving its estimation through objective means. First we described modifications to ViSQOL, a public and free software tool that is used by many researchers and engineers dealing with speech applications that provides an alternative to paid license. Second, we showed how monotonic speech constraints can be preserved using a deep lattice model, which harnesses the power of deep learning while allowing for some control on what each layer does. Lastly, we presented an experiment on the effect that language and individual raters have on quality scores. These experiments and methods are a part of the process in creating useful models of quality.

ViSQOL is a continuously evolving quality estimator that integrates improvements based on trending research and production use case requirements. It uses traditional signal processing modules in combination with machine learning techniques. The deep lattice network described in chapter 3 is an example of this, which provided three important improvements. First, the quality of predictions was improved over the previous model, reducing the MSE from 0.57 to 0.24. Second, the DLN model obeys desirable monotonic constraints between similarity and quality, which resolves issues when handling out-of-sample data that would be problematic for support vector regression and naive DNN based models. Lastly, as a quantile-based estimator that estimates a distribution of MOS, the model has a more useful representation of uncertainty that can be conveyed to the user.

This thesis also investigated the role of language and the individual bias in subjective testing. The effects of language and culture on ratings is known to be non-negligible based on the literature. This thesis quantifies the marginal effect

of language on top of ViSQOL's 1-D signal level predictors, and still finds it to be significant. The individual rater's bias may also contribute some smaller but significant effect. In the experiments, Language and rater show marginal improvement over signal only predictors; RMSE was improved over the baseline NSIM model from 0.684 to 0.612 with the addition of language predictors and to 0.603 with rater predictors. These findings were obtained using Bayesian models to provide a more general analysis, as opposed to over-parameterized deep learning models. Since the findings were significant, the next step would be to incorporate language and rater metadata into existing frameworks like ViSQOL or deep learning models.

A growing trend is the use of probabilistic models of speech in synthesis or recognition. Such a model can also be used for quality, as has been done for the models in this thesis. The way speech is observed and rated is full of both uncertainty and bias. While deterministic point estimate models provide a useful simplification of the model that is acceptable for certain use cases, probabilistic models allow for a more realistic mapping of human perception and quality ratings. The probabilistic model for MOS also has the advantage of being able to estimate a distribution that can be used to express the uncertainty of the estimation.

Speech quality is ever-changing, due to the natural developments in technology and culture. For the last decade, machine learning has been the driving force of change in speech related technologies. Another trend to watch is the development of attention and transformers replacing RNNs [81]. Transformers may be well fit to the intrusive speech quality problem due to their ability to handle sequence alignment. As machine learning grows, having sufficiently diverse and useful data will be important. Traditional machine learning methods typically require quality ratings, which are expensive to obtain. As a result, there is a dearth of data for this problem, as quality ratings are not organically created. However,

novel techniques provide additional data through augmentation as well as semi-supervised and unsupervised methods [72]. The data problem may hint at the next frontier for quality modelling that leverages existing data and priors about perception and quality to harness the power of machine learning.

Better models of quality are a reflection of the capability to understand the nature of human perception, enabling others in turn to create better codecs, synthesis, and analysis applications. The popularity of the limited one-dimensional MOS is explained by the difficulty and ambiguity of the underlying problem of how to measure quality. As models continue to improve for the MOS case, and as the various quality factors are better understood and researched, it can be expected that more interpretable, multidimensional models of quality are likely to become more popular than MOS models. These powerful quality models can be used directly in the speech production models. In the ultimate case, if quality factors are fully qualified by a quality model, quality and synthesis could become completely integrated. In the meantime, it will be useful to focus on improving the model of quality to not only be more accurate, but to be more interpretable and representative of human perception.

Bibliography

[1] ITU-T Rec., "P. 800: Methods for subjective determination of transmission quality," *International Telecommunication Union, Geneva*, 1996.

[2] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ) - A new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, vol. 2. IEEE, 2001, pp. 749–752.

[3] ITU, "Perceptual evaluation of speech quality (PESQ): an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Int. Telecomm. Union, Geneva, Switzerland, ITU-T Rec. P.862, 2001.

[4] J. G. Beerends, C. Schmidmer, J. Berger, M. Obermann, R. Ullmann, J. Pomy, and M. Keyhl, "Perceptual objective listening quality assessment (POLQA), the third generation ITU-T standard for end-to-end speech quality measurement part I," *Journal of the Audio Engineering Society*, vol. 61, no. 6, pp. 366–384, 2013.

[5] ITU, "Perceptual objective listening quality assessment," Int. Telecomm. Union, Geneva, Switzerland, ITU-T Rec. P.863, 2018.

[6] T. Thiede, W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerends, and C. Colomes, "PEAQ - The ITU standard for objective measurement of perceived audio quality," *Journal of the Audio Engineering Society*, vol. 48, no. 1/2, pp. 3–29, 2000.

[7] R. Huber and B. Kollmeier, "PEMO-Q-A new method for objective audio quality assessment using a model of auditory perception," *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 6, pp. 1902–1911, 2006.

[8] A. Hines, J. Skoglund, A. Kokaram, and N. Harte, "ViSQOL: The virtual speech quality objective listener," in *IWAENC 2012; International Workshop on Acoustic Signal Enhancement*. VDE, 2012, pp. 1–4.

[9] A. Hines, E. Gillen, D. Kelly, J. Skoglund, A. Kokaram, and N. Harte, "ViSQOLAudio: An objective audio quality metric for low bitrate codecs," *The Journal of the Acoustical Society of America*, vol. 137, no. 6, pp. EL449–EL455, 2015.

[10] M. Narbutt, A. Allen, J. Skoglund, M. Chinen, and A. Hines, "AMBIQUAL - A full reference objective quality metric for ambisonic spatial audio," in *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2018, pp. 1–6.

[11] J.-M. Valin and J. Skoglund, "LPCNet: Improving neural speech synthesis through linear prediction," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2019, pp. 5891–5895.

[12] W. B. Kleijn, F. S. Lim, A. Luebs, J. Skoglund, F. Stimberg, Q. Wang, and T. C. Walters, "WaveNet based low rate speech coding," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 676–680.

[13] W. B. Kleijn, A. Storus, M. Chinen, T. Denton, F. S. Lim, A. Luebs, J. Skoglund, and H. Yeh, "Generative speech coding with predictive variance regularization," in *ICASSP 2021-2021 IEEE International Conference*

on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021, pp. 6478–6482.

[14] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *arXiv preprint arXiv:2107.03312*, 2021.

[15] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," *arXiv preprint arXiv:2104.00355*, 2021.

[16] S. Wisdom, J. R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R. A. Saurous, "Differentiable consistency constraints for improved deep speech enhancement," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 900–904.

[17] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE network*, vol. 12, no. 5, pp. 40–48, 1998.

[18] M. Chinen, F. S. C. Lim, J. Skoglund, N. Gureev, F. O'Gorman, and A. Hines, "ViSQOL v3: An open source production ready objective speech and audio metric," in *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*, 2020.

[19] A. Hines and N. Harte, "Speech intelligibility prediction using a neurogram similarity index measure," *Speech Communication*, vol. 54, no. 2, pp. 306 – 320, 2012.

[20] A. Hines, J. Skoglund, A. Kokaram, and N. Harte, "Robustness of speech quality metrics to background noise and network degradations: Comparing

ViSQOL, PESQ and POLQA," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 3697–3701.

[21] A. Hines, E. Gillen, and N. Harte, "Measuring and monitoring speech quality for Voice over IP with POLQA, ViSQOL and P. 563," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[22] C. Sloan, N. Harte, D. Kelly, A. C. Kokaram, and A. Hines, "Bitrate classification of twice-encoded audio using objective quality features," in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2016, pp. 1–6.

[23] ——, "Objective assessment of perceptual audio quality using ViSQOLAudio," *IEEE Transactions on Broadcasting*, vol. 63, no. 4, pp. 693–705, 2017.

[24] A. Hines and N. Harte, "Speech intelligibility prediction using a neurogram similarity index measure," *Speech Communication*, vol. 54, no. 2, pp. 306–320, 2012.

[25] Google. (2017) Abseil. [Online]. Available: https://abseil.io/

[26] ——. (2008) Google test framework. [Online]. Available: https://github.com/google/googletest

[27] ——. (2015) Bazel build system. [Online]. Available: https://bazel.build/

[28] T. Mo and A. Hines, "Jitter buffer compensation in Voice over IP quality estimation," in *2019 30th Irish Signals and Systems Conference (ISSC)*. IEEE, 2019, pp. 1–6.

[29] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, pp. 1–27, 2011.

[30] (2011) WebRTC. [Online]. Available: webrtc.org

[31] A. Hines, E. Gillen, D. Kelly, J. Skoglund, A. Kokaram, and N. Harte, "Perceived audio quality for streaming stereo music," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 1173–1176.

[32] CoreSV Team. (2014) CoreSV listening test. [Online]. Available: http://listening-test.coresv.net/results.htm

[33] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "LibriTTS: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.

[34] M. Chinen, J. Skoglund, and A. Hines, "Speech quality estimation with deep lattice networks," *The Journal of the Acoustical Society of America*, vol. 149, no. 6, pp. 3851–3861, 2021.

[35] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[36] W. A. Jassim, J. Skoglund, M. Chinen, and A. Hines, "WARP-Q: Quality prediction for generative neural speech codecs," in *ICASSP 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021.

[37] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.

[38] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. Courville, "MelGAN: Generative

adversarial networks for conditional waveform synthesis," *arXiv preprint arXiv:1910.06711*, 2019.

[39] B.-K. Lee and J.-H. Chang, "Packet loss concealment based on deep neural networks for digital speech transmission," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 2, pp. 378–387, 2015.

[40] M.-K. Lee and H.-G. Kang, "Speech quality estimation of voice over internet protocol codec using a packet loss impairment model," *The Journal of the Acoustical Society of America*, vol. 134, no. 5, pp. EL438–EL444, 2013.

[41] Z. Cai, N. Kitawaki, T. Yamada, and S. Makino, "Comparison of MOS evaluation characteristics for Chinese, Japanese, and English in IP telephony," in *2010 4th International Universal Communication Symposium*. IEEE, 2010, pp. 112–115.

[42] W. Labov, S. Ash, and C. Boberg, *The atlas of North American English: Phonetics, phonology and sound change*. Walter de Gruyter, 2008.

[43] H. Z. Jahromi, D. T. Delaney, and A. Hines, "Beyond first impressions: Estimating quality of experience for interactive web applications," *IEEE Access*, vol. 8, pp. 47 741–47 755, 2020.

[44] S. Zielinski, F. Rumsey, and S. Bech, "On some biases encountered in modern audio quality listening tests-a review," *Journal of the Audio Engineering Society*, vol. 56, no. 6, pp. 427–451, 2008.

[45] I. Mossavat, P. N. Petkov, W. B. Kleijn, and O. Amft, "A hierarchical bayesian approach to modeling heterogeneity in speech quality assessment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 136–146, 2012.

[46] J. Nawala, L. Janowski, B. Cmiel, and K. Rusek, "Describing subjective experiment consistency by p-value p–p plot," *Proceedings of the 28th ACM International Conference on Multimedia*, Oct 2020. [Online]. Available: http://dx.doi.org/10.1145/3394171.3413749

[47] M. Seufert, "Fundamental advantages of considering quality of experience distributions over mean opinion scores," in *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*, 2019, pp. 1–6.

[48] T. Hoßfeld, P. E. Heegaard, M. Varela, and S. Möller, "QoE beyond the MOS: an in-depth look at QoE via better metrics and their relation to MOS," *Quality and User Experience*, vol. 1, no. 1, pp. 1–23, 2016.

[49] M. J. Bianco, P. Gerstoft, J. Traer, E. Ozanich, M. A. Roch, S. Gannot, and C.-A. Deledalle, "Machine learning in acoustics: Theory and applications," *The Journal of the Acoustical Society of America*, vol. 146, no. 5, pp. 3590–3628, 2019.

[50] H. Gamper, C. K. Reddy, R. Cutler, I. J. Tashev, and J. Gehrke, "Intrusive and non-intrusive perceptual speech quality assessment using a convolutional neural network," in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2019, pp. 85–89.

[51] G. Chen and V. Parsa, "Loudness pattern-based speech quality evaluation using Bayesian modeling and Markov chain Monte Carlo methods," *The Journal of the Acoustical Society of America*, vol. 121, no. 2, pp. EL77–EL83, 2007.

[52] S. You, D. Ding, K. Canini, J. Pfeifer, and M. Gupta, "Deep lattice networks and partial monotonic functions," in *Advances in neural information processing systems*, 2017, pp. 2981–2989.

[53] N. Osaka, K. Kakehi, S. Iai, and N. Kitawaki, "A model for evaluating talker echo and sidetone in a telephone transmission network," *IEEE transactions on communications*, vol. 40, no. 11, pp. 1684–1692, 1992.

[54] Supplement 3 to ITU-T Series P Recommendations , *Models for predicting transmission quality from objective measurements*. Geneva: International Telecommunication Union, 1993.

[55] J. A. Bergstra and C. Middelburg, "ITU-T Recommendation G. 107: The E-Model, a computational model for use in transmission planning," *Technical report, ITU*, 2003.

[56] M. Slaney *et al.*, "An efficient implementation of the Patterson-Holdsworth auditory filter bank," *Apple Computer, Perception Group, Tech. Rep*, vol. 35, no. 8, 1993.

[57] ITU-T Rec., "P. 862.1: Mapping function for transforming P. 862 raw result scores to MOS-LQO," *International Telecommunication Union, Geneva*, vol. 24, 2003.

[58] C.-C. Chang and C.-J. Lin, "Training v-support vector regression: theory and algorithms," *Neural computation*, vol. 14, no. 8, pp. 1959–1977, 2002.

[59] M. Gupta, A. Cotter, J. Pfeifer, K. Voevodski, K. Canini, A. Mangylov, W. Moczydlowski, and A. van Esbroeck, "Monotonic calibrated interpolated look-up tables," *Journal of Machine Learning Research*, vol. 17, no. 109, pp. 1–47, 2016. [Online]. Available: http://jmlr.org/papers/v17/15-243.html

[60] M. M. Fard, O. Mangylov, W. Bakst, T. Narayan, N. Morioka, Y. Zhou, E. Louidor, and S. Wang, "Tensorflow lattice," 2020. [Online]. Available: https://github.com/tensorflow/lattice

[61] ITU-T Recommendation P.1401, *Methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models*. Geneva: International Telecommunication Union, 2020.

[62] R. Koenker and G. Bassett Jr, "Regression quantiles," *Econometrica: journal of the Econometric Society*, pp. 33–50, 1978.

[63] T. Narayan, S. Wang, K. Canini, and M. Gupta, "Regularization strategies for quantile regression," *arXiv preprint arXiv:2102.05135*, 2021.

[64] N. Harte, E. Gillen, and A. Hines, "TCD-VoIP, a research database of degraded speech for assessing quality in VoIP applications," in *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*. IEEE, 2015, pp. 1–6.

[65] ITU-T, *ITU-T Supplement P.23 coded-speech database*. Geneva: International Telecommunication Union, 1998.

[66] ITU-T Rec., "P.862.3 : Application guide for objective quality measurement based on recommendations p.862, p.862.1 and p.862.2," *International Telecommunication Union, Geneva*, 2007.

[67] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *2010 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2010, pp. 4214–4217.

[68] S. Wang and M. Gupta, "Deontological ethics by monotonicity shape constraints," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2043–2054.

[69] M. Chinen, "Marginal effects of language and individual raters on speech quality models," *IEEE Access*, vol. 9, pp. 127 320–127 334, 2021.

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

[71] L. Ding and R. A. Goubran, "Speech quality prediction in voip using the extended e-model," in *GLOBECOM'03. IEEE Global Telecommunications Conference (IEEE Cat. No. 03CH37489)*, vol. 7. IEEE, 2003, pp. 3974–3978.

[72] J. Serrà, J. Pons, and S. Pascual, "SESQA: semi-supervised learning for speech quality assessment," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 381–385.

[73] A. Ragano, E. Benetos, and A. Hines, "More for less: Non-intrusive speech quality assessment with limited annotations," in *2021 13th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2021, pp. 103–108.

[74] R. Z. Jiménez, L. F. Gallardo, and S. Möller, "Influence of number of stimuli for subjective speech quality assessment in crowdsourcing," in *2018 Tenth international conference on quality of multimedia experience (QoMEX)*. IEEE, 2018, pp. 1–6.

[75] M. Nakayama and Y. Wan, "Same sushi, different impressions: a cross-cultural analysis of yelp reviews," *Information Technology & Tourism*, vol. 21, no. 2, pp. 181–207, 2019.

[76] A. Takahashi, H. Yoshino, and N. Kitawaki, "Perceptual QoS assessment technologies for VoIP," *IEEE Communications Magazine*, vol. 42, no. 7, pp.

28–34, 2004.

[77] ——, “Quality assessment methodologies for IP-telephony services,” *IEICE Trans. Commun.(Japanese Edition)*, vol. 88, pp. 863–874, 2005.

[78] ITU-T Rec., “P.1401 : Methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models,” *International Telecommunication Union, Geneva*, 2020.

[79] M. D. Hoffman, A. Gelman *et al.*, “The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1593–1623, 2014.

[80] J. V. Dillon, I. Langmore, D. Tran, E. Brevdo, S. Vasudevan, D. Moore, B. Patton, A. Alemi, M. Hoffman, and R. A. Saurous, “Tensorflow distributions,” *Probabilistic Programming Languages, Semantics, and Systems (PPS 2018)*, 2018.

[81] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in neural information processing systems*, 2017, pp. 5998–6008.