An Intrusive Neural Approach for Speech Intelligibility Prediction using Whisper Embeddings and Attention Pooling

Robson de Souza Pedroso

Independent Researcher

rdesouzapedroso@gmail.com

Abstract

This paper describes my system for the 3rd Clarity Prediction Challenge (CPC3), focused on predicting speech intelligibility for hearing aid users. My approach is based on an intrusive neural regressor that leverages the power of pre-trained openai/whisper encoders. For each audio sample, embeddings are extracted from both the processed signal and the clean reference waveform using a frozen Whisper encoder. These embeddings are then aggregated using a custom Attention Pooling layer before being combined with acoustic metadata. The final prediction is generated by a regression head. To combat overfitting, I employed data augmentation and carefully tuned regularization parameters such as dropout and weight decay. My final submitted system, an arithmetic mean ensemble of two separately trained models based on the whisper-base and whisper-medium backbones, achieved a final Root Mean Squared Error (RMSE) of 27.82 on the official evaluation set.

1. Introduction

The reliable automatic evaluation of speech intelligibility is a critical component in the development of modern hearing enhancement technologies. For hearing aid users, the ability to understand speech in noisy environments is a primary measure of a device's effectiveness. The 3rd Clarity Prediction Challenge (CPC3) addresses this problem by tasking participants with creating a predictive model that estimates speech intelligibility scores, defined as the percentage of correctly recognized words by a listener.

This task requires a system to process an audio signal that has passed through a hearing aid and, given listener characteristics, predict the resulting intelligibility score. This paper details the system I developed for this challenge. My approach is based on an intrusive neural architecture that leverages the powerful feature extraction capabilities of large pre-trained speech models. Specifically, I use the openai/whisper encoder to generate rich embeddings from the audio waveforms. These embeddings, combined with engineered acoustic features, are then processed by a custom regression head incorporating an attention mechanism to produce the final score. In the following sections, I will describe the system architecture, the experimental setup, and the final results achieved on the evaluation set.

2. System Description

My system is an intrusive intelligibility predictor, meaning it utilizes both the processed signal and its corresponding clean reference. The overall pipeline can be broken down into three main stages: acoustic feature engineering, neural feature extraction using a Whisper-based model, and a final regression head.

2.1. Acoustic Feature Engineering

Before feeding the data into the neural model, I first performed a feature engineering step to extract a set of descriptive acoustic metadata for each audio sample. As this process can be computationally intensive, it was performed once for the entire dataset and the results were saved in Parquet files for efficient loading during training.

This process involved loading the audio files with librosa and calculating a variety of features, including:

- **Signal-to-Noise Ratio (SNR):** A classic measure of signal quality.
- **Spectral Features:** Mean spectral contrast, centroid, and bandwidth, which describe the frequency content of the signal.
- **Perceptual Features:** The mean and standard deviation of 13 Mel-Frequency Cepstral Coefficients (MFCCs).
- **Energy Features:** The Root Mean Square (RMS) energy of both the processed signal and the estimated noise.

These engineered features were later concatenated with the neural embeddings to provide the model with explicit acoustic information.

2.2. Model Architecture

The core of my system is a neural regressor built on top of a pre-trained openai/whisper encoder. The complete architecture is shown bellow.

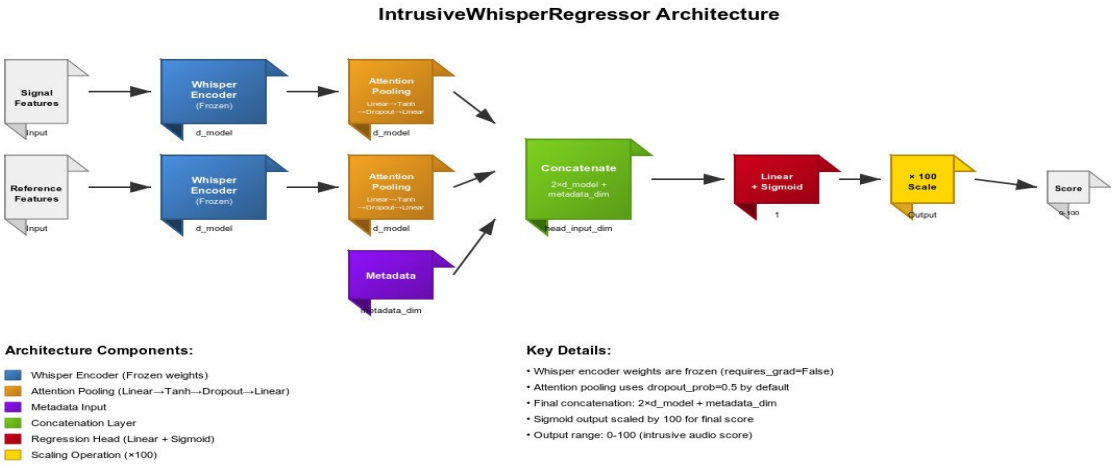

Figure [1].

The model processes the input waveforms as follows:

1. **Whisper Encoder:** Both the processed signal and the clean reference signal are independently passed through the frozen encoder of a pre-trained Whisper model. This step acts as a feature extractor, converting the raw audio into high-dimensional embedding sequences that capture rich semantic and acoustic information.
2. **Attention Pooling:** Instead of a simple mean-pooling of the embedding sequence, which was tested in early experiments, I implemented an AttentionPooling layer for each branch (signal and reference). This layer calculates a weighted average of the embedding sequence, allowing the model to dynamically focus on the most relevant temporal frames for intelligibility prediction. A dropout rate of 0.3 was used for the whisper-base model, and this was increased to 0.5 for the larger whisper-medium model to provide stronger regularization.
3. **Concatenation:** The two aggregated embeddings (one from the signal, one from the reference) are concatenated with the pre-calculated acoustic metadata vector described in Section 2.1.
4. **Regression Head:** This combined vector is fed into a final linear layer followed by a Sigmoid activation function. The output is scaled by 100 to produce the final intelligibility score prediction between 0 and 100.

2.3. Training and Regularization

I trained two primary versions of this model, one using the whisper-base encoder and a larger one using whisper-medium. The models were trained using the NAdam optimizer. The learning rate was set to 1.0e-4 for the base model, and a more aggressive weight_decay of 1.0e-4 was used for the medium model to counteract its increased capacity.

To improve generalization and make the model more robust, two key regularization strategies were employed:

- **Data Augmentation:** During training, I applied a pipeline of augmentations from the torch_audiomentations library exclusively to the processed signal audio. This included techniques such as adding colored noise, applying gain, polarity inversion, and frequency-based filtering. This step proved to be surprisingly effective at reducing overfitting.
- **Early Stopping:** Training was monitored on a validation set, and early stopping with a patience of 3-5 epochs was used to save the model with the best validation RMSE and prevent further overfitting.

3. Experiments and Results

3.1. Experimental Setup: The system was implemented using PyTorch. All models were trained on a Kaggle environment equipped with NVIDIA GPUs. The dataset provided by the CPC3 organizers was split into training and validation sets using a GroupShuffleSplit strategy. To ensure a realistic evaluation of generalization, the data was grouped by scene_id, which guarantees that all audio samples from a given recording scene belong exclusively to either the training or the validation set.

The initial training experiments were conducted with a model using the whisper-base encoder. Based on its stable performance, a larger model using the whisper-medium encoder was subsequently trained with stronger regularization parameters to manage its increased capacity.

3.2. Results and Analysis: The training history for both the base and medium models is presented bellow**.**

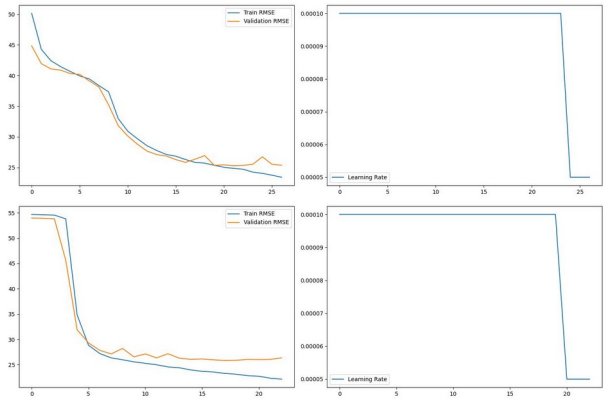

Figure 2: Training history for the whisper-base model (top) and the whisper-medium model (bottom). The plots show the Root Mean Squared Error (RMSE) on the training and validation sets, and the learning rate schedule over 20-25 epochs.

As shown in Figure 2, both models exhibit a stable training progression. The validation loss closely tracks the training loss, indicating that the regularization strategies, particularly data augmentation and early stopping, were effective in preventing severe overfitting. The whisper-base model shows a slightly smaller gap between training and validation RMSE, while the larger whisper-medium model converges to a similar validation performance, demonstrating that its increased complexity did not yield significant gains with the current training strategy. The learning rate was automatically reduced by a ReduceLROnPlateau scheduler when the validation RMSE plateaued.

I hypothesize that the performance of the whisper-medium model could be further improved by unfreezing the encoder's final layers and continuing training with a lower learning rate (e.g., 1e-5). However, due to the competition's time constraints and the significant increase in computational cost, this hypothesis was not tested.

My final submission to the challenge was an ensemble created by taking the arithmetic mean of the predictions from both the trained whisper-base and whisper-medium models. This approach leverages the slightly different patterns learned by each model to produce a more robust final prediction.

On the official evaluation set, my submitted system (ID E038) achieved a final Root Mean Squared Error (RMSE) of 27.82 and a correlation of 0.741. This result is consistent with the expected performance drop from the development set, as noted by the challenge organizers, due to the increased diversity of listeners and systems in the final evaluation data.

4. Conclusions

In this paper, I presented an intrusive neural system for speech intelligibility prediction. The approach successfully combined the powerful feature extraction capabilities of large, pre-trained Whisper encoders with a custom, attention-based regression head. Key to the system's stable performance was the implementation of strong regularization techniques, including aggressive data augmentation and carefully tuned dropout and weight decay, which effectively controlled overfitting.

The final ensemble of models based on whisper-base and whisper-medium backbones demonstrated a robust performance on the final evaluation set. Future work could explore the impact of fine-tuning the full encoder on a larger corpus of intelligibility data before adapting it to the final task, which may unlock further performance gains from the larger model architectures.

5. Acknowledgements

I would like to thank the organizers of the Clarity Prediction Challenge for creating this valuable dataset and for organizing the workshop.

6. References

[1] *Whisper: Radford, A., et al. (2023). "Robust Speech Recognition via Large-Scale Weak Supervision."*

[2] PyTorch: Paszke, A., et al. (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library."

[3] Librosa: McFee, B., et al. (2015). "librosa: Audio and music signal analysis in python."

[4] Er-Rady, A. (2020). "TorchAudioMentations: A library for audio data augmentation."