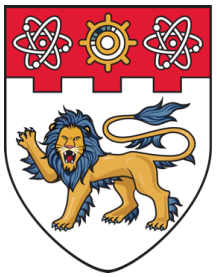


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

Optimizing Speech Representation Learning for Enhanced Noise Robustness in Downstream Applications

Dianwen Ng

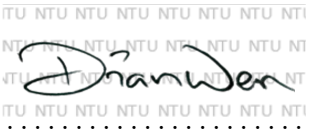
College of Computing and Data Science

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2025

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.



10/01/2025

Date

Dianwen Ng

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

10/01/2025

.....

Date

.....

Prof. Chng Eng Siong

Authorship Attribution Statement

This thesis includes material from four accepted papers at conferences.

Chapter 3 is published as Ng, Dianwen, Ruixi Zhang, Jia Qi Yip, Zhao Yang, Jinjie Ni, Chong Zhang, Yukun Ma, Chongjia Ni, Eng Siong Chng, and Bin Ma. "deHuBERT: Disentangling noise in a self-supervised model for robust speech recognition." In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5. IEEE, 2023. The contributions of the co-authors are as follows:

- I designed the methodology, planned the experiments, and prepared the manuscript drafts.
- Ruixi Zhang provided rigorous discussion and guidance in interpreting related works.
- Prof. Eng Siong Chng and Dr. Bin Ma provided supervision for the experiments and the writing of the manuscript.
- The rest of the authors proofread the manuscript.

Chapter 4 includes analysis and discussion from the published paper Ng, Dianwen, Chong Zhang, Ruixi Zhang, Yukun Ma, Fabian Ritter-Gutierrez, Trung Hieu Nguyen, Chongjia Ni, Shengkui Zhao, Eng Siong Chng, and Bin Ma. "Are Soft Prompts Good Zero-Shot Learners for Speech Recognition?" In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10366-10370. IEEE, 2024. The contributions of the co-authors are as follows:

- I designed the methodology, planned the experiments, conducted the evaluations and prepared the manuscript drafts.
- Ruixi Zhang and Fabian Ritter-Gutierrez contributed to the discussion of the results.
- Prof. Eng Siong Chng and Dr. Bin Ma provided supervision for the experiments and the writing of the manuscript.
- The rest of the authors proofread the manuscript.

Chapter 4 includes the methodology from the published paper Ng, Dianwen, Kun Zhou, Bin Ma, and Eng Siong Chng. "Thinking Fast and Slow: Robust Speech Recognition via Deep Filter-Tuning" In *Proceedings of INTERSPEECH*. 2025. The contributions of the co-authors are as follows:

- I designed the methodology, planned the experiments, conducted the evaluations and prepared the manuscript drafts.
- Prof. Eng Siong Chng and Dr. Bin Ma provided supervision for the experiments and the writing of the manuscript.
- The rest of the authors proofread the manuscript.

Chapter 5 is published as Ng, Dianwen, Kun Zhou, Yi-wen Chao, Zhiwei Xiong, Bin Ma, and Eng Siong Chng. "Multi-band Frequency Reconstruction for Neural Psychoacoustic Coding" In *Forty-Second International Conference on Machine Learning (ICML)*, 2025. The contributions of the co-authors are as follows:

- I designed the methodology, planned the experiments, conducted the evaluations and prepared the manuscript drafts.
- Kun Zhou provided feedback and assisted with some of the experiments related to the text-to-speech (TTS) section.
- Prof. Eng Siong Chng and Dr. Bin Ma provided supervision for the experiments and the writing of the manuscript.
- The other authors contributed to the survey of related works and proofread the manuscript.

10/01/2025

Date

Dianwen Ng

Acknowledgements

I wish to express my profound appreciation to my advisors, Professor Eng Siong Chng and Dr. Bin Ma, for introducing me to the captivating field of speech processing and for their steadfast support and insightful guidance throughout my research journey. Their expertise and mentorship have been the bedrock of my understanding and passion for this dynamic area of study.

Special thanks go to the Singapore Economic Development Board (EDB) and Alibaba Group for their pivotal role in funding my exploration of the marvels of artificial intelligence, enabling me to satisfy my deep-seated curiosity and expand my academic horizons. Their generous support has provided me with invaluable opportunities to present my work on international stages and to forge connections with leading professionals and like-minded peers worldwide.

I am also deeply grateful to my friends and colleagues, the pillars of my daily resolve, whose shared experiences and uplifting spirits have buoyed me through our collective academic endeavors. To my family, whose unwavering belief in my potential has been the wind beneath my wings, thank you for propelling me forward when the journey grew arduous.

To my future furry friend, I'm eagerly ticking off the days until our paths cross and I can finally give you the grand welcome you deserve, complete with endless treats and the prime spot on the couch. Together, we'll navigate the joyful escapades of life, from mastering the art of the belly rub to sharing quiet moments of companionship. We'll laugh, we'll play, and we'll learn about love and life in ways only a true friend can teach.

Finally, I dedicate this milestone to my wife, Jiang Yiqiao—my soulmate, best friend, comrade, and partner in all adventures. Here's to the dreams we will chase together, the challenges we will overcome, and the lifelong bond we will build, fortified by mutual support and shared growth.

"Our greatest glory is not in never failing, but in rising up every time we fail."

— Ralph Waldo Emerson

Abstract

The primary objective of this thesis is to enhance the effectiveness and efficiency of speech representations, specifically improving noise robustness for downstream applications. Current speech representation learning frameworks, despite their advanced foundational knowledge and powerful speech understanding capabilities, fall short in critical areas essential for real-world applications, such as adaptability to noise, expressiveness, and computational efficiency. For instance, their performance varies greatly across different levels of noise corruption, showing high vulnerability to distortion from external influences. Moreover, learning for domain adaptation and performing inference are computationally intensive due to the large scale and complex design of the model structures. Hence, this thesis introduces innovative solutions to bridge these gaps, offering substantial improvements over existing methods.

Adaptability to Noise: We address noise robustness by integrating Barlow Twins learning, an advanced regularization technique that strategically reduces channel redundancy and effectively disentangles speech features, as presented in chapter 3. This ensures that our models capture essential, noise-free features critical for accurate speech recognition, even in adverse acoustic environments, thereby ensuring more stable and robust performance to noise distortion.

Expressiveness: To enhance the expressivity of pre-trained models, we employ a parameter-efficient fine-tuning approach that incorporates the proposed deep filter tuning with Feature-wise Linear Modulation (FiLM)-inspired integration, as detailed in Chapter 4. This method refines the handling of speech nuances, allowing the models to more effectively express important information through targeted feature extraction from FiLM. This adaptation better accommodates diverse vocal attributes and acoustic variations.

Efficiency: Our proposed deep filter tuning strategy enables efficient adaptation of frozen pre-trained models with the aim of minimizing the number of trainable parameters. This approach optimizes computational resources and maximizes efficiency, particularly when faced with constraints in computational memory. In addition to the work, we also innovate in data compression through multi-band quantization, optimizing bit allocation by prioritizing perceptually significant features in chapter 5. This approach leverages psychoacoustic principles to enhance the efficiency of speech processing and is particularly beneficial for speech synthesis and voice conversion tasks, ensuring high fidelity in the outputs.

Our methods have been rigorously tested across tasks such as automatic speech recognition and speech reconstruction, demonstrating significant enhancements in accuracy, robustness, and processing efficiency. These improvements underscore the potential of our approaches to revolutionize speech representation learning, making it more adaptive, scalable, and context-aware.

By meticulously addressing the inherent limitations of current models, this research advances the field of speech representation learning, setting a new benchmark for future developments and applications in adaptive and efficient speech technology.

Contents

List of Figures

List of Tables

Acronyms

CER	Character Error Rate
WER	Word Error Rates
PESQ	Perceptual Evaluation of Speech Quality
STOI	Short-Time Objective Intelligibility
MOS	Mean Opinion Score
ViSQOL	Virtual Speech Quality Objective Listener
F0CORR	F0 Pearson Correlation Coefficient
MAC	Multiply-Accumulate Operations
SNRs	Signal-to-noise Ratios
GMMs	Gaussian Mixture Models
RNNs	Recurrent Neural Networks
RNN-T	Recurrent Neural Network Transducers
BiLSTM	Bidirectional Long Short-term Memory
FiLM	Feature-wise Linear Modulation
MRF	Multi-receptive Field
RELU	Rectified Linear Unit
STFT	Short-time Fourier Transform
FFT	Fast Fourier Transform
MFCC	Mel-Frequency Cepstral Coefficients
DCT	Discrete Cosine Transform
LM	Language Model
LLMs	Large Language Models
w2v2	Wav2vec 2.0
MAE	Masked Autoencoder
DFT	Deep Filter-tuning
GPT	Generative Pre-trained Transformer
NAC	Neural Audio Codec
EMA	Exponential Moving Average
MSE	Mean Squared Error
MPC	Masked Predictive Coding
CPC	Contrastive Predictive Coding
APC	Auto-regressive Predictive Coding
CC	Cross-correlation

SC	Self-correlation
MPD	Multi-period Discriminator
MSD	Multi-scale Discriminator
OOD	Out-of-domain
NLP	Natural Language Processing
ASR	Automatic Speech Recognition
PEFT	Parameter Efficient Fine-tuning
LoRA	Low-Rank Adaptation
RVQ	Residual Vector Quantization
PQ	Product Quantization
FVQ	Factor Vector Quantization

LS	LibriSpeech
TED	Ted-Lium 3
WSJ10	Wall Street Journal
SWBD	Switchboard
ESD	Emotional Speech Dataset

Chapter 1

Introduction

1.1 Background and Motivation

Recent advancements in artificial general intelligence (AGI), powered by large foundational models, are reshaping generative and interactive AI. These systems are coming closer to achieving human-like communication and understanding [5, 6]. Achieving this level of performance requires the development of high-quality and efficient speech representations that serve as inputs to the foundational models. When these representations are inadequate, they can bottleneck the performance of AGI systems, limiting their potential and diminishing the user experience. This challenge underscores the importance for robust frameworks in efficient speech representation learning.

Speech representation learning is a specialized subfield of machine learning focused on transforming raw speech signals into high-level features that reflect both linguistic and acoustic information [7–9]. Unlike text or images, speech is continuous and highly variable, containing layers of information beyond just phonetic content, such as prosody (intonation, rhythm, and stress), speaker identity, and environmental noise [10–13]. These interconnected elements can differ significantly depending on the context and speaker [14, 15], making speech processing a complex task. Poorly designed representations often fail to generalize well to these nuances, resulting in

poor adaptation and noise-sensitive performance in downstream tasks. Sophisticated methods are therefore being researched and proposed to improve the speech representation learning paradigm.

1.2 Research Problems

Despite ongoing efforts to improve speech representation learning frameworks, substantial challenges remain in developing models that are characteristically adaptable, expressive and efficient for real-world applications.

Adaptability to noise is a critical requirement for practical speech recognition systems [16–18], particularly in real-world applications where speech understanding often occurs in open and uncontrolled environments. In such settings, noise and reverberation are unavoidable, frequently causing signal distortions that degrade the performance of speech recognition models. These distortions interfere with the system's ability to accurately interpret and transcribe speech. To mitigate these challenges, models must be designed to either suppress the adverse effects of background noise or intelligently exploit rich noise contexts to improve information processing [19–21]. Such capabilities ensure robust and accurate speech recognition, even in complex acoustic environments.

Expressiveness is equally critical, as models must capture the multidimensional nature of speech, which extends beyond linguistic content [22–27]. An effective representation encodes not only lexical meaning but also essential paralinguistic features such as speaker identity, emotional nuances, prosodic variations (including intonation, rhythm, and stress), and contextual subtleties [28–30]. By integrating these complex dimensions, machine learning models can achieve a more sophisticated understanding of the semantic and structural aspects of speech. This enables them to generalize latent features across a diverse range of users with varying voices, accents, and speaking styles, thereby facilitating more accurate and context-aware speech-to-text transcription [31].

Efficiency in handling model structure and data is another crucial consideration. Current learnt speech representations often suffer from channel redundancy and

extraneous information, such as noise distortion, which reduces efficiency and hampers their ability to compactly encode meaningful features [32–35]. Studies have consistently shown that channel redundancy is a pervasive issue in deep learning models, particularly as these models scale in size and complexity. Despite dramatic increases in model dimensionality, a significant portion of channels fail to contribute unique or distinctly meaningful representations [36]. Empirical investigations reveal that large neural networks often contain substantial redundant channels, with some research suggesting that 50-70% of channels could be pruned without significant performance degradation [37, 38]. This redundancy arises from complex interactions between model architecture, optimization dynamics, and feature learning mechanisms. As models grow in size, the additional channels do not necessarily lead to proportional improvements in representational diversity. Instead, these channels frequently capture highly correlated or minimally distinct feature maps, resulting in computational inefficiency. At times, this inefficiency in model scaling can result in ineffective handling of sophisticated and diverse feature representations, exacerbating the entanglement of multifaceted speaker attributes. This entanglement increases the system's susceptibility to cascading effects from noise disturbances, even when such disturbances are subtle. These challenges highlight the need for model design strategies that prioritize feature distinctiveness and computational efficiency.

1.2.1 Research Goals

Building on these challenges, our work aims to explore innovative approaches to advancing speech representation learning, with a focus on improving adaptability to background noise and enhancing encoding efficiency. Specifically, we investigate the impact of noise distortion on the generalization of feature representations, identifying system limitations and proposing novel strategies to redesign model architectures. These frameworks introduce the concept of neural filtering during representation encoding, aimed at enhancing noise robustness by selectively emphasizing critical features while suppressing irrelevant distortions.

Additionally, we apply principles of psychoacoustics from speech perception to computational structures. By leveraging these principles [39–41], we develop methods

for more efficient bit allocation, ensuring that perceptually significant features are prioritized over less relevant information. This results in encoded representations that are both precise and expressive, leading to improved context-aware outputs.

In downstream applications, such as speech recognition systems, these enhanced representations enable more accurate and robust transcription, leading to lower Word Error Rates (WER) even in challenging acoustic environments. For generative tasks such as speech synthesis or voice conversion, the ability to preserve nuanced prosodic and contextual features while minimizing noise artifacts ensures higher fidelity and more natural outputs. This is reflected in improved perceptual metrics such as Perceptual Evaluation of Speech Quality (PESQ) [42], Short-Time Objective Intelligibility (STOI) [43], and Mean Opinion Score (MOS) [44]. Together, these advancements lay the groundwork for robust, efficient, and adaptable speech representation models, optimized to enhance the foundational model's understanding of speech input. This enables more accurate recognition that assist more contextually relevant responses, and shorter response times for a smoother user experience.

1.2.2 Objectives Outline

Specifically, we outline the objectives and contributions that address the goal of this thesis in three key dimensions:

Objective 1: Enhance noise robustness and channel redundancy in self-supervised speech representation model.
We begin by investigating self-supervised speech representation models, which offer a solid foundation for acquiring meaningful speech features without the need for extensive labeled data. These models have shown impressive performance in a variety of speech-related tasks, including automatic speech recognition, speaker identification, and emotion recognition [45–49]. However, their susceptibility to noise and inefficient channel utilization present considerable obstacles in practical applications [33]. To address these challenges, we utilize advanced regularization techniques and integrate Barlow Twins learning [50] into our optimization framework. Barlow Twins, inspired by the information bottleneck principle, encourages the model to produce embeddings that are both invariant to input distortions

and statistically independent across feature dimensions. This approach aims to minimize the influence of noise and reduce redundancy in the neural embedding representations.

Our method involves using two identical neural networks, commonly referred to as "twins", that process two altered versions of the same input sample. Our goal is to make the outputs of these networks as similar as possible for identical inputs, thereby ensuring that the networks can robustly encode the same features despite noise distortions to reduce the impact of background noises. Additionally, the Barlow Twins methodology strives to diminish redundancy in the embeddings by analyzing the cross-correlation matrix of the outputs from the two networks. We aim for this matrix to approach an identity matrix, which would indicate that each channel of information is as independent as possible from the others. This, in turn, enhances the expressiveness of latent embeddings by promoting disentangled representations, which in turn facilitates more efficient downstream processing in tasks such as speech recognition under heavy background noise, ultimately achieving lower word error rates across more generalized noisy environments–for example, transcribing speech in a noisy cafe during teleconferences, which represents our North Star.

Objective 2: Adapting representations of pre-trained foundational model with parameter efficient fine-tuning to enhance its noise robustness.
Our research extends to large-scale self-supervised speech representation models. According to scaling laws [51], larger models generally show improved performance as the quantity of self-labeled samples increases. However, adapting these models requires extensive computational resources, particularly when fine-tuning for noise-robust automatic speech recognition. This creates a significant bottleneck, especially when customizing these models to handle specific domain background noises in environments with limited resources [52, 53].

To mitigate this, we propose a parameter-efficient fine-tuning strategy using deep filter tuning to refine latent speech representations, described in Chapter 4 [53]. In this method, the pre-trained foundational model remains frozen, which avoids the necessity to update model parameters, thus conserving computational power. Adaptation of speech representations is facilitated through the use of externally introduced soft tokens, known as prompts for prompt tuning.

Although prompt tuning offers some benefits, we observe that it typically lacks sufficient expressive capability to interact effectively with the frozen representations at the prompt attention stage [54]. This deficiency limits the tuning's effectiveness in enhancing noise robustness. To overcome this, we employ a method adapted from speaker extraction techniques, utilizing feature-wise linear modulation (FiLM) [55, 56] with the soft tokens to directly modulate the frozen representations. This approach significantly enhances model expressivity and mitigates noise-induced distortion, effectively overcoming the limitations of prompt tuning. This leads to improved recognition performance and consistently lower word error rates across diverse and noisy environments, while also enabling cost-effective noise adaptation, particularly for communities with limited computational resources to fine-tune large pre-trained models for better noise distorted automatic speech recognition.

Objective 3: Enhances bit allocation efficiency in compression techniques to achieve higher rates without compromising speech representation expressivity.

Typically, modern speech representations encode audio signals at 16kHz and 24kHz to capture high-quality audio and superior speech representations. Conventionally, subsampling within a window of frames reduces the rate to 50Hz or 75Hz, enhancing downstream task optimization efficiency and achieving a compression factor of up to 320 times [57, 58]. These latent embeddings are then quantized into tokens by a neural audio codec, which serve as inputs to a foundational language model, effectively bridging speech modalities and language processing [59]. Traditional quantization processes, particularly residual vector quantization, often prioritize filling the first codebook, which can lead to significant information loss as compression rates increase.

To overcome this limitation, we introduce multi-band quantization at the latent embedding level, illustrated in Chapter 5, partitioning encoded information across distinct frequency bands, assigning each to separate codebooks. This method enhances bit rate allocation and supports higher compression rates by prioritizing perceptually significant features. This approach minimizes the impact of noise and reduces artifacts, thereby improving speech recognition performance. Additionally, this approach leverages psychoacoustic findings by tuning different spectral bands

to capture specific types of information—low frequencies enhance speech intelligibility measured by Short-Time Objective Intelligibility (STOI) [43], mid frequencies focus on articulation, and high frequencies capture speaker identification measured using speaker classification. This targeted quantization not only improves model efficiency but also provides a basis for more controlled speaker-specific voice clone text-to-speech or voice conversion applications.

1.3 Organization of the Thesis

The thesis is organized into six chapters, covering the following content:

Chapter 1 sets the stage for this thesis by outlining the motivation behind the research, discussing existing challenges in the field, and clearly defining the research objectives. Additionally, this chapter provides a brief overview of the contributions made by this work.

Chapter 2 offers an overview of speech representation learning and its applications in downstream automatic speech recognition systems. It reviews state-of-the-art adaptation techniques aimed at enhancing noise robustness, focusing on full fine-tuning methods, and then explores alternative approaches using parameter-efficient fine-tuning for more efficient adaptation. The chapter also examines the role of neural audio codecs in creating discrete unit speech representations. Additionally, it describes common open-source datasets used to build these models and discusses the metrics and tools employed to evaluate their performance.

Chapter 3 introduces deHuBERT, a novel framework for self-supervised speech representation learning. It addresses the challenges previous models faced in adapting to noisy speech for downstream speech recognition tasks. This chapter presents the Barlow Twins learning framework, designed to learn representations with reduced noise and channel redundancy, enhancing domain adaptation. Additionally, it outlines the methodology, experimental designs, and evaluations to demonstrate the effectiveness of these noise adaptation strategies.

Chapter 4 introduces deep filter-tuning, a parameter-efficient fine-tuning framework for adapting foundational pre-trained models to downstream tasks. It highlights the challenges of adapting large-scale models to noisy speech for speech recognition tasks, particularly under resource constraints. The chapter proposes an effective adaptation module, inspired by prompt tuning variants and feature-wise linear modulation, which extracts content from frozen representations while reducing noise. Additionally, it outlines the methodology, experimental designs, and evaluations to demonstrate the effectiveness of this noise adaptation strategy.

Chapter 5 introduces MUFFIN, a neural audio codec that employs multi-band spectral vector quantization to tokenize speech. This model enhances generative quality by leveraging the perceptual entropy bound, resulting in high fidelity and noise-reduced reconstructions. The chapter discusses the limitations of previous speech quantization approaches and explains how multi-band processing significantly improves the efficiency of learning tokenized speech representations. Additionally, it details the methodology, experimental designs, and evaluations used to demonstrate the effectiveness of this noise adaptation strategy.

Chapter 6 concludes the thesis by summarizing the main contributions and observations from the study. It also presents suggestions for future research that could extend and refine the findings discussed. This final chapter provides a modest overview of the study's impact and potential directions for continued exploration in the field.

Chapter 2

Literature Review

This chapter delves into the fundamental concepts of speech representation learning as applied to automatic speech recognition (ASR), charting the evolution from traditional methods to contemporary approaches. We begin by introducing several end-to-end architectures that have been prominent in encoding speech over recent years. Following this, we explore modern self-supervised learning (SSL) methods that leverage self-learning strategies to address challenges associated with training data scarcity. Further, we discuss specific training techniques for ASR tasks, focusing on full model fine-tuning and parameter-efficient fine-tuning. These methods are particularly relevant for larger foundational models, especially when training resources are limited. Finally, we detail commonly used datasets pertinent to this field, training python toolkits and describe the metrics used to evaluate model performance. This comprehensive overview not only highlights the progression in speech representation learning but also underscores the adaptive strategies used to enhance model training under resource constraints.

2.1 Overview of Speech Representation Learning

2.1.1 A Brief Historical Context

Speech processing has traditionally relied on extracting key handcrafted features from audio signals to enable machines to understand and process spoken language

effectively. Two commonly used techniques, spectrograms [60] and Mel-Frequency Cepstral Coefficients (MFCCs) [61], have played a pivotal role in this domain by transforming raw audio waveforms into structured and interpretable representations. These methods have been instrumental in a wide range of applications, including speech recognition, speaker identification, and emotion analysis.

The spectrogram, a time-frequency representation, is created by segmenting an audio signal into overlapping time windows, applying a Fourier Transform to each segment, and visualizing the spectral content over time. This results in a 2-dimensional representation, where the x-axis corresponds to time, the y-axis to frequency, and the intensity to the amplitude of the frequency components. Spectrograms excel at capturing the temporal dynamics of speech, such as shifts in pitch, harmonics, and formants, making them invaluable for analyzing acoustic patterns and understanding the rhythmic and tonal nuances of speech.

On the other hand, Mel-Frequency Cepstral Coefficients (MFCCs) offer a biologically inspired alternative, designed to mimic human auditory perception. Their computation involves mapping the signal's frequencies to the Mel scale, which reflects the non-linear sensitivity of human hearing. After applying a logarithmic transformation to approximate perceived loudness, the data is further processed using a Discrete Cosine Transform (DCT) to generate compact feature vectors. MFCCs effectively capture the shape of the vocal tract, which is crucial for distinguishing phonemes, while filtering out less relevant information such as pitch. This makes them particularly suited for tasks that require fine-grained phonetic analysis.

In the formative stages of speech representation learning, these traditional representations frequently served as inputs for deep learning models aiming to achieve end-to-end speech recognition. This early phase was characterized by pivotal research questions that sought to leverage deep learning more effectively, particularly in (1) processing long sequences, (2) reducing inference latency, and (3) scaling models to accommodate an increasing volume of data with enhanced training efficiency. An illustration of the evolution and goals is summarized in Figure 2.1.

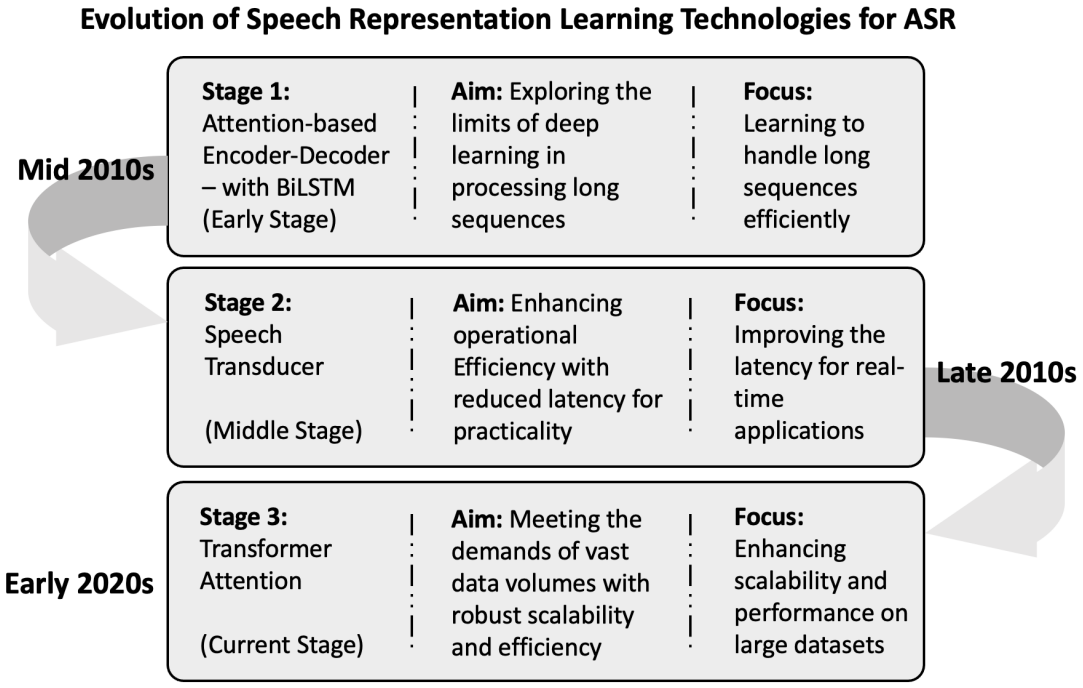


FIGURE 2.1: An illustration of the evolution of speech representation learning technologies for Automatic Speech Recognition (ASR), highlighting three key stages in the technological development. Stage 1 focuses on using attention-based encoder-decoder [1] models with BiLSTM to process long sequences efficiently. Stage 2 progresses to speech transducers [2] aimed at improving latency for real-time applications. The current Stage 3 employs Transformer [3] attention mechanisms to meet the challenges posed by vast data volumes, emphasizing scalability and enhanced performance on large datasets.

Early methodologies were primarily focused on extracting salient features from traditional representations, which were then utilized to train attention-based encoder-decoder networks, capable of transcribing spoken content into text [1]. This period marked the inception of using machine learning to effectively handle long sequences and decode human speech, with a dual emphasis on enhancing both the accuracy and efficiency of speech recognition systems.

As the field advanced, the imperative for methodologies that could enhance operational efficiency and reduce latency in real-time applications intensified. This urgency catalyzed the development of Recurrent Neural Network Transducers (RNN-T) [2], engineered specifically to deliver faster response times. Such enhancements made them exceptionally suited for interactive environments, including virtual assistants and real-time communication platforms. The fundamental motivation for these innovations was to construct models that could not only comprehend human

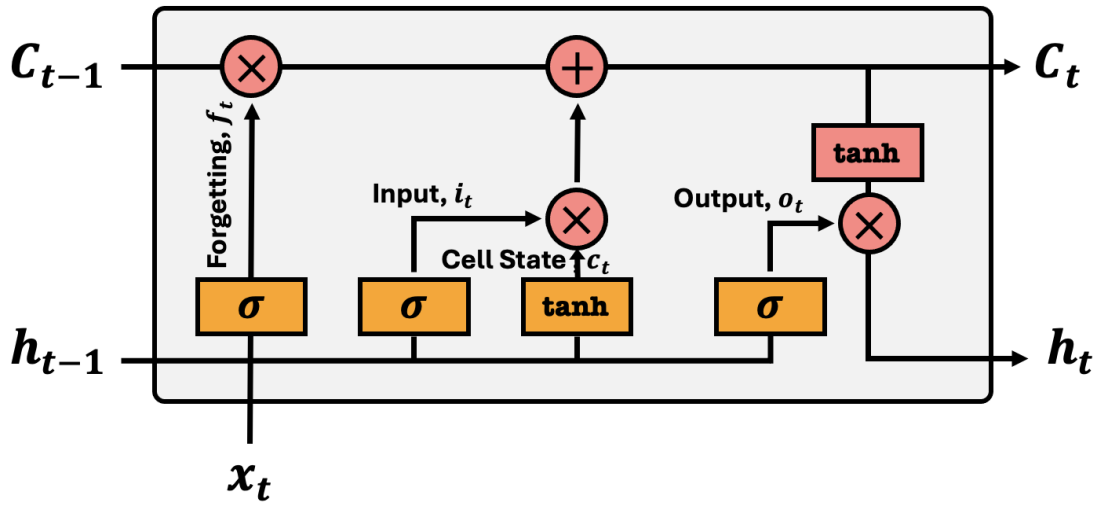


FIGURE 2.2: Diagram of an LSTM cell showcasing the flow and transformation of data through its internal gates: input gate, forget gate, output gate, and cell state.

language but also facilitate seamless and instantaneous interactions with users.

Responding to the escalating complexity and volume of data, the adoption of Transformer models [62, 63] represented a significant evolution within the field. These models, employing self-attention mechanisms, process entire sequences of speech data simultaneously–a notable departure from the sequential processing paradigm inherent in earlier models. This approach has facilitated remarkable scalability and efficiency, enabling the effective handling of complex tasks across multiple domains. The introduction of Transformers has redefined industry benchmarks, expanding the possibilities within speech recognition technology and catalyzing new avenues for both research and practical applications.

In the subsequent sections, we will explore these neural architectures in detail, underscoring the specific neural computations and their pivotal contributions to advancing speech representation learning for automatic speech recognition. We will also examine how the integration of self-supervised learning objectives into these frameworks can propel further enhancements in performance on downstream tasks.

Attention Based Encoder Decoder. The attention-based encoder-decoder model architecture has emerged as a cornerstone, fundamentally comprising an

encoder and a decoder, as presented in Figure 2.3. The encoder processes the entirety of the input MFCC sequence, denoted as X, to extract high-level representations H. These encapsulate critical phonetic, prosodic, and linguistic attributes of the speech, which the decoder then utilizes to transcribe spoken content into text (Y). Historically, these networks have relied on recurrent neural network (RNN) frameworks [64, 65], specifically employing bidirectional long short-term memory (BiLSTM) networks, recognized for their proficiency in capturing temporal dependencies within data streams. In particular, the hidden state vector, h_t, within the sequence $H = (h_1, ..., h_t)$, is represented as follows for t sequences:

$$h_t = \left[\overrightarrow{h_t}, \overleftarrow{h_t}\right] \tag{2.1}$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are the forward and backward hidden state outputs at the time frame t of the BiLSTM. An illustration of a single forward direction LSTM is presented in Figure 2.2. The backward direction adopts the same structure as the Figure but reverses the order from h_t to h_{t-1} instead.

Here, the bidirectional approach of BiLSTMs significantly enhances the model's contextual understanding, a critical factor in the accuracy and efficacy of speech recognition systems. However, managing longer input sequences introduces notable computational challenges and potential decoding inefficiencies. To mitigate these issues, subsampling techniques are frequently adopted [66], effectively reducing sequence lengths and thereby enhancing processing efficiency without compromising the richness of the data necessary for precise recognition.

Central to this architecture is the attention mechanism, which perform task-relevant weighting with their corresponding textual outputs. This dynamic focusing not only improves decoding accuracy but also facilitates a more nuanced understanding of the contextual nuances in speech [67].

Nevertheless, the reliance on RNNs, including BiLSTMs, introduces inherent limitations such as susceptibility to the vanishing gradient problem [68] and constraints in parallel computation, which can significantly impede the scalability and efficiency of training processes. Furthermore, the autoregressive nature of these models complicates their application in real-time speech recognition scenarios, necessitating adaptations to accommodate streaming requirements [69].

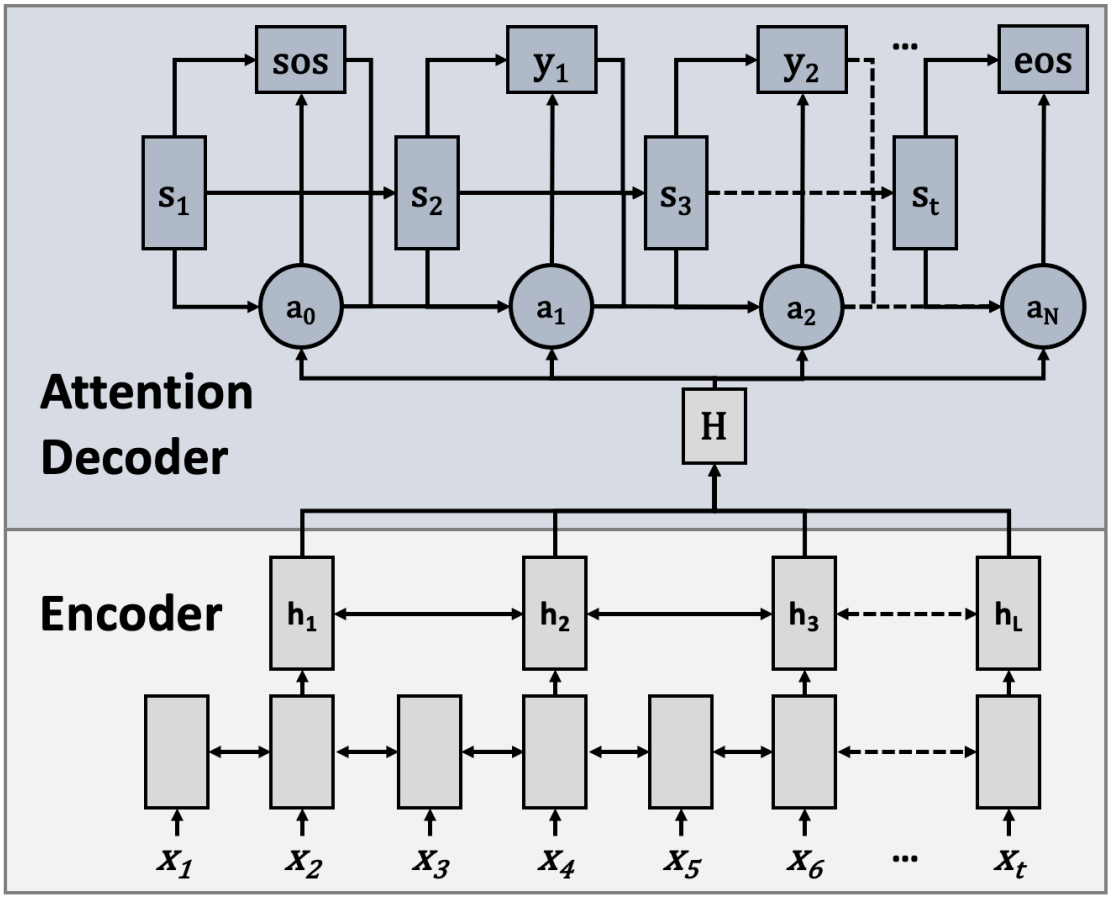


FIGURE 2.3: An architectural framework of the attention-based encoder-decoder. The encoder maps the input sequence X into a high-level feature sequence H, and the decoder helps to generate the sequence of characters or subwords Y through the attention module.

This discussion not only elucidates the operational principles and contributions of attention-based encoder-decoder architectures but also critically examines the challenges they present in contemporary speech recognition tasks. It underscores the imperative for ongoing innovation that adeptly balances complexity with computational efficiency, ensuring the evolution of speech recognition technologies to meet the demands of real-time applications.

Recurrent Neural Network Transducer. An RNN Transducer (RNN-T) model [2] is widely used in speech representation learning and consists of three primary components, namely an encoder, a prediction network and a joint network. A graphical representation of the framework is shown in Figure 2.4. The encoder, typically constructed with RNN architectures such as LSTM or BiLSTM, processes

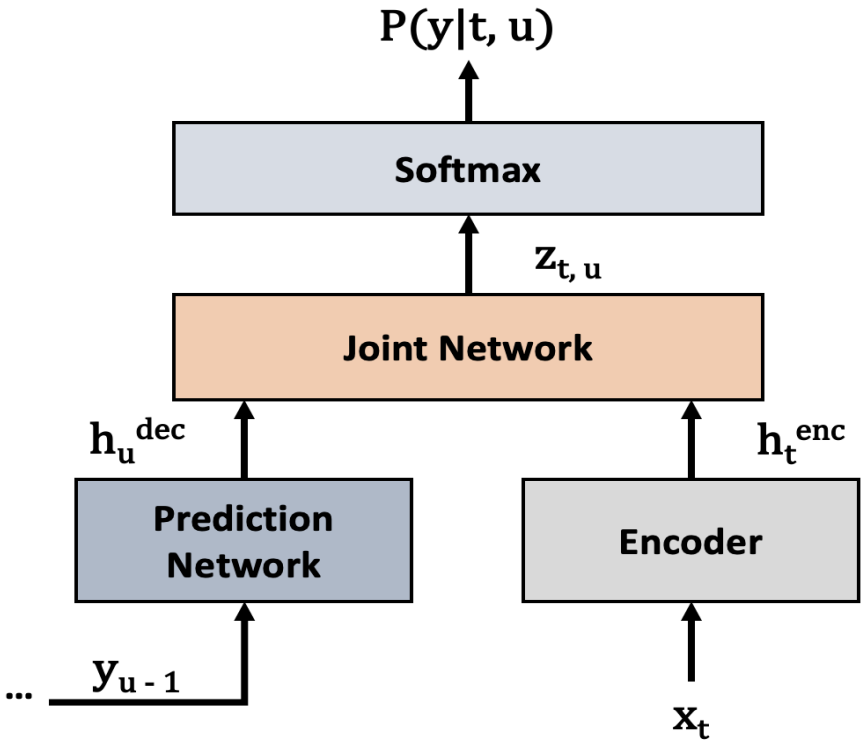


FIGURE 2.4: An Architectural Framework of the RNN-Transducer: The encoder, typically constructed with RNNs, encodes the acoustic features to produce latent features, denoted as h_t^{enc}. The prediction network, also built on RNNs, takes the predicted output from the previous time step, y_{u-1}, to generate another latent feature, h_u^{dec}. Both sets of latent features are then inputted into the joint network. This network performs a linear fusion, conditioning on the audio samples of current and past outputs, to predict the word token $y_{t,u}$

the sequential spectral features X and maps them into high-level latent speech representations h_t^{enc}. These representations encode the temporal and spectral characteristics of speech, serving as a compact and informative basis for downstream tasks such as transcription. That is,

$$h_t^{enc} = \text{Encoder}(x_t) \tag{2.2}$$

Then, the prediction network acts as a language model (LM) that learns the interdependencies within the output label sequence with an RNN function. It maintains a hidden state s_u and outputs the value h_u^{dec} for any label position $u \in (1, N)$. We

express the computations as the followings:

$$\begin{aligned} s_u &= \text{RNN}(W_{ys} y_{u-1} + W_{ss} s_{u-1} + b_s) \\ h_u^{dec} &= W_{sh} s_u + b_h \end{aligned} \tag{2.3}$$

where $W_\star$ is the trainable weights for its corresponding variable and b is the trainable bias. Then, a joint network takes in the outputs of the encoder and the prediction network, h_t^{enc} and h_u^{dec}, to calculate the label distribution at the output location u using a forward-backward algorithm.

$$\begin{aligned} z_{t,u} &= W_{zz} \cdot \sigma(W_{tz} h_t^{enc} + W_{uz} h_u^{dec} + b_z) + b_{zz} \\ P(y|t,u) &= \text{Softmax}(z_{t,u}) \end{aligned} \tag{2.4}$$

where σ is a non-linear activation function.

Since the joint network uses both latent acoustic and language features to calculate the conditional probability distribution of our prediction, RNN-T models the interdependence between the input sequence and output sequence, accomplishing the joint training of the acoustic and language models.

In comparison to the previous attention-based encoder-decoder network, the RNN-T model is capable of fulfilling online streaming speech recognition as it is able to process the features from the acoustic and language model concurrently [2]. Moreover, during decoding, it does not need to search over a large decoder graph that is typically computationally expensive. Instead, Bagby et al. [70] has shown to utilise a Viterbi beam search algorithm [71] to achieve similar accuracy as the traditional ASR models. Nevertheless, RNN-T is built mainly on the RNN function, which also inherits the same problems of gradient vanishing and incapability in parallel training as previous framework. Furthermore, it does not have an efficient aligning mechanism as attention-based models, which creates a lot of unreasonable data alignment between input and output. Therefore, training within this framework is more challenging, and it is often preferable to pre-train the prediction network and encoder to achieve better performance [72].

Speech Transformer. The Transformer model has emerged as a foundational element in the evolution of deep learning, particularly revolutionizing the field

of natural language processing (NLP). Originally conceived for machine translation, this model has outperformed traditional Recurrent Neural Networks (RNNs) across a multitude of NLP tasks, thanks to its unique self-attention mechanism. This mechanism enables simultaneous processing of all elements in the input data, effectively bypassing the sequential dependency constraints that hampered earlier architectures like RNNs. Beyond NLP, the Transformer has been adeptly applied to other sequential modeling tasks, including speech recognition, where it has consistently delivered substantial performance enhancements.

Distinct from the previous aforementioned models that rely on sequential data processing, the Transformer employs an internal mechanism to learn direct relationships between any two elements within a sequence. This method not only refines the output representations but also crucially distills the features that most significantly impact the task outcome. Furthermore, the architecture's inherent support for parallel computation drastically reduces training times and enhances operational efficiency, a critical advantage in large-scale deployment scenarios.

The speech transformer model consists of N_e blocks of encoders and N_d blocks of decoders, as depicted in Figure 2.5. It receives an input spectral sequence X and applies a subsampling module with two convolutional layers of stride two that reduce the sequence length by half in each layer. Since Transformer does not contain recurrence in its computations, a sinusoidal positional encoding is added to encode position information, allowing the model to be aware of the order of sequence before feeding to the encoder and decoder stacks.

Then, the encoder encodes the intermediate features into a set of high-level vectors using a multi-head attention network. The multi-head attention network computes the attention map in parallel with three argument matrices, i.e. "key", "query" and "value", where these three matrices are split into equal-sized vectors to be processed by the h independent self-attention head. The multi-head setup serves as a computation parallelization trick rather than power expansion. Subsequently, the processed features are concatenated to reconstruct the original sequence length. Further details can be found in [3].

The decoder block resembles the encoder block except for an additional sublayer between the feed-forward network and multi-head attention. This sublayer, known

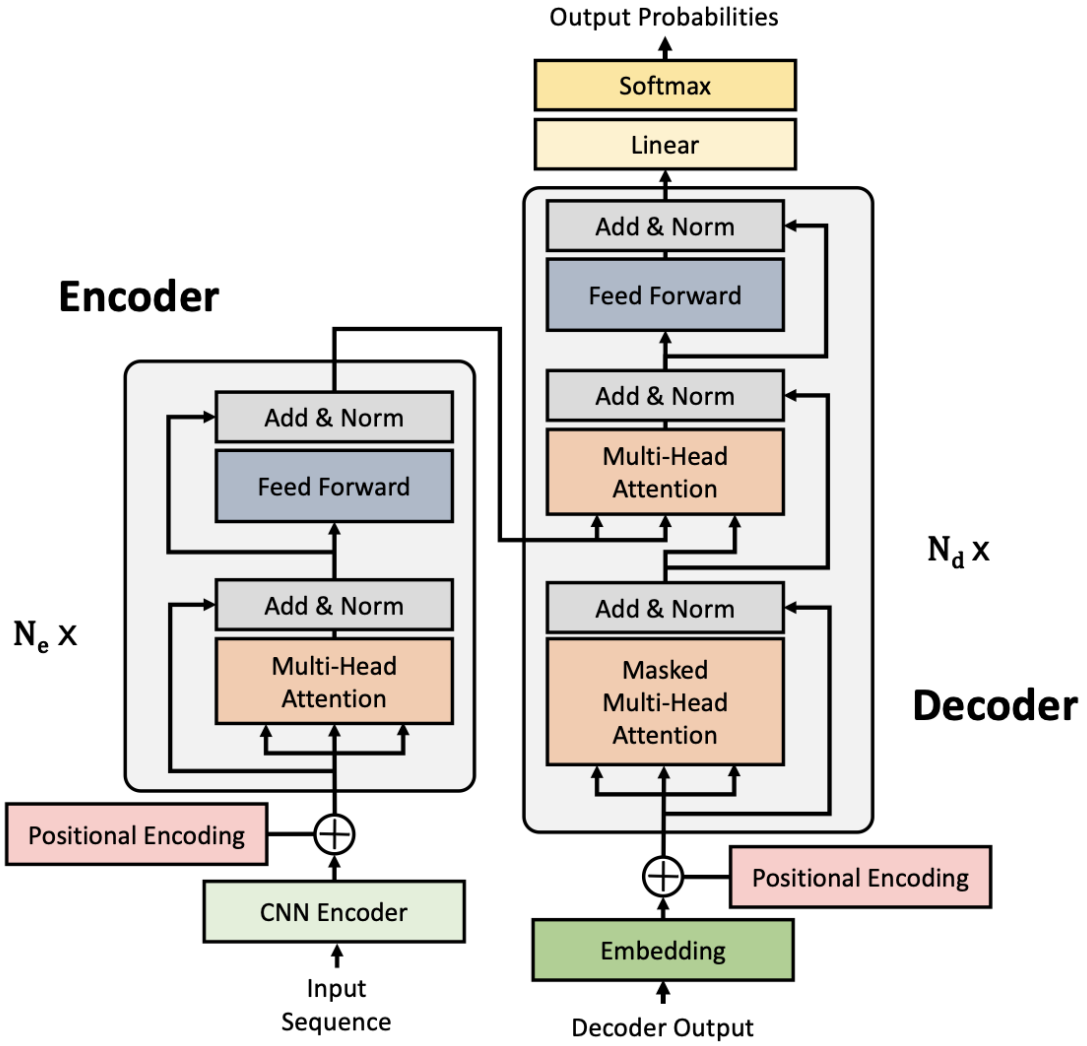


FIGURE 2.5: An architectural framework of the speech transformer. It consists of a subsampling convolutional encoder to reduce the sequence length of an acoustic input X, and several blocks of autoregressive encoder and decoder are used to generated the attended feature representations to predict the transcribed words.

as cross-attention, aligns the encoder and decoder outputs. Key and value vectors are derived from the encoder, while the query vector is generated from the decoder. In addition, attention masking is applied in a causal autoregressive manner. This prevents information from future positions from influencing the generation of features for the current position.

However, the model architecture has a quadratic complexity in sequence length $O(N^2)$. The computational complexity derives mainly from the complexity of $N \times N$ attention matrix, which increases the computational latency that limits its

applicative function. For instance, low latency is especially important when we are running the recognition application on-device on a small memory constraint tablet-PC or smart phone. Furthermore, the trainable model parameter is massive most of the times. This would require a substantial amount of labeled training data to build a decent recognition system, which is nevertheless extremely costly and rare for some low-resource languages. To address the data resource issue, modern representation learning uses self-supervised learning framework to utilize unlabeled data for initial pre-training. This pre-trained model will then transfer its prior knowledge to the downstream fine-tuning task. Additionally, the framework is optimized for a more end-to-end process, reducing the need for handcrafted features in the overall system. The following section will provide a comprehensive overview of the model architecture to underscore the significance of architectural designs such as self-supervised learning.

TABLE 2.1: Summary table of the pros and cons of three main model architectures used in speech representation learning

Model Architecture	Pros	Cons
Attention Based Encoder Decoder	-Excels in handling long-range dependencies	-Requires substantial computational resources
	-Flexible integration of context	-May struggle with real-time applications due to latency
	-High accuracy in end-to-end models	
RNN-Transducer	-Good for streaming applications	-Can suffer from vanishing gradient problem
	-Allows incremental output generation	-Generally slower training compared to attention models
	-Efficient at processing long sequences without performance degradation	
Speech Transformer	-Very effective at parallel processing	-High memory consumption for long sequences
	-Superior performance in modeling global dependencies	-More complex to implement and optimize compared to RNNs (Requiring more training dataset)
	-Scales well with increased data and computational power	

Quick Summary. In general, the landscape of speech representation learning has been profoundly shaped by these three main model architectures: the Attention-Based Encoder-Decoder, RNN-Transducer, and Speech Transformer. These frameworks have established themselves as the fundamental and most commonly used approaches in the field, setting the standards for how speech data is processed and understood before the advent of self-supervised speech representation learning [73, 74]. Their robust capabilities and distinct advantages have made them crucial for many real-world deployment applications, where the need for efficient and accurate speech recognition is paramount. Each architecture brings unique strengths and faces specific challenges, catering to different requirements and operational environments. A summary of their pros and cons, which highlights how each model performs under various conditions, is presented in Table 2.1.

2.1.2 Speech Representation in the Modern Context

In the modern context, there is an increasing use of raw waveform speech as input to neural models for speech representation learning. Using raw waveforms as input for deep learning offers distinct advantages over conventional handcrafted features like spectrograms or Mel-Frequency Cepstral Coefficients (MFCCs) [24]. Traditional methods rely on predefined transformations, such as windowing, frequency binning, and applying the Mel scale, which can introduce biases and discard critical information, including phase and fine-grained temporal details [75]. In contrast, raw waveforms retain the complete signal, enabling deep models to learn directly from the data and discover task-specific features without relying on domain-specific assumptions. This end-to-end learning approach simplifies the pipeline, integrates feature extraction with classification, and adapts more flexibly across tasks, from speech recognition to environmental sound classification [76]. Moreover, raw waveform models can capture subtle patterns and phase information that are often crucial for tasks like speaker identification or emotion detection [77, 78]. While they require sophisticated architectures and larger datasets due to high dimensionality, the ability to fully exploit the rich information in raw waveforms makes them a powerful alternative, particularly in scenarios where traditional features may fail to generalize or capture essential details. However, as we aim for higher quality speech representations, larger models are often employed with exponential scaling

of parameter sizes. To prevent these larger models from overfitting, we need even more training data.

To address the challenges associated with low-resource data settings, self-supervised speech models have emerged as a transformative approach in speech representation learning. These models utilize large quantities of unlabeled raw waveform data to learn robust and generalizable representations, significantly reducing the dependence on annotated datasets. By designing self-learning tasks such as predicting masked regions of the speech signal [79, 80], distinguishing between temporal segments [81], or reconstructing noisy or altered inputs [82], self-supervised models are able to uncover intricate patterns and relationships within the data.

Prominent models like Wav2vec 2.0 [83], HuBERT [84] and Data2vec [85] have demonstrated exceptional versatility and performance in extracting meaningful features from raw waveforms. These features can then be fine-tuned for downstream tasks, including automatic speech recognition, speaker identification, and emotion detection, achieving state-of-the-art results even in low-resource scenarios. Likewise, we will introduce the key speech representation learning components of these architectures in the following paragraphs. Moreover, the evolutionary trajectory that has culminated in the advancements of these models is briefly outlined in the timeline presented in Figure 2.6. This illustrates the pivotal shift from supervised end-to-end speech representation learning to self-supervised learning objectives.

Wav2vec 2.0. Wav2vec 2.0 [83] builds on the earlier speech transformer encoder by introducing a self-supervised learning objective tailored for unlabeled datasets. This approach leverages a vector quantization module combined with bidirectional masking predictive coding (MPC) [80, 86] to capture rich acoustic and linguistic representations from raw audio data. The framework primarily comprises two components using a convolutional encoder that extracts low-level speech features and a contextual transformer that models long-range dependencies to produce high-level, task-agnostic representations. An illustration is shown in Figure 2.7. A raw waveform is first sent into the convolutional encoder to generate the sequence of latent vectors Z, designed to capture more of the acoustic meaning of the given speech. Here, the vectors are discretized by a vector quantizer and the discrete units are used in the later training loss computation. Importantly, the quantizer composes

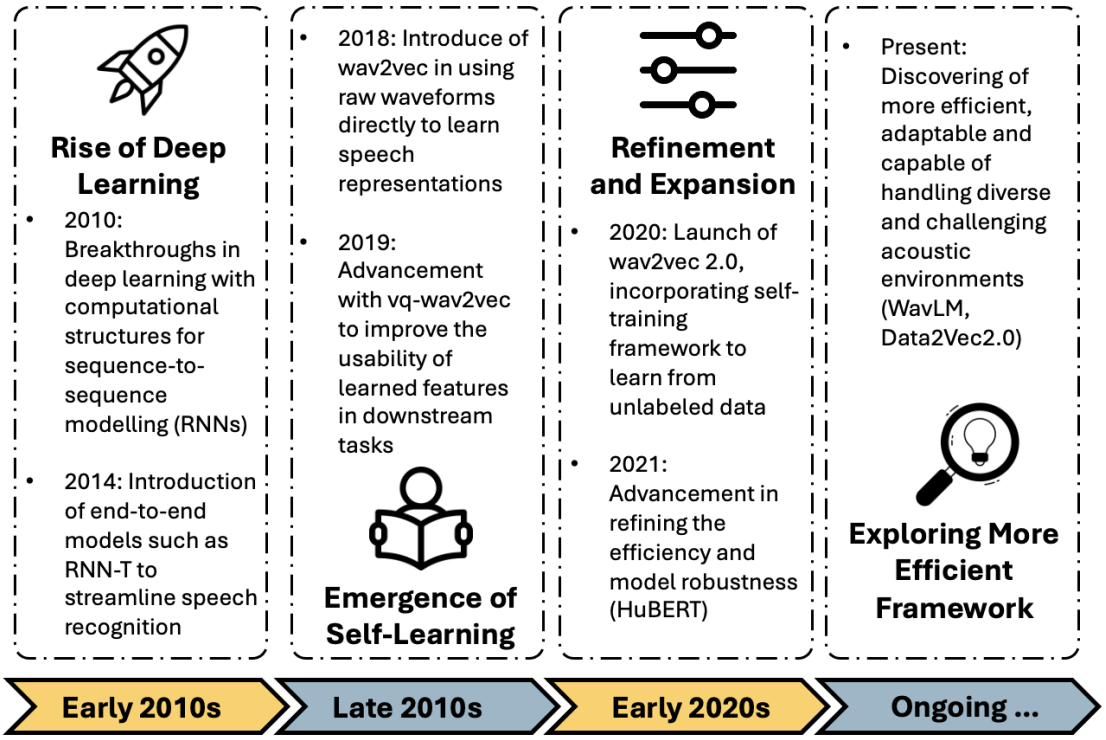


FIGURE 2.6: Timeline of key developments in modern self-supervised learning framework for speech representation.

multiple codebooks (i.e. product quantization) by selecting the quantized representations from different codebooks and concatenating them afterwards. This is because using a single codebook would tend to cause mode collapse in which the module fails to exploit the diversity of the codebook and only selects from a subset of the codewords. Formally, given G codebooks with V entries $e \in \mathbb{R}^{V \times d/G}$, one entry from each codebook will be selected. A linear transformation is performed after the concatenation. The probabilities for choosing the entry from v-th codebook of group g are

$$p_{g,v} = \frac{\exp(l_{g,v} + n_v)/\tau}{\sum_{k=1}^{V} \exp(l_{g,k} + n_k)/\tau} \tag{2.5}$$

where $l \in \mathbb{R}^{G \times V}$ denotes the logits from the linear projection of the encoded representations, $n = -\log(-\log(u))$ and u are the uniform samples from $U(0, 1)$, and τ is a non-negative temperature hyperparameter. During the forward pass, the codeword i in group g is choosen by $\text{argmax}_j p_{g,j}$ and in the backward pass, the true gradient from the outputs of the Gumbel softmax [87] is used. During training, we apply masking to the encoded latent vectors at Z, similar to the motivation of [86] by randomly sampling without replacement a certain ratio p of all time steps T to

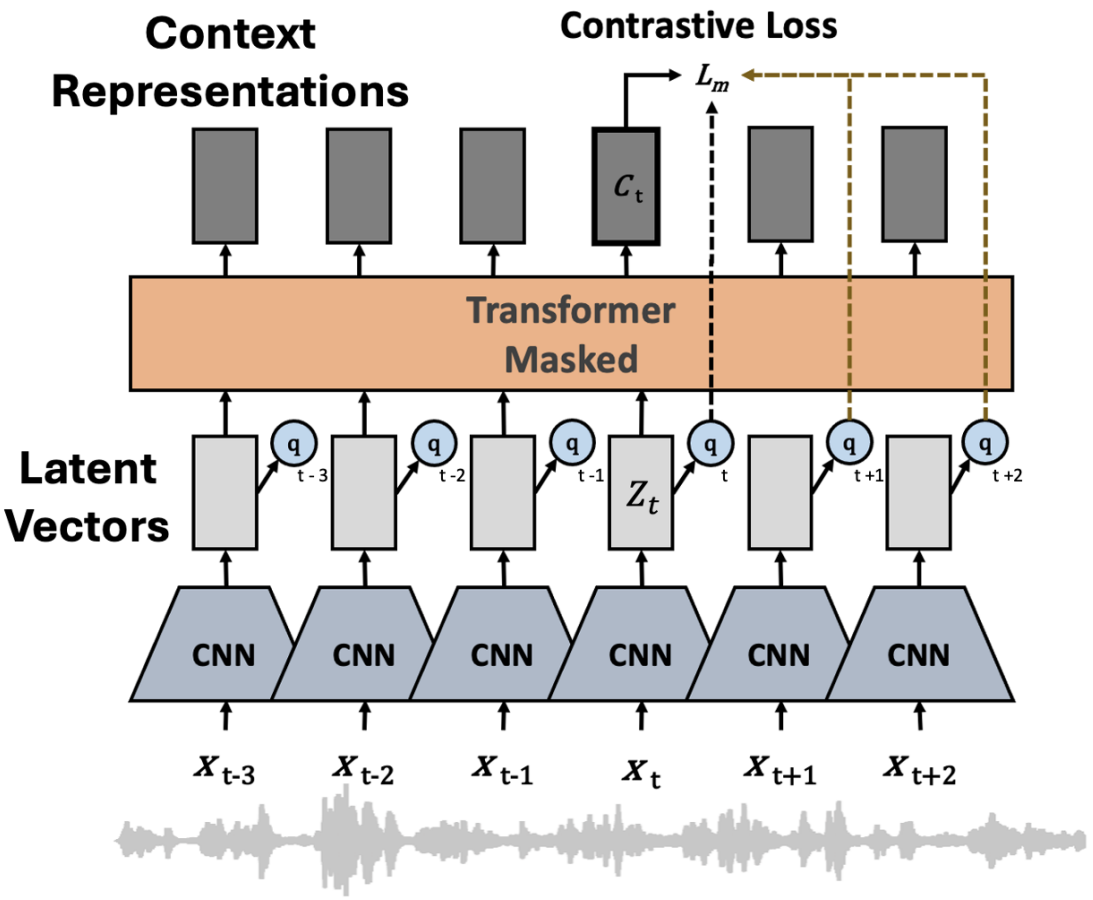


FIGURE 2.7: An architectural framework of Wav2vec 2.0. The model is built on a CNN encoder and a contextual Transformer that learns from contrastive coding between the quantized speech representations from CNN output and context representations that is encoded from the raw waveform.

be the start of the indices and then mask the subsequent M consecutive time steps from every sampled index. Note that these masked indices may overlap. Finally, the model is optimized on the loss function as defined by

$$L = L_m + \alpha L_d \tag{2.6}$$

where L_m is the contrastive loss, L_d is the codebook diversity loss and α is the choosen optimal hyperparameter.

The contrastive loss, measuring closeness using Euclidean distance, is computed by

$$L_m = -\log \frac{\exp(sim(c_t, q_t)/\kappa)}{\sum_{\tilde{q} \sim Q_t} \exp(sim(c_t, \tilde{q})/\kappa)} \tag{2.7}$$

where $sim(a, b)$ is the cosine similarity of the dot product computed by $a^\top b/\|a\|\|b\|$ between the quantized speech representations and context representations, C, derived from the transformer's output. It enhances its understanding of sequence-based semantic meanings by inferring the masked content in relation to each quantized unit. Note that the sequential length of C, Z and q remain the same in model forwarding.

And the diversity loss is denoted by

$$L_d = \frac{1}{GV} \sum_{g=1}^{G} -H(\bar{p}_g) = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \bar{p}_{g,v} \log \bar{p}_{g,v} \tag{2.8}$$

While pre-training this system to its full potential demands a substantial volume of unlabeled data, this process can be computationally costly and time-intensive. Fortunately, pre-trained weights derived from open-source corpora are readily accessible through various modeling toolkit sources such as Fairseq and ESPnet. [88–90].

HuBERT. HuBERT which stands for Hidden Unit BERT, is a self-supervised speech representation learning framework that employs masked predictive coding (MPC) to learn robust speech representations. The framework builds upon previous approaches by introducing an innovative strategy for learning from intrinsic input hidden units. Instead of relying on a contrastive objective function, HuBERT assigns pseudo-labels to encoded contextualized representations, which are derived from intermediate latent features or handcrafted features like Mel-Frequency Cepstral Coefficients (MFCCs).

Specifically, the system first generates pseudo-labels by clustering features such as MFCCs in the Euclidean space using simple discrete latent variable models, including K-Means [91] or Gaussian Mixture Models (GMMs) [92]. A convolutional waveform encoder then processes the raw waveform to produce a sequence of acoustic features. These features are passed through a masking function r, inspired by approaches like BERT [86], which corrupts the input sequence X to create $\tilde{X} = r(X, M)$, where M represents the masked region of a subset of the total sequence length T.

Finally, the masked prediction model takes $\tilde{X}$ as the input and predicts the intrinsic pseudo-labels generated from the start of the masked and unmasked time steps. The loss function for this is defined by

$$L = \alpha \cdot \sum_{t \in M} \log p_f(z_t|\tilde{X}, t) + (1 - \alpha) \cdot \sum_{t \notin M} \log p_f(z_t|\tilde{X}, t) \tag{2.9}$$

where $\alpha \in (0, 1)$ is the normalized weights given to each components. The first term denotes the masked time steps, whereas the second term denotes the unmasked regions. Nevertheless, to improve the target quality, cluster ensembles can be used to create pseudo targets of different granularity. In addition, these target labels are refined on the second iteration of the training. This time, the intermediate representations are used instead of the MFCCs. The illustration of the framework is shown in Figure 2.8.

In general, the above two frameworks have contributed to many SOTA performances in various downstream tasks and on multiple languages. Besides, they have laid the ground work for a few successful and interesting branched off, for example the Wav2vec-C [93] and the Wav2vec-U [94]. However, these models are usually trained on an individual open-source speech corpus that only covers a specific domain environment. A few investigations [95–97] have been conducted to explore the robustness of these models from the perspective of domain shift, where the data for pre-training, fine-tuning, and testing will originate from different sources. Without surprise, the model struggles with domain shifts, and numerous studies underscore the critical importance of aligning the conditions between pre-training and testing data to achieve satisfactory speech recognition results. However, integrating data from multiple sources across various domains can enhance the generalization capabilities of the learned representations. While this approach incurs a minimal cost, there is no guarantee that the aggregated data will sufficiently encompass the domain of the intended downstream task. To address this, [98–101] have been investigating to perform better adaptation or incorporating speech enhancement on the pipeline architecture to boost the encoded speech representations for improved results. However, a performance gap between in-domain and out-of-domain inferencing still persists. Consequently, it remains an unresolved question how we can refine our pre-training model architecture to bolster the robustness of our downstream model.

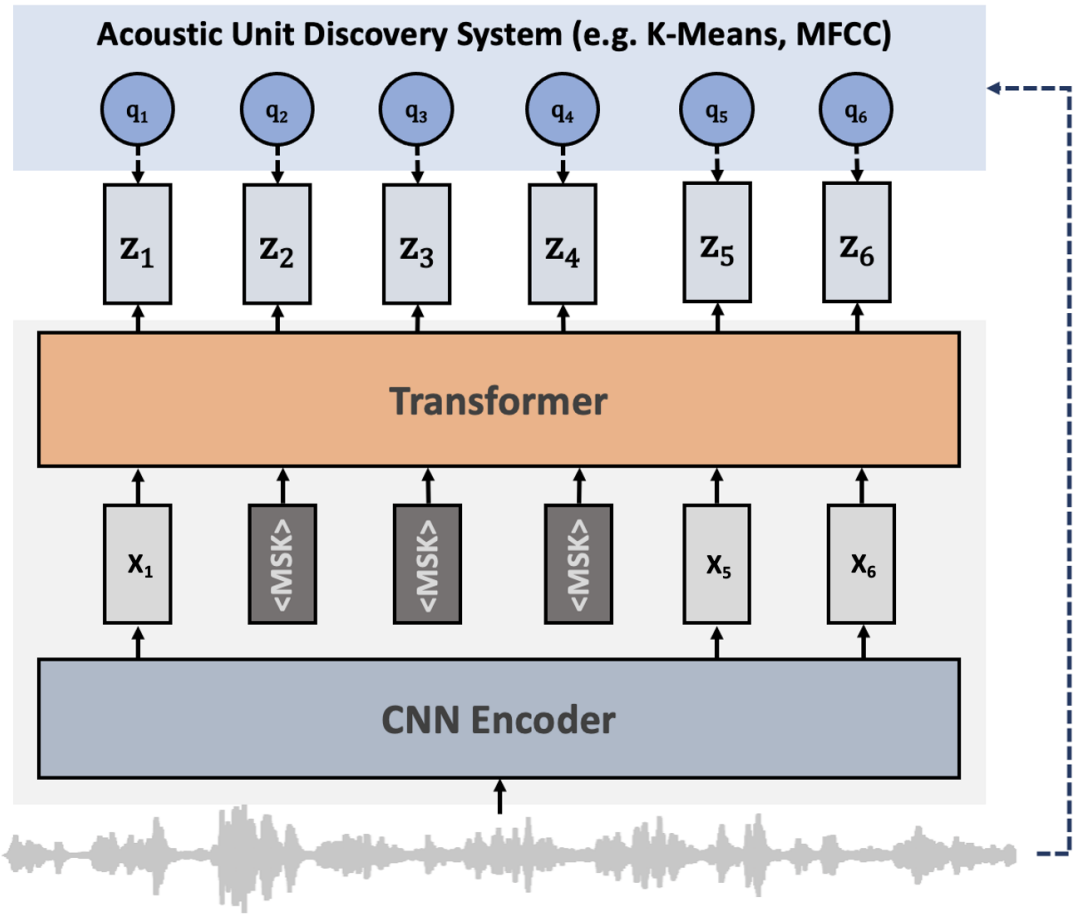


FIGURE 2.8: An architectural framework of HuBERT. The model is built on a CNN encoder and a contextual Transformer that learns from predictive coding using the acoustic unit discovery system from MFCCs and intermediate latent representations.

Data2vec 2.0. To date, Data2vec 2.0 uniquely introduces a modality-agnostic framework for self-supervised speech representation learning by unifying learning paradigms across speech, vision, and text. The architecture builds upon a teacher-student framework [102], where the teacher network generates smoothed target representations for the student to predict. The teacher network, updated using an exponential moving average (EMA) of the student model parameters [103], i.e., $\Delta \leftarrow \tau\Delta + (1-\tau)\theta$ where τ follows a linearly increasing schedule from a strting value τ_0 to a final value τ_e over τ_n updates. After reaching τ_e, the value of τ is kept constant. This EMA-based update mechanism allows the teacher network to provide stable and smoothed latent target representations Z_t, which encode high-level contextual information from raw waveform inputs, ensuring robust and meaningful supervision for the student network. An illustration of the network is

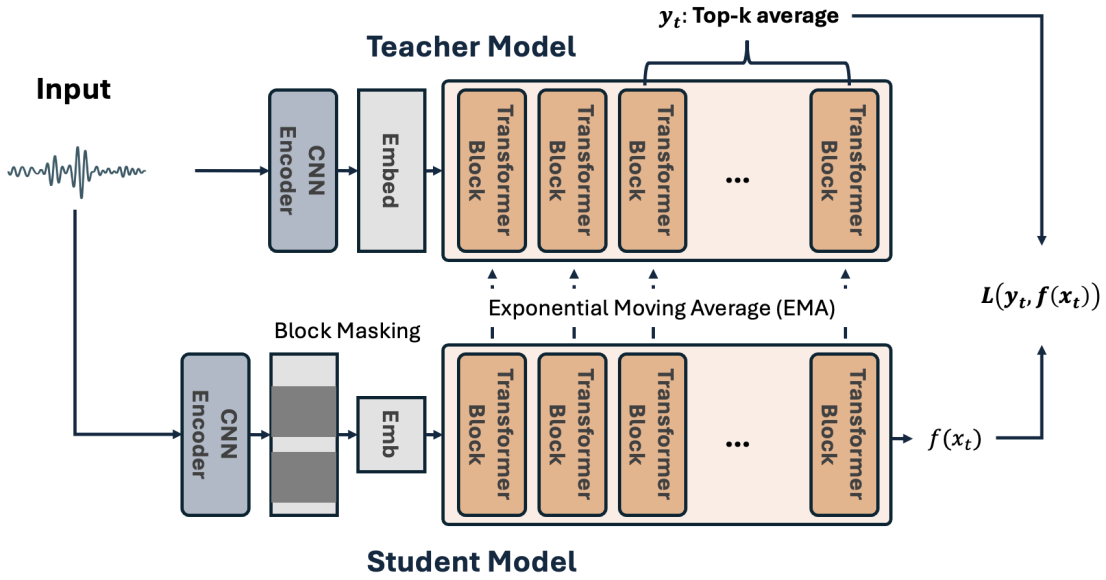


FIGURE 2.9: Architecture of Data2vec 2.0 for speech representation learning. The framework consists of a convolutional encoder that processes raw waveform inputs into low-level feature representations, followed by a masking module that selectively masks a portion of the feature sequence. The masked features are then passed through a transformer-based student network to produce contextualized representations. A teacher network, updated using an exponential moving average (EMA) of the student parameters, generates smoothed latent target representations. The student network learns by minimizing the Mean Squared Error (MSE) between its outputs and the teacher's targets at masked time steps. This architecture effectively captures high-level contextual information, enabling robust self-supervised learning across diverse downstream tasks.

presented in Figure 2.9.

Similar to other self-supervised speech encoder, the student model, consisting of a convolutional encoder and a transformer-based contextual network, processes the input speech waveform X, which is corrupted by a masking mechanism similar to BERT. The resulting masked feature sequence $h^{\tilde{enc}}$ is passed through the student network to generate contextualized representations $h^{\tilde{ctx}}$. During representation learning, the optimization objective of Data2vec 2.0 focuses on minimizing the Mean Squared Error (MSE) between the prediction of the masked regions by the student's contextual outputs $h^{\tilde{ctx}}$ and the teacher's targets Z_t using the formula:

$$\mathcal{L} = \frac{1}{|M|} \sum_{t \in M} \|h_t^{ctx} - Z_t\|^2 \tag{2.10}$$

where M represents the set of masked time steps in the input sequence. This

TABLE 2.2: Summary table of the pros and cons of the self-supervised learning framework for used in speech representation learning

Model Architecture	Pros	Cons
Wav2vec 2.0	-Highly effective at learning rich representations from raw audio. -Demonstrates significant improvement in speech recognition tasks. -Efficient use of unlabeled data through contrastive learning.	-Requires large amounts of compute and data to train effectively. (i.e., Quantize units) -Contrastive learning setup can be complex to tune.
HuBERT	-Improves over wav2vec by clustering hidden units of speech thus capturing more nuanced speech features. -Reduces dependency on labeled data. -Versatile in adapting to different downstream tasks.	-Training involves multiple stages which can be computationally intensive. -Requires careful selection of k clusters in K-Means to optimize performance.
Data2vec 2.0	-Generalizes across different modalities (audio, text, vision), increasing its utility. -Leverages a teacher-student model setup to learn from unlabeled data. -Demonstrates robustness across various languages and domains.	-Complex model architecture can be challenging for model distillation. -Significant memory overhead due to the need to simultaneously store and process the complex, high-dimensional latent representations generated by both the teacher and student models.

objective promotes an effective distillation of knowledge from the teacher to the student, enhancing the learning process. Note that the framework employs the novel masking strategy that takes the style of masked autoencoder (MAE) [104] which removes the masked information from the input feature sequence to improve efficiency. Additionally, this also removes the ability to store information in the activations of masked time-steps which makes the training task more challenging, encouraging the student network to infer the missing information [105]. This approach allows the model to learn more robust and contextually rich representations. Unlike methods that rely on discrete pseudo-labels or contrastive objectives,

Data2vec 2.0 directly aligns the student network's outputs with continuous latent targets generated by the teacher network. This design simplifies the learning pipeline, enhances the model's adaptability, and facilitates better generalization across various downstream tasks, including automatic speech recognition, speaker identification, and emotion detection.

Quick Summary. Similarly, we conclude the pros and cons of the three important self-supervised learning models, highlighting their unique strengths and challenges, in Table 2.2. This analysis provides a clear comparison, underscoring how each model's design and methodology influence its effectiveness of different approaches to speech representation learning.

2.2 Adapting Speech Representation for Noise-robust ASR

2.2.1 Model Adaptation Fine-Tuning

Domain Adaptation involves fine-tuning pre-trained speech representations on target domain-specific data to align the general-purpose knowledge encoded during pre-training with the unique characteristics of the target domain. This process is crucial in scenarios where the pre-trained model, trained on broad and diverse datasets, may face challenges in handling specialized tasks, such as recognizing noisy speech.

Noise adaptation in speech recognition includes various methodologies designed to augment model robustness against diverse acoustic disturbances. A prevalent technique is naive multi-condition training [106], in which model parameters are tuned using a dataset containing a spectrum of domain-specific noises. This strategy conditions the model to better recognize and process speech under those specific noise conditions. While simple and effective, its efficacy heavily depends on the availability and diversity of domain-specific training data. This dependency often limits the model's ability to generalize across a broader array of noise environments, particularly for communities with fewer data resources.

To overcome the shortcomings of naive multi-conditioning, advanced techniques have been developed to refine the optimization framework. Adversarial domain adaptation [107] is a notable technique that employs a discriminator to challenge the model into producing domain-agnostic features, effectively concealing non-generalizable, domain-specific characteristics. This setup not only preserves essential task-relevant information but also minimizes domain-induced variances. Complementarily, the Information Bottleneck [98, 108] method focuses on reducing the impact of noise on speech representations. By compelling the model to retain only crucial information and filter out superfluous noise-related elements, this approach enhances the model's generalizability and robustness across diverse noisy environments, significantly reducing dependence on noise-specific training data. We introduce these methods for noise robust adaptation in the following paragraphs.

WavLM. WavLM builds on SSL speech representation learning paradigms, such as HuBERT, by introducing architectural enhancements and noise-robust training strategies. It retains the core structure of a transformer-based speech encoder while incorporating a gated relative position bias into the multi-headed self-attention mechanism. Originally proposed by Press et al. [109], relative position biases improve the model's ability to encode long-range temporal dependencies while emphasizing nearby contextual content, as illustrated in Figure 2.7. This addition is particularly critical for capturing localized speech features, such as phonemes and transitions, without losing the broader context. This modification ensures that the contextualized representations generated by WavLM are both comprehensive and sensitive to the structure of speech.

To improve robustness in noisy environments, WavLM augments the training data with artificially generated background noise, simulating real-world conditions. This multi-condition training forces the model to process noisy inputs while still learning to produce clean and reliable representations. The target labels, derived using the HuBERT approach of clustering clean speech features with K-Means, remain unaffected by the added noise. This creates an information bottleneck, requiring the model to denoise the input by filtering out noise-induced distortions and inferring clean target labels. This approach not only improves the model's noise robustness but also enhances its ability to generalize to diverse acoustic conditions.

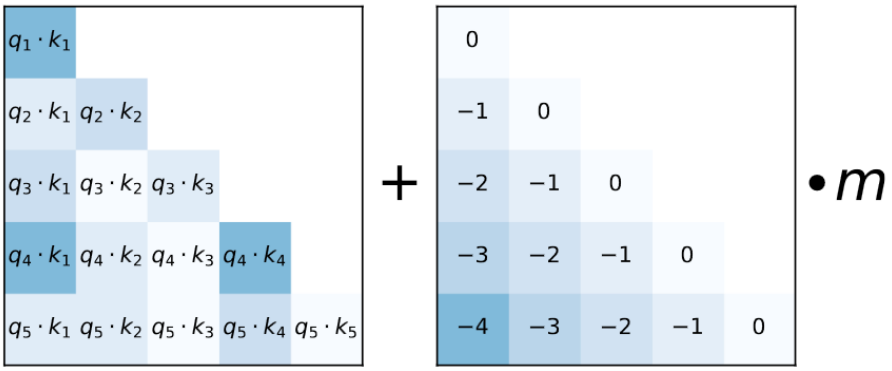


FIGURE 2.10: An illustration of relative position bias operation. The module adds a constant, head-specific bias m to each attention score, modifying the dot product $(q_i \cdot k_j)$ between query and key vectors. The bias is applied based on the positional distance between tokens, as shown on the right. The softmax function is then applied to the adjusted scores, while the rest of the attention computation remains unchanged. The scalar m is fixed and not learned during training, allowing the model to prioritize closer tokens efficiently.

By integrating these architectural and training innovations, WavLM significantly enhances speech representation quality, achieving strong downstream performance adapting to noise corrupted speech.

However, the relative position bias introduces additional latency due to the computation of the bias matrix, which scales with the sequence length and adversely affects inference speed. Additionally, the model's ability to generalize to noisy environments is highly dependent on the diversity of background noises used during data augmentation. This reliance can make it particularly challenging for the model to handle previously unseen noise types, limiting its robustness in real-world scenarios.

Wav2vec-Switch. Wav2vec-Switch [99] is an extension of the wav2vec 2.0 SSL speech representation learning framework. The core of the model is an information bottleneck optimization framework that employs a cross-target contrastive loss to reduce noise interference. This enhancement allows the model to learn resilient speech representations by aligning clean and noisy inputs in a shared feature space.

Unlike wav2vec 2.0, Wav2vec-Switch processes two streams of speech latent representations simultaneously. One stream is derived from the original, clean input, while the other comes from the same input corrupted with background noise.

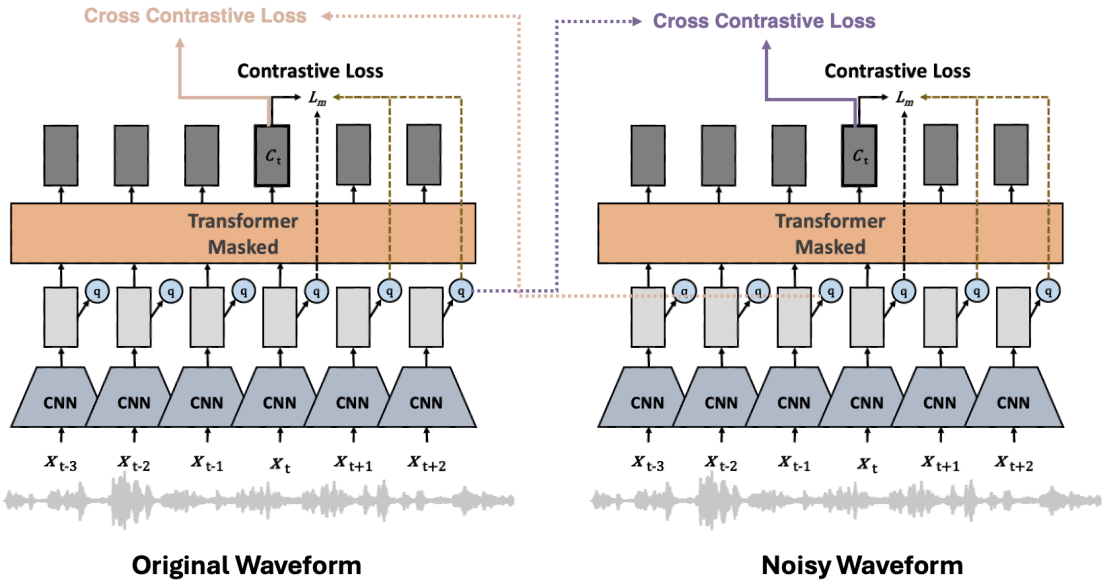


FIGURE 2.11: Architecture of Wav2vec-switch model. It processes two parallel streams of speech representations: one from clean audio and the other from noise-corrupted audio. It employs a cross-target contrastive loss to align these two streams, promoting the extraction of noise-invariant features.

The model is trained to predict targets across streams, where clean representations align with noisy targets and vice versa. This cross-stream prediction task pushes the model to extract features invariant to noise, minimizing the influence of background interference and emphasizing essential speech characteristics. A figure illustrating of the learning framework is presented in Figure 2.11.

The cross-target contrastive loss adheres to the computation specified in Equation 2.7, integrating both cross-context and quantized features. This approach effectively orchestrates the optimization process, aligning clean and noisy latent representations without necessitating additional modules within the base transformer encoder. However, fine-tuning such a large-scale model demands significant computational resources, particularly as recent developments aim to increase the model's parameter size. To alleviate these computational requirements, strategies for parameter-efficient fine-tuning are implemented, optimizing the model's performance while minimizing resource expenditure. We examine this in the next subsection to answer why we need a more parameter efficient fine-tuning method for domain adaptation.

2.2.2 Parameter Efficient Fine-tuning

In the contemporary landscape of deep learning, the growth of model sizes presents challenges, particularly in adapting these models for specific tasks such as noisy speech recognition. Traditional fine-tuning methods, which often involve extensive retraining of large-scale models, require substantial computational resources, making them impractical for real-time or resource-constrained environments. This has spurred the development of parameter-efficient tuning techniques [110–112], crucial for deploying deep learning models more effectively in noisy environments.

Parameter-efficient tuning specifically aims to adapt pre-trained models to new tasks by modifying only a small subset of the model's parameters. This approach is particularly beneficial for downstream tasks like noisy speech recognition, where models must be robust against a variety of background noises. Techniques such as the insertion of trainable adapter layers, the application of low-rank matrix factorization, and the use of soft prompts have proven to be effective. These methods freeze the entire pre-trained model and allow for subtle tuning of external neural components to improve their ability to filter and interpret speech from noisy inputs without the need for comprehensive retraining. This eases the resource constraints faced by specific user groups. We explore these methods in the following paragraphs and provide figure illustration in Figure 2.12.

Adapter Tuning. Adapters [4] are typically lightweight neural layers that are inserted into the transformer architecture of a model. In the context of transformers, each adapter consists of a bottleneck modules that include down-projection that reduces the dimensionality of the layer outputs, a non-linear activation function, and an up-projection that restores the dimensions, forming a residual connection. This design enables the model to learn task-specific features without a substantial increase in the number of trainable parameters.

LoRA. LoRA [113] works by decomposing updates to the weight matrices into low-rank factors. Specifically, for a weight matrix W in a transformer model, LoRA introduces two smaller matrices A and B such that the update ΔW is approximated as $A \times B$. This low-rank approximation alters the original weight matrix W to $W + A \times B$, where A and B are much smaller in size compared to W, resulting in a significant reduction in the number of trainable parameters. In

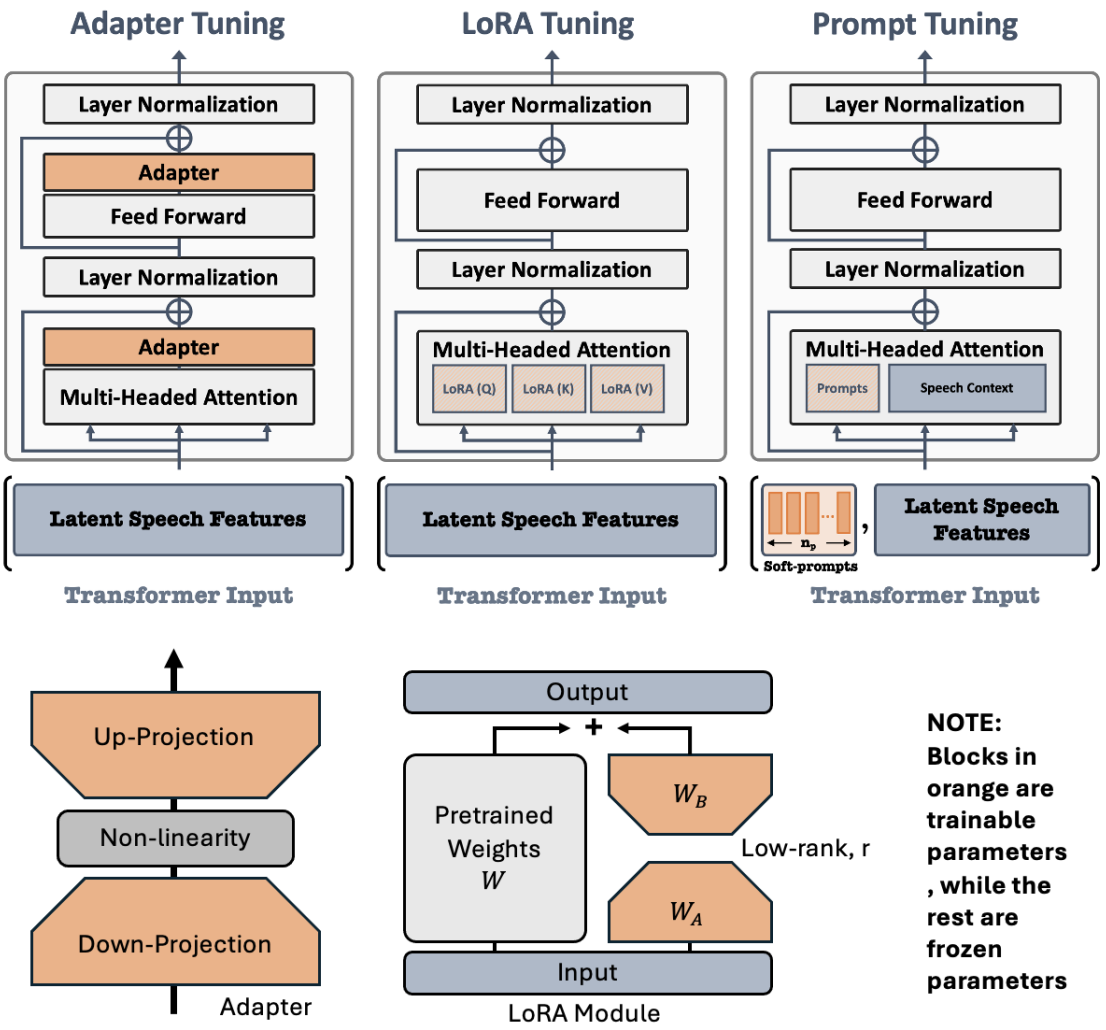


FIGURE 2.12: Illustration of three parameter-efficient tuning methods: Adapter Tuning, Low-Rank Adaptation (LoRA), and Prompt Tuning. This figure compares their integration into a transformer model architecture. Adapter Tuning involves inserting trainable layers between existing layers; LoRA applies low-rank updates to the weight matrices; and Prompt Tuning adds trainable prompt tokens at the input stage. Each method aims to enhance model performance on specific tasks with minimal updates to the pre-trained model's parameters, demonstrating their unique approaches to efficient model adaptation

most cases, LoRA is added to the attention modules of the speech transformer layers. However, adapting learned speech representations to a new downstream task that involves changes in noise levels or recognized languages can challenge the assumption of LoRA's low-rank updating. Such tasks often require higher-rank learning to handle the complexity introduced by changes in information effectively.

Prompt Tuning. Prompt tuning [114] involves adding a small set of trainable vectors, known as soft prompts, directly into the input sequence before passing it

through the model. These vectors are optimized during training while the rest of the model's parameters remain frozen. The soft prompts act as a form of "instruction" to the model, effectively tuning its output by adjusting the initial context it receives. This method allows the model to apply its vast pre-trained knowledge in a way that is contextually relevant to the new task. Nevertheless, if the pre-trained representations lack specific domain knowledge, effective adaptation may prove challenging. As such, we will study and address some of these problems in the coming chapter of this thesis.

2.3 Alternative Speech Representation

While self-supervised learning frameworks have gained popularity in recent years, the speech representations generated by Transformer models are continuous. To facilitate the integration of speech representations with large language models, researchers are exploring methods to tokenize these continuous embeddings [6, 115, 116]. Consequently, a new direction in speech representation learning is emerging, which involves leveraging neural audio codecs to encode speech signals into compact latent representations [82]. This approach is based on the principle that audio codecs are meticulously engineered to capture essential speech features such as intelligibility and perceptual quality. Specifically, the encoder processes raw audio waveforms, transforming them into high-dimensional latent representations that encapsulate the critical attributes of the audio signal. These representations are then quantized into discrete latent spaces, a process that preserves essential audio information into codes with a neural embedding (often referred to as a codebook), while optimizing storage and transmission efficiency. The resulting quantized tokens are extracted and employed as inputs to large language models (LLMs), facilitating the seamless integration of speech data into broader computational frameworks. Simultaneously, the decoder reconstructs the audio waveform from these quantized codes, ensuring that the tokenized speech contains all necessary information to accurately reconstruct waveforms with high fidelity and perceptual quality.

The current research appears to divide the neural audio codec into two primary areas: tokenization of speech and reconstruction of high fidelity speech, as outlined in the summarized schema in Figure 2.13 [59, 117–119].

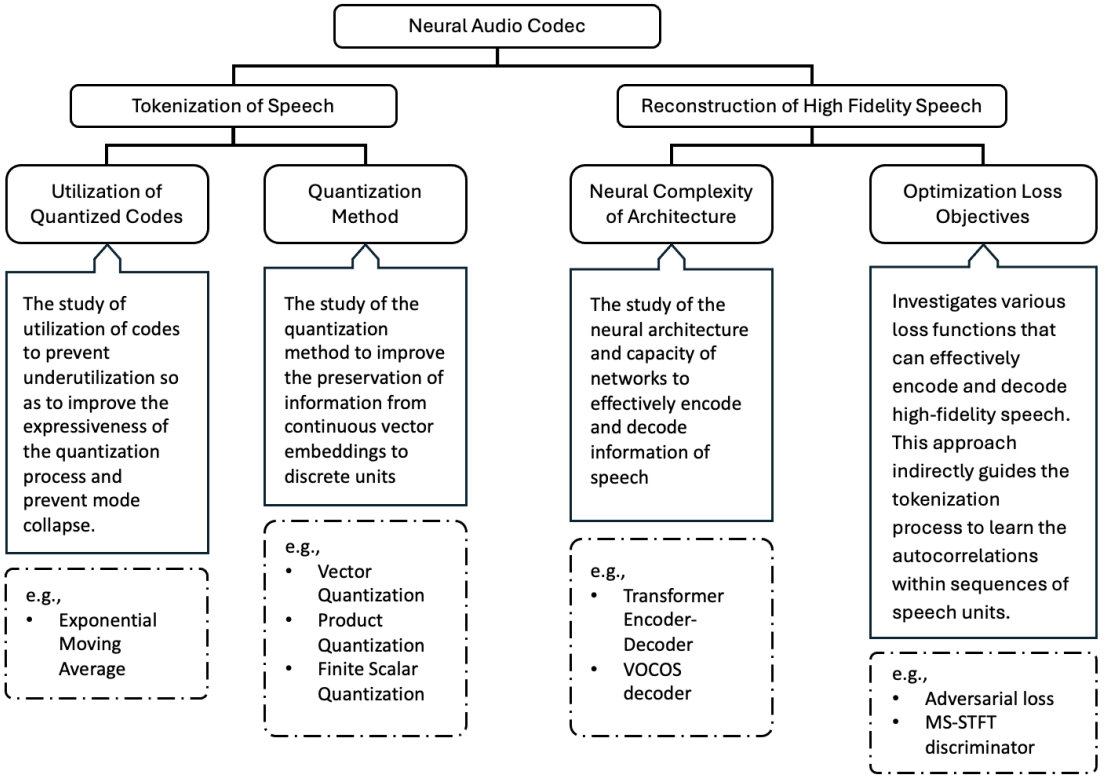


FIGURE 2.13: Schema illustrating the current research directions in neural audio codecs, focusing on the tokenization of speech and the reconstruction of high fidelity speech.

The research direction with tokenization of speech on code utilization focuses on maximizing discrete code usage within a codebook to achieve more diverse and expressive representations [120]. By encouraging higher code utilization, models allocate fewer samples to each code, enriching the granularity of the latent space and enabling more nuanced outputs. This strategy helps capture complex data distributions and enhances advanced generative tasks [121]. Activating a broader range of codes expands the model's representational power, improving its ability to capture underlying patterns and leading to better performance in large-scale learning systems. Moreover, effective quantization methods help preserve critical information during the inherently lossy tokenization process. By refining the mapping from continuous input signals to discrete representations, these methods mitigate the risk of data collapse and ensure that important details remain accessible for downstream tasks or reconstructions. As a result, well-designed quantization strategies bolster system stability, reduce the likelihood of mode collapse, and deliver richer, more reliable outputs

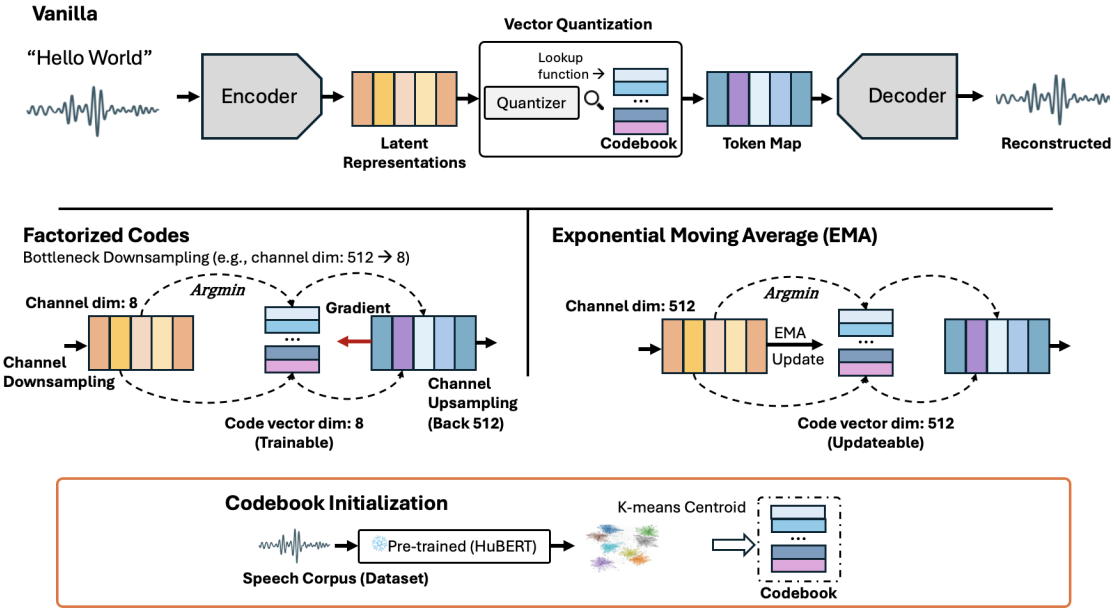


FIGURE 2.14: Overview of existing strategies to enhance the utilization rate of the vanilla vector quantization method, highlighting factorized codes, exponential moving average, and codebook initialization using pre-trained speech encoder knowledge for efficient starting configurations.

In speech reconstruction, encoder-decoder architectures are carefully optimized to ensure high-fidelity decoding from quantized tokens, preserving both the integrity and perceptual quality of the audio. Achieving this balance requires fine-tuning loss objectives for accuracy and computational efficiency. By producing high-fidelity outputs, these models retain crucial speech information, resulting in a more robust representation of underlying audio characteristics. Recent advances in neural network designs and learning paradigms [119, 122, 123] guide the development of these architectures and their loss functions, particularly those that enhance interactions between speech and language models. Subsequent subsections will examine these topics in greater depth, illustrating how existing model architectures implement the described schema. We will analyze the strategies they employ, providing concrete examples to highlight key design decisions and their impact on system performance.

Utilization of Quantized Codes. An ongoing challenge in vector quantization is achieving effective code utilization. In particular, mapping high-dimensional representations to discrete code units often results in underused codebooks, as seen in vanilla VQ-VAE architectures [58]. Suboptimal initialization can cause

a considerable portion of the codebook to remain inactive, effectively shrinking the usable set of codes, reducing the expressiveness of the quantized output, and degrading reconstruction quality.

To address this, recent audio codec models explore two commonly adopted strategies: factorized codes and exponential moving average, to enhance utilization rates [124–126], as depicted in Figure 2.14. Specially, factorized codes employ channel downsampling, a form of dimensionality reduction, to transform high-dimensional latent vectors into more manageable lower-dimensional sub-spaces. In these constrained spaces, the distributions of sub-vectors are not only closer but often overlap, increasing the variance among sampled code entries and enhancing the probability of diverse code utilization. Such overlap promotes balanced code usage across the spectrum, reducing the incidence of idle codes.

In contrast, the exponential moving average method updates the codebook vectors in a more stable and continuous fashion than direct gradient-based approaches. This technique ensures smooth and incremental adjustments, effectively preventing drastic fluctuations. Consequently, it helps preserve a uniform distribution of code embeddings, maintaining consistency with their initial uniform configuration. This process counteracts the collapse of certain code embeddings and thus preserves broader coverage of the latent space.

Besides, initializing codebook embeddings with insights from pre-trained speech models can significantly streamline the learning process for speech representation. By starting with embeddings that are already aligned with key speech features, the model more rapidly achieves meaningful and effective speech encoding. For instance, models such as HuBERT are used to extract the K-Means centroids from a speech corpus, which then serve as the initial settings for the codebooks, as detailed in the accompanying figure. This strategic approach accelerates the adaptation of the codebooks to relevant speech patterns, enhancing overall code utilization efficiency. Note that the challenge of code utilization is distinct from earlier self-supervised speech representation frameworks, where embeddings remain continuous. In contrast, vector quantization forces a discrete, code-based representation by performing a table lookup in the codebook to find quantized units. This discrete lookup process can be detrimental in cases of codebook collapse, which is crucial

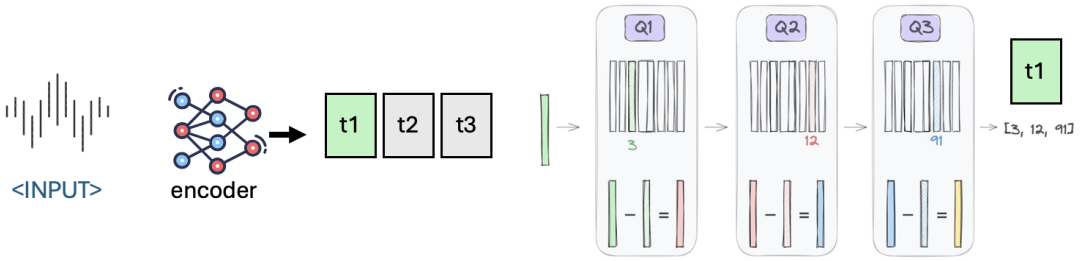


FIGURE 2.15: An illustration of RVQ. It refines the quantization process iteratively by computing the residual after each quantization step and encoding it using subsequent codebooks, drawing inspiration from gradient boosting techniques

for ensuring that the learned representations are both meaningful and expressive. This ultimately contributes to more robust speech modeling and generation.

Quantization Method. To enhance the preservation and representation of speech using discrete units, several advanced techniques have been developed. These include Residual Vector Quantization (RVQ) [82], Product Quantization (PQ) [127], and Factor Vector Quantization (FVQ) [128], each designed to capture more nuanced and effective representations. To illustrate, RVQ extends basic vector quantization by progressively refining the encoding of a signal through multiple quantization stages. Instead of relying on a single codebook to capture all the information, RVQ splits the quantization process into a series of smaller "residual" quantizations. Each stage encodes the remaining error left by the previous stage, enhances the preservation of speech information. A diagram is presented in Figure 2.15.

On the other hand, PQ simply employs a divide-and-conquer approach to vector quantization by splitting a high-dimensional input vector into multiple low-dimensional sub-vectors. Each sub-vector is independently quantized using its own codebook. The final representation is then formed by concatenating the discrete indices (or codewords) selected for each sub-vector. This splitting reduces the complexity associated with learning and maintaining a single large codebook for the entire space. Additionally, it proves particularly effective for very high-dimensional vectors, commonly found in deep learning features, as each sub-vector becomes more manageable.

Lastly, Factor Vector Quantization, as implemented by FACodec [128], organizes speech signals into distinct attribute subspaces for more precise and disentangled

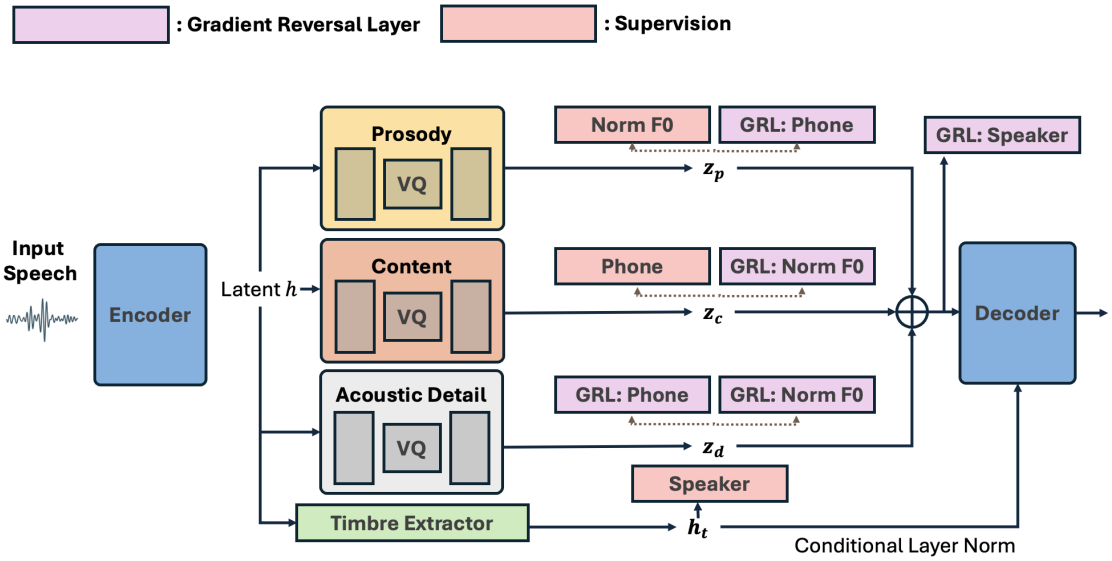


FIGURE 2.16: Overview of the FACodec architecture for speech representation learning. FACodec factorizes speech into prosody, content, timbre, and acoustic detail subspaces using a speech encoder, timbre extractor, three factorized vector quantizers (FVQ), and a speech decoder. The encoder processes raw audio into latent representations, while FVQs discretize prosody, content, and acoustic details. Timbre is extracted via a Transformer encoder and fused into the decoder using conditional layer normalization. Advanced techniques, including information bottlenecks, auxiliary supervision, gradient reversal layers, and detail dropout, ensure robust attribute disentanglement, enabling high-quality and expressive speech reconstruction.

representations. Each subspace is tailored to a specific speech attribute—such as timbre, prosody, speech intelligibility, or acoustic content—allowing for independent control over each facet. To maintain this separation, several attribute disentanglement techniques are employed: an information bottleneck facilitates low-dimensional quantization, attribute-specific supervision tasks (e.g., pitch prediction for prosody) enhance targeted learning, adversarial classifiers with gradient reversal layers purge undesired information, and detail dropout minimizes cross-attribute interference by selectively masking acoustic details during training. Other variant [59] includes contrastive learning of information from pre-trained HuBERT representations, which are rich in semantic meaning, to distill speech intelligibility into the first codebook in an RVQ setup.

By structuring speech in this manner, FVQ significantly improves the interpretability and modularity of the speech representations and enables high-quality audio

generation, as depicted in Figure 2.16. This approach marks a clear advancement over previous self-supervised learning models by providing more interpretable speech representations and greater controllability in downstream tasks through the selective manipulation of individual attributes.

Improving Neural Complexity of Encode-Decoder. Achieving higher-quality speech representations hinges on developing more sophisticated encoder-decoder frameworks. For example, Hifi-Codec [127] replaces the earlier SeanNet [129] model with Hifi-GAN [119], which leverages a wider receptive field to better capture phase variations and subtle details in speech signals. Similarly, Vocos [123] transitions from ResNet to ConvNeXt [130], integrating design principles from both convolutional and Transformer-based models—such as larger kernel sizes, fewer activation bottlenecks, and advanced normalization layers—to refine speech feature extraction. Further illustrating this trend, Moshi [6] adopts a Transformer-based NAC module, utilizing attention mechanisms for more effective information compression during both encoding and decoding. These innovations collectively underscore the importance of architecture choice in driving fidelity and robustness across modern speech processing tasks.

Optimization Loss Objectives. Finally, to optimize for high-fidelity reconstruction, there has been an ongoing effort to refine the learning objective to better capture phase shift information and patterns in speech signals, reducing the formation of artifact noises. To date, the SOTA approach captures the intricate phase information of the speech signal by incorporating a reconstruction loss for the predicted waveform and using adversarial learning with a multi-scale short-time Fourier transform (STFT) discriminator. This discriminator processes both the real and imaginary components of the audio signal, concatenated into a two-dimensional representation, and employs dilated convolutional layers to extract discriminative embeddings. These embeddings target multiple STFT window lengths, including 2048, 1024, 512, 256, and 128, enabling the model to accurately capture both fine and coarse-grained temporal details.

The adversarial loss for the generator is defined as:

$$\mathcal{L}_g(\hat{x}) = \frac{1}{K} \sum_k \max(0, 1 - D_k(\hat{x})) \tag{2.11}$$

where K represents the number of discriminators, and D_k assesses the quality of the generated waveform across different resolutions.

In addition, a relative feature matching loss is introduced for the generator, ensuring alignment between the generated and reference features across discriminator layers. Formally, the relative feature matching loss is defined as:

$$\mathcal{L}_{\text{feat}}(x, \hat{x}) = \frac{1}{KL} \sum_{k=1}^{K} \sum_{l=1}^{L} \frac{\|D_{k,l}(x) - D_{k,l}(\hat{x})\|_1}{\text{mean}(\|D_{k,l}(x)\|_1)} \tag{2.12}$$

where D_k represents the k-th discriminator, L is the number of layers in each discriminator, and the mean is computed across all dimensions. This loss function ensures that the reconstructed features are consistent with the input features across multiple granularities.

The discriminators are trained with a hinge-loss adversarial objective:

$$\mathcal{L}_d(x, \hat{x}) = \frac{1}{K} \sum_{k=1}^{K} \big[\max(0, 1 - D_k(x)) + \max(0, 1 + D_k(\hat{x})) \big] \tag{2.13}$$

To prevent the discriminator from overpowering the decoder during training, its weight is updated with a probability of $\frac{2}{3}$ when operating at a 24 kHz sampling rate. This balancing mechanism ensures stable adversarial training and optimal reconstruction performance. Lastly, the overall training loss is given by

$$\mathcal{L}_G = \lambda_t \cdot \mathcal{L}_t(x, \hat{x}) + \lambda_f \cdot \mathcal{L}_f(x, \hat{x}) + \lambda_g \cdot \mathcal{L}_g(\hat{x}) + \lambda_{\text{feat}} \cdot \mathcal{L}_{\text{feat}}(x, \hat{x}) + \lambda_w \cdot \mathcal{L}_w(w), \tag{2.14}$$

where $\lambda_t, \lambda_f, \lambda_g, \lambda_{\text{feat}}, \lambda_w$ are scalar coefficients that balance the contributions of different terms. $\mathcal{L}_t(.)$ and $\mathcal{L}_f(.)$ represent the reconstruction losses of the temporal and frequency domain, respectively. $\mathcal{L}_w(w)$ denotes the commitment loss, $\sum_{c \in C} \|z_c - q_c(z_c)\|^2$, which is typically used for codebook quantization optimization.

Importantly, this optimization stands apart from earlier self-supervised speech learning frameworks. While prior approaches often concentrate on masked prediction or high-level feature extraction, the emphasis here is on reconstructing latent speech representations back to the original audio. This design choice facilitates a deeper understanding of sequential relationships between frames, ultimately

enhancing both acoustic fidelity and generative flexibility in the learned representations.

2.4 Datasets, Training Tools and Evaluation

In this section, we introduce some of the commonly used open-source corpora for constructing speech representations and fine-tuning them for ASR tasks.

2.4.1 Overview of Open Source Data Corpora

The summarized details of the datasets, which vary in duration, context, and languages, are presented in Table 2.4.

TABLE 2.3: Summary table of available open-source noise dataset

Corpus	Description	Sampling Rate (kHz)	Size (Hrs)
MUSAN	Data is categorized by music, noise, and speech. The speech audio is derived from the Librivox project that collects book-reading audio. The music audio is collected from Jamendo, Free Music Archive, Incompetech and HD Classical Music. It contains music categories from Western art music to popular genres (e.g. Jazz, hip-hop, bluegrass). The noise data comprises 929 files of mixed noises, e.g. car idling, footsteps and animal noises, accumulated over 6 hours.	16	109
Freesound	The noise type is categorized into two classes (i.e. type-A and type-B). Type-A noise is relatively stationary, which includes "Metro", "Car" and "Traffic" noise, and type-B noise is relatively non-stationary, which comprises noises like "Cafe", "AC/Vacuum", "Babble" and "Airport/Station". Every individual noise type has 10 and 8 distinct audio streams in the train and test sets, respectively.	16	2

One of our goals in our research work is to study the noise robustness of the recognition system under different decibels of noises and out-of-domain noises. However, some of the corpora in this table are originally clean, e.g. LibriSpeech. Hence, we adopt a common approach by the research community that employs an

TABLE 2.4: Summary table of available open-source data

Corpus	Description	Language	Sampling Rate (kHz)	Size (Hrs)
LibriSpeech (LS)	1. Collection of data derived from audiobooks in the LibriVox project. 2. Contains training, development, and test sets categorized as clean and other based on audio quality. 3. Contains 2,338 unique speakers.	English	16	1000
Ted-Lium 3 (TED)	1. Audio is collected from TED talks in sphere format (SPH), and text transcription is automatically aligned to audio using the Kaldi toolkit 2. Contains 2,028 unique speakers.	English	16	425
CHiME 4	1. Contains two datasets. The first set is denoted as "real data", based on the portion of the Wall Street Journal (WSJ0) corpus with actual noisy environments, that includes pedestrian area, on a bus, cafe, and street junction. The second set denoted as " simulated data" comprises artificially mixed clean WSJ speech with background noise. 2. All utterances are recorded using a 6-channel distant microphone array and a close-talk microphone.	English	16	18
Switchboard (SWBD)	1. Audio collected from telephony speech around US region. 2. Contains dual-channel utterances.	English	8	286
LibriTTS	1. Collection of data derived from audiobooks in the LibriVox project. 2. Data includes original and normalized texts, contextual information, and excludes noisy utterances. 3. Contains 2,456 speakers	English	24	585
Libri-Light	1. Derived from open-source audiobooks from the LibriVox project. 2. Includes extensive unlabelled and labeled speech.	English	16	60,000

external noise dataset to make the audio noisy. This involves the artificial addition of random sampled noise to the clean audio at the desired signal-to-noise ratio. As

such, we provide some details to the available open-source noise data in Table 2.3.

2.4.2 Training Toolkits

In the rapidly evolving landscape of artificial intelligence, particularly in the realms of natural language processing and speech technology, two significant frameworks have emerged: Fairseq [88] and ESPnet [89]. Both toolkits serve as powerful platforms for developing state-of-the-art machine learning models, catering to different aspects of AI research and applications.

Fairseq, developed by Facebook AI Research, is a robust sequence-to-sequence learning toolkit that excels in a variety of tasks, including machine translation, text summarization, and speech-to-text modeling. Built on the PyTorch framework, Fairseq is designed for scalability and efficiency, supporting multi-GPU training and mixed precision training to effectively handle large-scale experiments. Its modular architecture allows researchers to implement custom models easily while providing pre-trained models that demonstrate competitive performance on standard benchmarks. Fairseq's versatility is further enhanced by its support for diverse tasks beyond machine translation, such as automatic speech recognition and self-supervised learning, making it a comprehensive tool for researchers aiming to push the boundaries of AI capabilities.

On the other hand, ESPnet (End-to-End Speech Processing Toolkit) focuses primarily on speech-related tasks. It provides an integrated framework for automatic speech recognition, text-to-speech synthesis, speech translation, and more. ESPnet adopts a unified end-to-end approach that simplifies the model training process by utilizing a single neural network architecture for various speech processing tasks. This toolkit leverages popular deep learning engines like PyTorch and Chainer, while also drawing on the Kaldi ASR toolkit for data processing and feature extraction. Notably, ESPnet emphasizes end-to-end learning paradigms that enhance model performance across different applications in speech technology.

2.4.3 Performance Evaluation Metrics

The most commonly used metric in evaluating the performance of ASR is the word error rate (WER). This metric composes three error terms, that is substitution errors, insertion errors and deletion errors.

$$\text{WER} = \text{Substitution Error} + \text{Deletion Error} + \text{Insertion Error} \tag{2.15}$$

where the three terms are computed by

$$\begin{aligned} \text{Substitution Error} &= \frac{\text{No. of substitution errors}}{\text{No. of ground truth words}} \\ \text{Deletion Error} &= \frac{\text{No. of deletion errors}}{\text{No. of ground truth words}} \\ \text{Insertion Error} &= \frac{\text{No. of insertion errors}}{\text{No. of ground truth words}} \end{aligned} \tag{2.16}$$

WER has a scale of $[0, +\infty)$. A perfect transcription with zero mistake will score the value of 0, whereas every mistake in the transcription will cause the error rate to be higher. Note that for some character-based languages (i.e. Chinese Mandarin, Cantonese), we use another metric similar to WER called Character Error Rate (CER). Instead of counting for word errors, it accounts for each mistake in the transcribed character, which is more natural for these languages.

Evaluating the quality of speech reconstruction involves measuring both speech quality and intelligibility. We introduce some commonly used key metrics for this purpose, as outlined below.

PESQ [42]: Perceptual Evaluation of Speech Quality is an internationally standardized method for assessing the speech quality of voice communications systems and codecs. It predicts subjective listening quality tests on a Mean Opinion Score (MOS) scale. PESQ is valuable because it simulates human perception and evaluates the quality of speech as it would be heard by an end-user.

STOI [43]: Short-Time Objective Intelligibility is a metric designed to predict the intelligibility of processed speech signals, especially in environments with background noise. It is commonly used to evaluate speech enhancement algorithms,

assessing how much an algorithm can improve speech comprehension in noisy conditions.

Mel-distance [131]: Mel-distance measures the perceptual distance between two speech signals based on their Mel Frequency Cepstral Coefficients (MFCCs). This metric is often used in speech and speaker recognition systems to quantify the similarity between the original and reconstructed speech, reflecting how well a model preserves timbral characteristics.

STFT [131]: STFT distance evaluates the spectral distance between the original and reconstructed signals using the Short-Time Fourier Transform. This metric provides insights into how accurately a model captures and reconstructs the spectral dynamics of speech, which are critical for maintaining the natural sound quality.

UTMOS [44]: UTMOS is a non-intrusive objective speech quality assessment metric that estimates the perceived quality of speech signals. It is designed to be highly correlated with human subjective ratings and can evaluate both narrowband and wideband speech quality.

ViSQOL [132]: The Virtual Speech Quality Objective Listener (ViSQOL) is a model for objective speech quality assessment that predicts the quality of voice calls and speech signals. It uses a spectro-temporal measure of similarity between a reference and a degraded audio signal to assess quality, making it useful for codec evaluation and network performance monitoring.

Each of these metrics offers distinct insights into the various facets of speech quality, from intelligibility to timbral fidelity, allowing researchers and engineers to finely tune and improve their audio processing models. Together, they provide a robust framework for rigorously evaluating the performance of speech reconstruction algorithms.

Chapter 3

Self-Supervised Speech Representation: Noise Robustness and Reduced Redundancy

3.1 Introduction

Transformers, known for their ability to model complex dependencies and contextual nuances, are the preferred architecture for developing cutting-edge speech systems. However, their effectiveness largely depends on access to extensive, annotated datasets, which poses a significant challenge in low-resource settings. In this chapter, we explore the use of HuBERT, a self-supervised learning model built on the structure of the speech transformer, to address the challenges of representation learning in such contexts. Despite its strengths, HuBERT's design is not optimally noise-robust. To enhance this, we introduce a novel training framework called deHuBERT, inspired by H. Barlow's redundancy-reduction principle. This framework modifies the HuBERT training algorithm by incorporating auxiliary losses that align the self- and cross-correlation matrices of pairwise noise-distorted embeddings with the identity matrix. This adjustment helps the model generate speech representations that are not only resilient to noise but also exhibit reduced channel-wise redundancy. We discuss the motivations behind our work, outline the methodology, and present an ablation study to demonstrate the enhanced noise

robustness of our model when applied to downstream ASR task. This chapter addresses the following questions:

- What are the shortcomings of recent innovations in SSL speech representations?
- How can we modify the recent architecture to improve its robustness to noise?
- How does our proposed work compare to the recent SOTA models?
- How well does our proposed work generalize to out-of-domain noises?

3.2 Motivation

Recently, self-supervised pre-training for speech has gained prominence, leading to significant advancements in creating effective automatic speech recognition (ASR) systems [83, 84], particularly for low-resource languages [133]. These successes are driven by the use of large volumes of unannotated utterances to develop universal speech representations that enhance downstream ASR tasks. Key techniques include contrastive predictive coding (CPC) [80], which predicts the next frame using a contrastive loss, and autoregressive predictive coding (APC) [134] that reconstructs future frames from past sequences.

While these works have shown promise, they typically focus on domains with relatively clean audio, such as LibriSpeech, and lack variability. In real-world environments, speech often includes background noises, reverberation, and other distortions, which can degrade ASR performance when there's a domain shift from the training data [96].

To address these challenges, Wang et al. [99] have adapted models like wav2vec 2.0 (w2v2) to include a contrastive loss that learns from cross-quantized targets between original and noisy pairs. Other studies [135] have used contrastive loss as a regularizer to enhance noise-reduced speech features. For example, Huang et al. [136] employed a teacher-student framework to encode denoising representations. Despite these advancements, noise robustness remains a challenge.

This chapter aims to improve the noise robustness of the self-supervised pre-trained HuBERT model for noisy ASR environments. We introduce a novel self-supervised training framework, disentangled HuBERT (deHuBERT), which incorporates a new pair of auxiliary loss functions to foster noise invariance in embedded contextual representations. Inspired by the Barlow Twins method [50]–originally designed to reduce redundancy in image vector representations–we adapt this approach for sequential modeling. This framework aggregates the cross-correlation matrix between embeddings from two identical networks fed with different noise-augmented samples, pushing it towards an identity matrix. This encourages the network to focus on consistent speech features across noisy variations, while reducing other background disturbances. Our experimental results show that deHuBERT consistently outperforms previous models in noisy conditions without losing accuracy on clean audio tests.

3.3 Methodology

In our work, we introduce "deHuBERT", a training algorithm that enhances the HuBERT model's ability to generate disentangled, noise-agnostic representations through a siamese-style architecture. This approach trains the model on two distinct noise-augmented versions of the input, each corrupted by a different noise type. These inputs are concurrently processed through a shared CNN encoder, which helps in producing robust and meaningful latent acoustic embeddings, similar to those intended by the original model design [84], yet enhanced to achieve more noise-resistant representations, as illustrated in Figure 3.1. Specifically, the training data for each input version is randomly augmented with one of several noise types, with signal-to-noise ratios (SNRs) ranging from 0 to 25 dB. The encoded feature representations, X and $\tilde{X}$, obtained from these noise-augmented speech through CNN encoder of HuBERT, are then forwarded through a common linear projection block, depicted in the figure, to produce the respective outputs Y and $\tilde{Y}$.

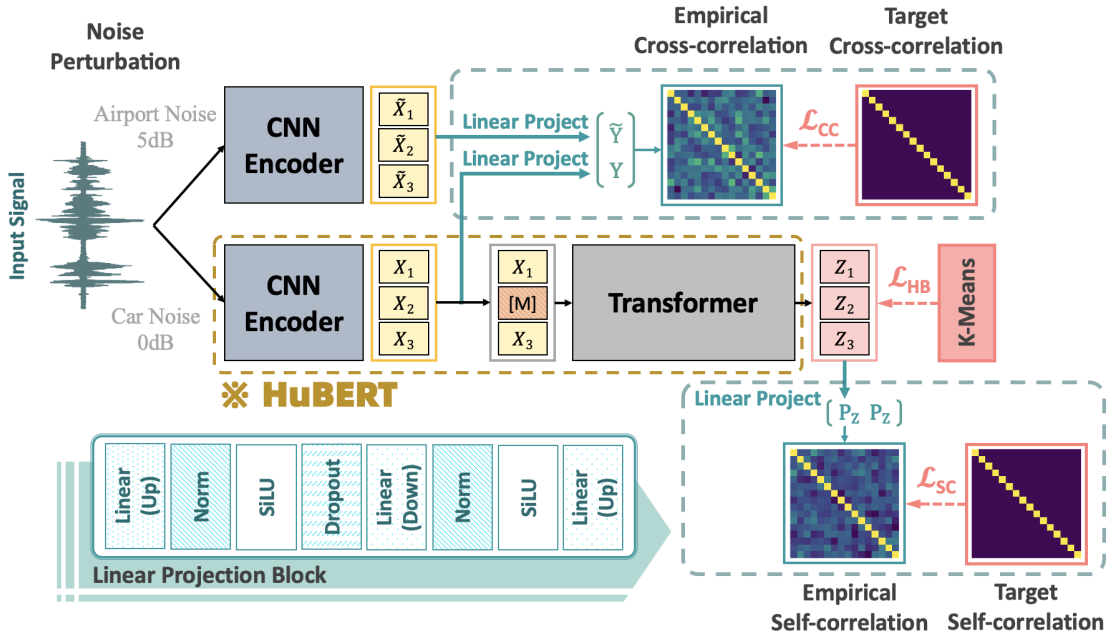


FIGURE 3.1: Architecture of deHuBERT for learning noise-Robust, redundancy-reduced representations. This framework features two parallel streams of speech, each augmented with different background noises, fed into the model. The learning objective is to minimize both self- and cross-correlation of the latent embeddings towards an identity matrix. This setup creates an information bottleneck, reducing the impact of noise and ensuring consistent embeddings across the parallel streams. Additionally, the correlation losses promote learning that reduces channel-wise redundancy.

We follow the approach introduced by Zbontar et al. [50] to determine the empirical cross-correlation (CC) matrix between these representations:

$$C_{ij}^{(cc)} \triangleq \frac{\sum_n y_{n,i}\tilde{y}_{n,j}}{\sqrt{\sum_n (y_{n,i})^2}\sqrt{\sum_n (\tilde{y}_{n,j})^2}} \tag{3.1}$$

Here, n denotes the index of n-th frames, while i and j indicate the channel dimensional positions within the frame-level representations. C is a d-dimensional square matrix based on the size of the channel projected output, with values ranging from $[-1, 1]$.

To encourage the decorrelation of information across different channels, with the objective that each channel remains independent except for itself, we target the off-diagonals of the cross-correlation matrix to be zero and the diagonals to be one. This alignment ensures minimal redundancy among the channels. Further methodological insights will be shared in the subsequent subsection. To facilitate

this, we introduce a Cross-Correlation (CC) loss designed to optimize the matrix towards an identity matrix. This loss is calculated as follows:

$$\mathcal{L}_{cc} \triangleq \underbrace{\sum_{i}(1 - C_{ii}^{(cc)})^2}_{\text{invariance term}} + \lambda \underbrace{\sum_{i}\sum_{j \neq i} C_{ij}^{(cc)^2}}_{\text{disentangling term}} \tag{3.2}$$

Here, setting $\lambda < 1$ balances the importance of diagonal dominance versus off-diagonal minimization, effectively steering the cross-correlation towards an ideal identity matrix. This strategy enhances the model's ability to treat each channel's information as distinct and independent, reducing redundant information to make speech representations more expressive and controllable.

Note that given the sequential nature of Y and $\tilde{Y}$, disregarding frame-level correlation can lead to an overestimation of variability. To mitigate this, we consolidate the outputs' batch dimension into a single, unified dimension, simplifying the data structure for more efficient processing. Additionally, we eliminate any zero-padded frames within each minibatch, optimizing data handling and boosting computational performance. We then perform random sampling of size n from the consolidated outputs, indexing identically on both Y and $\tilde{Y}$. This approach ensures greater independence of features and enhances the stability of our framework, allowing us to fine-tune the model's performance in noisy conditions more effectively.

To further enhance the disentanglement of output representations, we introduce an additional linear projection block that mirrors the original structure and processes the bottleneck representations Z to generate the projected P_Z. This projection is used to estimate the empirical self-correlation (SC) by leveraging the computational framework outlined in Equation 3.1, applied to P_Z through random sampling, thereby computing the self-correlation between the projected outputs.

The SC loss is then computed in a manner akin to the cross-correlation (CC) loss (as described in Equation 3.2), but using the SC matrix instead (i.e., $P_z P_z'$ from the output in Figure 3.1 and shown in Equation 3.5). While the CC loss is effective, our goal is to achieve more efficient, expressive, and controllable speech representations. By disentangling the bottleneck features and using them to predict the hidden units of the original clean training audio (such as HuBERT's codes), we

guide the encoder to identify and disregard non-contextual background noise. This approach helps to suppress residual noise in the final contextual representations, enhancing the clarity and accuracy of the output.

The comprehensive optimization loss for our pre-training framework is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{HB}} + \alpha\mathcal{L}_{\text{CC}} + \beta\mathcal{L}_{\text{SC}} \tag{3.3}$$

Here, $\mathcal{L}_{\text{HB}}$, $\mathcal{L}_{\text{CC}}$, and $\mathcal{L}_{\text{SC}}$ denote the HuBERT loss, cross-correlation loss, and self-correlation loss, respectively, where the HuBERT loss is defined as

$$L = \alpha \cdot \sum_{t \in M} \log p_f(z_t|\tilde{X}, t) + (1-\alpha) \cdot \sum_{t \notin M} \log p_f(z_t|\tilde{X}, t) \tag{3.4}$$

and the self-correlation loss follows the cross-correlation loss in Equation 3.1, but it is defined with the following modifications:

$$C_{ij}^{(sc)} \triangleq \frac{\sum_n z_{n,i} z_{n,j}}{\sqrt{\sum_n (z_{n,i})^2} \sqrt{\sum_n (z_{n,j})^2}} \tag{3.5}$$

replacing $C_{ij}^{(cc)}$ with $C_{ij}^{(sc)}$. Here, z_* is the output of the transformer encoder. The parameters α and β are both set to 0.5, balancing the contributions of the respective losses to the overall training objective. This configuration ensures a robust framework for enhancing the noise robustness of the model while maintaining fidelity to the clean speech signals. Note that although the additional loss from Barlow Twins introduces a constant-order complexity—due to the fixed size of the predefined correlation matrix—the computational overhead during inference and post-deployment remains unchanged, as these optimization losses are applied only during training.

Method Insight: Optimizing cross-correlation towards identity enhances noise reduction in representations. To understand how the proposed CC loss loss facilitates noise reduction and feature invariance, we draw a comparison with the infoNCE loss [137]. Specifically, the invariance term of our CC loss (i.e., the

first term in Equation 3.2) given by:

$$\underbrace{\sum_{i}(1-\frac{\langle y_{.,i},\tilde{y}_{.,i}\rangle_n}{\|y_{.,i}\|_2\|\tilde{y}_{.,i}\|_2})^2}_{\text{proposed invariance term}} \tag{3.6}$$

mirrors the function of the positive contrastive pair found in the infoNCE formulation, as outlined in Equation 3.7.

$$\underbrace{-\sum_{n}\frac{\langle y_n,\tilde{y}_n\rangle_i}{\tau\|y_n\|_2\|\tilde{y}_n\|_2}}_{\text{infoNCE's positive contrastive}} \tag{3.7}$$

This similarity highlights the underlying principle of enhancing feature consistency across varied conditions.

To note, the primary objective of the infoNCE loss is to enhance the similarity between an anchor and a positive sample while simultaneously reducing the similarity between the anchor and any negative samples. This strategy is instrumental in shaping the feature space, thereby improving the model's ability to discern data-specific patterns or characteristics independently of labeled data.

In a similar vein, our model employs the positive contrastive loss component with the aim of maximizing the agreement on speech content between two noise-distorted embeddings. We strive to minimize variations, such as noise, by achieving perfect correlation across the channel dimensional features of the two embeddings. This approach aligns with the principles of information bottleneck theory, where the model is trained to filter out noise and extract contextual representations that remain consistent despite distortions in the inputs.

Moreover, by decorrelating the off-diagonal elements of the cross-correlation matrix, our method actively discourages unnecessary information sharing across feature components. This encourages the formation of disentangled representations, enhancing the model's ability to function effectively in noisy environments by isolating useful speech signals from irrelevant background noise.

3.4 Experiments

3.4.1 Dataset

For our experimental setup, we aligned our data environments with protocols specified in prior research [138, 139] to ensure valid performance comparisons. We utilized the comprehensive 960-hour LibriSpeech dataset for pre-training purposes. Validation was performed using the dev-clean subset of LibriSpeech, which facilitates effective model evaluation under controlled conditions.

Noise Data Configuration: The noise dataset, sourced from FreeSound [140], comprises recordings sampled at 16kHz. This dataset is bifurcated into two primary noise categories:

- Stationary Noises (Type A): This category includes consistent environmental sounds such as Car, Metro, and Traffic noises. For these noise types, our training set includes 10 unique audio streams each, while our testing set includes 8 distinct streams per noise type.
- Non-Stationary Noises (Type B): This category encompasses dynamic and variable noises such as Babble, Airport/Station, Cafe, and AC/Vacuum sounds. Each noise type in this category is represented by 10 audio streams in the training dataset and 8 in the testing dataset.

The total duration of the noise data utilized is approximately 2 hours, providing a substantial variety of acoustic environments for robust testing. For the evaluation phase, we selected 120 sub-files at random from the LibriSpeech test-clean set, adhering to the dataset's standard testing protocol. Additionally, the LibriSpeech dataset offers pre-mixed noises at diverse signal-to-noise ratios (SNRs) ranging from 0 to 20 dB, culminating in 4200 noisy test instances.

This detailed and methodically curated dataset setup ensures that our models are tested across a wide spectrum of real-world and synthetic noisy conditions, allowing us to rigorously assess their performance and robustness in recognizing speech amidst various background noises.

3.4.2 Speech Representation Model Pre-training

We conducted continual pre-training using the weights provided by the Fairseq toolkit over 250,000 steps. In our architecture, the final projection block is configured with dimensional sizes of 2048 and 4096 for the Cross-Correlation (CC) and Self-Correlation (SC) components, respectively. This diverges from the findings in [50]; we observed a concave performance curve as a function of increasing dimensionality in the projection network, which suggests diminishing returns at higher dimensions.

Moreover, we adopted a sampling size of $n = 640$ for our experiments. Our results indicate that a smaller sample size is beneficial during the early stages of learning. It introduces a slightly higher estimation error, which stimulates the network by increasing its stochastic nature, thus helping the model to avoid local minima. To mitigate potential negative effects from this increased estimation error, we adjusted the penalty parameter λ to a smaller value of 0.005.

Additionally, we found that a reduced learning rate of 7×10^{-5} optimizes the pre-training process. This smaller learning rate helps in stabilizing the training dynamics, allowing for more gradual but consistent improvements in model performance.

3.4.3 Speech Representation Model Fine-tuning

We utilized the optimal checkpoint from the pre-training phase and proceeded with the standard base setup for durations of 100 hours, 10 hours, 1 hour, and 10 minutes. The ASR fine-tuning was exclusively conducted on the HuBERT component. Furthermore, we implemented multi-condition training using noise levels ranging from 0 to 20 dB. For final evaluations, we assessed our performance using the best checkpoint as determined by the lowest validation Word Error Rate (WER).

TABLE 3.1: Experimental results for speech representation models on the task of automatic recognition of synthesized noisy speech, covering various noise types with SNRs ranging from 0 to 20 dB, without using a language model.

Models	Pre-train	Type-A Noise			Clean
		Traffic	Metro	Car	(subset)
Fine-tuning: 100 hours labeled (with corruption of FreeSound)					
DEMUCS	FreeSound	26.46	23.22	16.02	10.9
AvT	No	27.88	24.28	17.76	13.1
Wav2vec 2.0	Clean	29.22	27.44	18.24	14.0
Wav2vec 2.0	FreeSound	24.52	22.48	16.24	13.5
EW2	FreeSound	20.94	19.84	14.88	12.3
HuBERT Base	FreeSound	12.43	12.20	8.39	9.4
deHuBERT (Ours)	FreeSound	**11.66**	**11.21**	**7.62**	**8.6**
Fine-tuning: 10 hours labeled (with corruption of FreeSound)					
HuBERT Base	Clean	19.05	18.26	12.91	13.5
HuBERT Base	FreeSound	17.08	17.30	13.05	13.7
deHuBERT (Ours)	FreeSound	**16.05**	**15.74**	**11.95**	**12.8**
Fine-tuning: 1 hours labeled (with corruption of FreeSound)					
HuBERT Base	Clean	34.42	33.08	26.74	**27.8**
HuBERT Base	FreeSound	32.19	31.77	27.60	29.1
deHuBERT (Ours)	FreeSound	**31.51**	**31.24**	**26.68**	28.4
Fine-tuning: 10 mins labeled (with corruption of FreeSound)					
HuBERT Base	Clean	55.41	54.66	47.95	48.4
HuBERT Base	FreeSound	53.16	52.58	49.56	50.7
deHuBERT (Ours)	FreeSound	**49.67**	**49.71**	**45.80**	**47.1**

3.4.4 Experimental Results

We compare our results without a language model with an off-the-shelf HuBERT as the baseline to determine the efficiency of our model in learning a noise-robust ASR with limited fine-tuning data. Also, we included results from the HuBERT base model that undergoes multi-conditioning pre-training to cast a holistic analysis. Table 3.1 and 3.2 display the ASR performance in terms of WER, using a predefined subset of test-clean audio pre-mixed with individual noise types at signal-to-noise ratios (SNRs) ranging from 0 to 20 dB, as described in [138]. We observe that pre-training HuBERT with noise helps to improve the adaptability to noise on the downstream ASR, but this comes at the cost of degrading clean speech performance. Nonetheless, deHuBERT outperforms baseline HuBERT on both noisy and clean

TABLE 3.2: (*Continued..*) Experimental results for speech representation models on the task of automatic recognition of synthesized noisy speech, covering various noise types with SNRs ranging from 0 to 20 dB, without using a language model.

Models	Pre-train	Type-B Noise				Avg. (Noisy)
		Babble	Airport/ Station	AC/ Vacuum	Cafe	
Fine-tuning: 100 hours labeled (with corruption of FreeSound)						
DEMUCS	FreeSound	45.56	36.98	38.20	27.02	30.49
AvT	No	43.42	35.32	36.62	27.06	30.33
Wav2vec 2.0	Clean	47.50	39.68	38.84	31.14	33.15
Wav2vec 2.0	FreeSound	39.56	32.50	34.94	25.22	27.92
EW2	FreeSound	33.88	27.36	27.94	22.08	23.85
HuBERT Base	FreeSound	22.52	16.91	15.94	12.79	14.45
deHuBERT (Ours)	FreeSound	**21.25**	**16.02**	**14.93**	**11.94**	**13.52**
Fine-tuning: 10 hours labeled (with corruption of FreeSound)						
HuBERT Base	Clean	33.71	26.85	23.82	20.19	22.11
HuBERT Base	FreeSound	27.93	22.33	20.77	17.58	19.43
deHuBERT (Ours)	FreeSound	**26.58**	**21.23**	**20.14**	**16.83**	**18.36**
Fine-tuning: 1 hours labeled (with corruption of FreeSound)						
HuBERT Base	Clean	49.72	41.86	39.98	35.79	37.37
HuBERT Base	FreeSound	42.54	36.83	36.11	32.82	34.27
deHuBERT (Ours)	FreeSound	**41.74**	**36.27**	**35.54**	**32.41**	**33.63**
Fine-tuning: 10 mins labeled (with corruption of FreeSound)						
HuBERT Base	Clean	70.25	63.62	61.89	57.68	58.78
HuBERT Base	FreeSound	60.53	56.31	56.00	52.92	54.44
deHuBERT (Ours)	FreeSound	**58.59**	**53.82**	**53.88**	**50.66**	**51.73**

speech regardless of the pre-training condition. Additionally, the difference in performance becomes more apparent with the increasing scarcity of fine-tuning resources. Finally, we investigate the experiment with the typical 100h fine-tuning to compare our deHuBERT with existing models. On the complete test-clean and test-other set, we achieved a WER of 6.3% and 13.2%, respectively. This score is comparable to the baseline performance despite using only noisy speech for fine-tuning. Additionally, deHuBERT achieves the top WER on the noisy data.

To illustrate the noise-agnostic properties of the deHuBERT embeddings, we employed the t-SNE visualization technique on the bottleneck features from both HuBERT and deHuBERT models, as shown in Figure 3.2. These features were derived from 720 randomly selected audio samples from the train-clean-100 dataset,

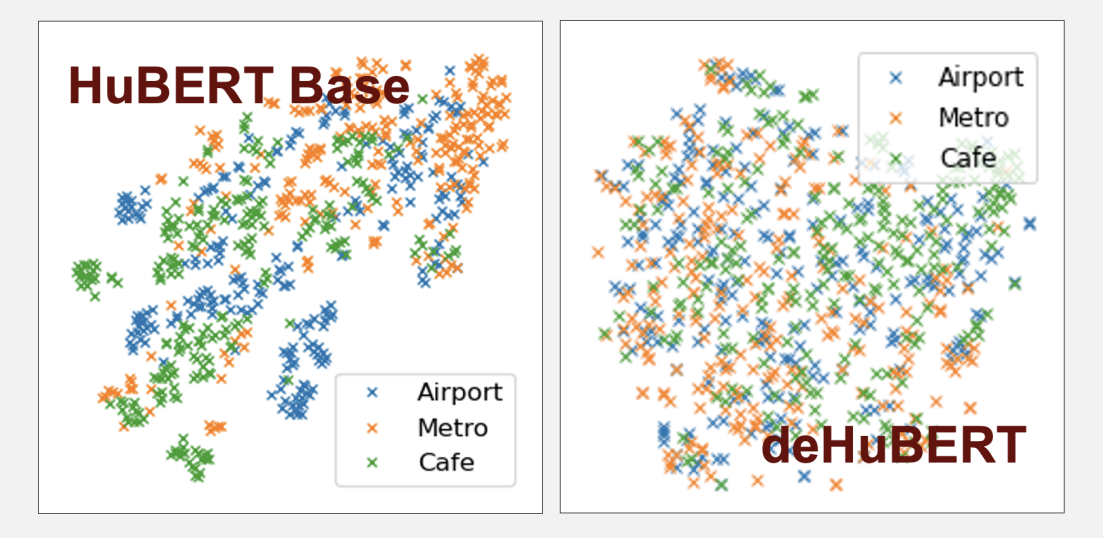


FIGURE 3.2: t-SNE plots comparing disentanglement and noise invariance across different networks exposed to 0 dB noise levels.

each mixed with 0 dB of Airport, Metro, and Cafe noises. Prior to visualization, we conducted global mean pooling on all the bottleneck features within a sequence to produce vector representations, which were then subjected to the t-SNE algorithm.

In the t-SNE plot for the HuBERT Base model (left), we can observe distinct clusters formed by samples associated with the same type of noise, indicating that the model retains noise information within its features. In stark contrast, the t-SNE plot for deHuBERT (right) shows a lack of clear clustering by noise type, suggesting that deHuBERT embeddings are effectively noise-agnostic.

3.4.5 Post-methodology Study

In this section, we conduct stress testing on our model to assess its robustness in out-of-domain (OOD) scenarios. We utilize the TEDLIUM3 dataset [141] to examine how domain shifts impact noisy ASR performance. Additionally, we incorporate out-of-domain office noise from FSD50K [142], selecting noise types such as Whispering, Writing, Typing, Typewriter, Telephone, Conversation, Laughter, Computer Keyboard, and Printer. We specifically chose files shorter than 10 minutes, resulting in a subset of 385 files. Table 3.3 details the performance after fine-tuning on a selected clean audio set (10h) across the complete test set under three different conditions:

Firstly, our pre-trained model shows comparable performance to the baseline under condition in-domain clean test set, confirming that the model remains robust and unaffected by noisy pre-training. Secondly, deHuBERT consistently outperforms the HuBERT base in handling unseen noises during fine-tuning, demonstrating superior performance in noisy environments for conditions in-domain pre-train noise with out-of-domain fine-tuning and out-of-domain pre-train noise with out-of-domain fine-tuning. Lastly, while there is a noticeable degradation in performance in both in-domain and out-of-domain noisy ASR scenarios, deHuBERT exhibits a relatively smaller increase in Word Error Rate (WER) compared to the HuBERT base, particularly under condition out-of-domain pre-train noise with out-of-domain fine-tuning. This indicates that deHuBERT is better equipped to handle variations in noise, enhancing its utility for applications in diverse and challenging acoustic environments.

TABLE 3.3: Results on various out-of-domain noisy conditions. We fine-tuned our model with 10h (respective) dataset. The rows of the table indicate training domains, while the columns represent testing domains.

Models	FT Data (10hrs)	WER (%) of testing data ↓			
		LS (Test set)		TEDLIUM	
		Clean	Other	Dev	Test
Testing set from the original data (Clean)					
HuBERT Base	LibriSpeech	**9.8**	18.2	**25.4**	**23.6**
deHuBERT (Ours)	LibriSpeech	10.1	**18.1**	25.5	23.8
HuBERT Base	TEDLIUM	**14.9**	23.8	**18.1**	**17.3**
deHuBERT (Ours)	TEDLIUM	15.2	**23.7**	18.2	17.4
Testing set with additive FreeSound noise (0–20 dB)					
HuBERT Base	LibriSpeech	20.3	36.4	35.8	36.4
deHuBERT (Ours)	LibriSpeech	**13.4**	**26.0**	**30.1**	**30.3**
HuBERT Base	TEDLIUM	23.5	38.8	26.4	27.8
deHuBERT (Ours)	TEDLIUM	**19.3**	**32.8**	**22.7**	**22.8**
Testing set with additive OOD, office noise (0–20 dB)					
HuBERT Base	LibriSpeech	26.6	44.5	42.2	43.9
deHuBERT (Ours)	LibriSpeech	**17.0**	**32.0**	**33.7**	**35.5**
HuBERT Base	TEDLIUM	30.6	46.2	34.5	35.3
deHuBERT (Ours)	TEDLIUM	**23.2**	**37.4**	**26.2**	**27.7**

3.5 Summary

In this chapter, we introduce a novel pre-training framework that enhances speech recognition robustness by effectively disentangling noise using self- and cross-correlation losses. Our model excels in managing noisy and out-of-domain ASR environments, while preserving excellent performance on clean audio tests. We further illustrate this improvement through t-SNE visualizations of the deHuBERT model's contextual representations. These plots reveal a pattern of randomly dispersed projections, clearly demonstrating the model's ability to minimize noise integration within its learned features. This significant reduction in embedded noise information not only confirms the model's enhanced noise robustness but also highlights its potential for deployment in real-world ASR applications (i.e., transcribing text from noisy speech, such as in video streams where the speaker may be affected by background noise or music), where varied and unpredictable noise conditions are common. This work not only advances the field of speech recognition but also sets a new benchmark for developing noise-resilient ASR systems.

Chapter 4

Adapting Speech Representations for Noise Robustness via Deep Filter-Tuning

4.1 Introduction

Modern advancements in speech representation learning have emphasized scaling models, such as self-supervised learning models, to foundational levels. This scaling exploits the deep and complex neural networks of transformers for improved sequential encoding. However, adapting fully-trained transformer models to new tasks or domains remains a significant challenge. This difficulty primarily arises from the need to fine-tune large-scale parameters, a process that is not only computationally demanding but also prone to overfitting, especially in scenarios with limited data resources. To address these challenges, parameter-efficient fine-tuning (PEFT) has emerged as a promising solution. This approach seeks to modify minimal parts of the transformer model while maintaining the integrity and knowledge encapsulated during the extensive pre-training phase.

In this chapter, we will delve into understanding the workings behind some of the PEFT approaches to identify their limitations for efficient noise adaptation in downstream ASR task. We then introduce a novel PEFT adapter called deep filter-tuning, akin to prompt tuning, inspired by speech extraction techniques that utilize

feature-wise linear modulation to extract contextual content from other signals. We discuss the motivations behind our work, outline the methodology, and present an ablation study to highlight the improved noise-robust adaptation of our model for more efficient and expressive speech representation learning. This chapter aims to address the following questions:

- What is the most efficient PEFT method, and how it adapts to downstream tasks, including a discussion of its limitations?
- How can we overcome the challenges to enhance the adaptability of the PEFT approach?
- How does our proposed work compare to the recent SOTA models?
- How well does our proposed work generalize to out-of-domain noises?

4.2 Motivation

Despite the remarkable achievements of foundational speech models and pre-trained SSL models, they are often criticized for their inefficiency in leveraging pre-trained knowledge. This inefficiency stems from their extensive parameter sizes, which range from millions to billions of updatable parameters. Such large-scale models frequently necessitate substantial computational resources and extensive memory for fine-tuning [143–148]. Furthermore, these models are susceptible to catastrophic forgetting, particularly when deployed on low-resource datasets. This vulnerability leads to overfitting and hampers their ability to generalize effectively across diverse acoustic environments marked by background noises, accents, and varied speaker emotions [149].

To overcome these challenges, researchers have turned to parameter-efficient fine-tuning (PEFT) methods, which minimize the number of trainable parameters within large pre-trained SSL models. These techniques typically involve freezing most of the model's weights and fine-tuning only a small subset of the network, thereby adapting the encoding speech representations for specific downstream tasks. Notable approaches include adapter tuning [4, 145, 150], which introduces

compact trainable low-rank modules within a bottleneck architecture, and prefix tuning [151] and prompt tuning [114], which prepend trainable tokens to either the training set or the intermediate features of multi-headed attention layers. Additionally, Low-Rank Adaptation (LoRA) [113] further reduces computational complexity by applying low-rank factorization to the model's weight matrices in the self-attention layer.

Among these methods, soft prompt tuning is recognized for its efficient design. It involves initializing instructional signals as trainable embeddings, which are then prepended to the sequential input features (i.e., illustrated in Figure 4.3) and processed by the frozen transformer for instructional tuning, requiring the fewest updatable parameters [114]. However, applying soft prompts effectively in speech processing poses distinct challenges compared to natural language processing (NLP). Speech signals are affected by factors like prosody, speaker emotion, and background noise, all of which manifest in continuous rather than discrete forms. These elements add complexity to the latent phonetic content crucial for ASR, complicating the development of high-quality and expressive soft prompts that can effectively interact with frozen model representations [143, 152]. Despite some initial attempts, soft-token-based tuning often underperforms relative to other PEFT methods, particularly in low-resource settings [153–155]. As such, in this chapter, we will focus on enhancing soft-token-based tuning methods for ASR in challenging environments, such as those with background noise and emotionally affected speech.

4.3 Methodology

4.3.1 Preliminaries: Soft-token Based Tuning

This section provides an overview of the foundational concepts of soft-token-based tuning, commonly used in prompt tuning. We will refer to prompt tuning to investigate how these soft tokens adapt the frozen pre-trained model, setting the stage for a deeper exploration later in the thesis.

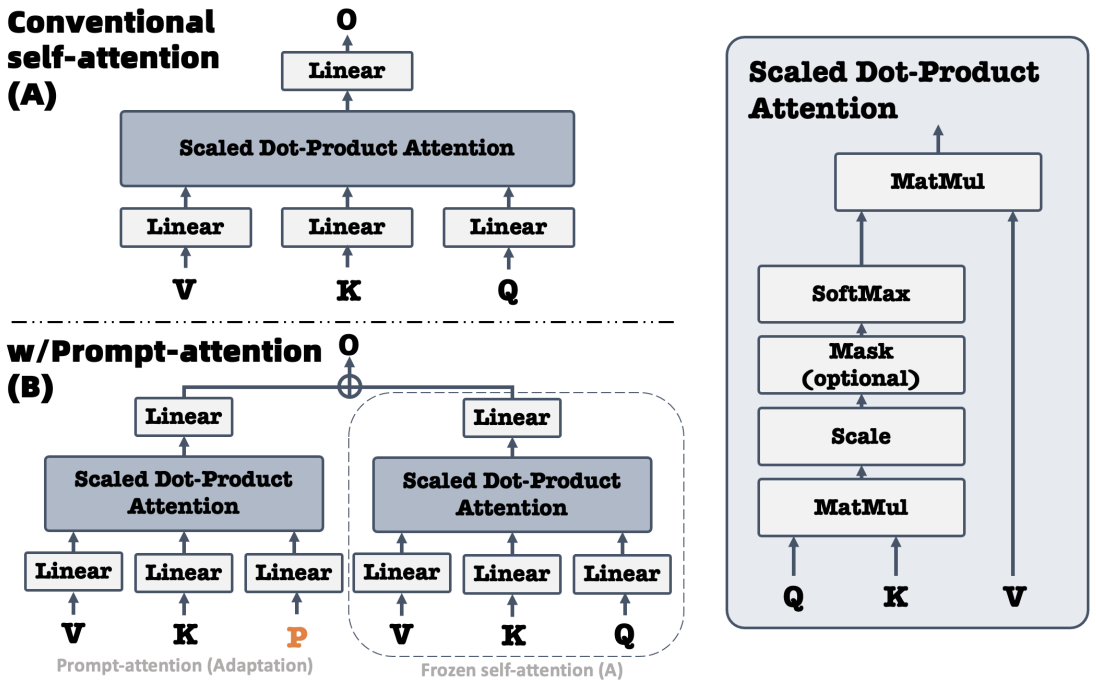


FIGURE 4.1: a comparison between conventional self-attention and self-attention with prompt tuning in transformer networks. On the above (A), the conventional self-attention architecture is shown, consisting of linear transformations applied to Q (Query), K (Key), and V (Value) components, followed by scaled dot-product attention. O (Output) represents the output of the self-attention model. At the bottom (B), the prompt-attention variant incorporates an additional prompt P input in the attention mechanism to interact with the key value component, potentially guiding the model to focus on relevant features for specific task adaptation.

To simplify the task of prompt tuning without the loss of generality, we start by considering a single-head self-attention layer,

$$\mathcal{O}_{\text{frozen}} = \varphi\Big(\frac{1}{\sqrt{d}}\mathbf{X}\mathbf{W}_{\text{Q}}\mathbf{W}_{\text{K}}^{\intercal}\mathbf{X}^{\intercal}\Big)\mathbf{X}\mathbf{W}_{V} \tag{4.1}$$

with input $X \in \mathbb{R}^{T\times d}$ consisting of T utterance frames of dimension d each. $\mathbf{W}_Q$, $\mathbf{W}_K$ and $\mathbf{W}_V$ are the frozen pre-trained weights for *query*, *key* and *value*. φ denotes the softmax nonlinearity function that acts row-wise for a $T \times T$ matrix. To incorporate trainable prompt tokens as instructional signals for PEFT on the downstream task, a series of soft embeddings, denoted as $\mathbf{P} \in \mathbb{R}^{m\times d}$, is prepended to X, resulting in the augmented matrix of $\mathbf{X_P} := [\mathbf{P}\ X] \in \mathbb{R}^{(T+m)\times d}$, serving as the latent input to the transformer blocks. The output of the attention-layer, as

introduced in [54, 114, 151] is thus in the form

$$\mathcal{O} = \varphi\Big(\frac{1}{\sqrt{d}}\mathbf{X_P}\mathbf{W}_{\mathrm{Q}}\mathbf{W}_{\mathrm{K}}^{\intercal}\mathbf{X}^{\intercal}\Big)\mathbf{X}\mathbf{W}_V \tag{4.2}$$

Note that this is slightly different from (4.1) in that now the layer computes a cross-attention between the augmented inputs $\mathbf{X_P}$ and the original inputs X. We can rearrange this to

$$\begin{aligned}\mathcal{O} &= \varphi\Big(\frac{1}{\sqrt{d}}[\mathbf{P}\ \mathbf{X}]\mathbf{W}_{\mathrm{Q}}\mathbf{W}_{\mathrm{K}}^{\intercal}\mathbf{X}^{\intercal}\Big)\mathbf{X}\mathbf{W}_V \\ &= \underbrace{\varphi\Big(\frac{1}{\sqrt{d}}\mathbf{P}\mathbf{W}_{\mathrm{Q}}\mathbf{W}_{\mathrm{K}}^{\intercal}\mathbf{X}^{\intercal}\Big)\mathbf{X}\mathbf{W}_V}_{\text{prompt-attention, } \mathcal{O}_{\text{prompt}}} + \underbrace{\varphi\Big(\frac{1}{\sqrt{d}}\mathbf{X}\mathbf{W}_{\mathrm{Q}}\mathbf{W}_{\mathrm{K}}^{\intercal}\mathbf{X}^{\intercal}\Big)\mathbf{X}\mathbf{W}_V}_{\text{frozen self-attention, } \mathcal{O}_{\text{frozen}}}\end{aligned} \tag{4.3}$$

Here, we observe that the fine-tuning and adaptation of the intermediate representations stem from the additive impact of prompt-attention on the frozen features (i.e., the dot product between projected $\mathbf{P}$ and X), as illustrated in Figure 4.1.

Therefore, to achieve optimal performance, it is crucial to accurately align instructional signals so that they follow the distribution within the vector space of the frozen features, ensuring effective communication. This precise alignment targets and mitigates complex noise distortions and variations in speaker attributes effectively. In simpler terms, the system learns to interpret and respond to instructions based on its pre-trained knowledge.

It is important to note that these instructional signals $\mathbf{P}$ are trained from scratch to operate within the embedding space of frozen features. The effectiveness of these guiding signals depends on the expressivity of the frozen speech representations. While pre-trained models are assumed to be well-trained and broadly generalized, the reality is that the training datasets for these models are often limited, particularly in their handling of noise. Consequently, this approach is constrained by the limited diversity of training datasets, which fail to adequately represent various domains and noise distortions, especially under low-resource conditions. As a result, the model's ability to adapt based on signal guidance may be compromised, leading to suboptimal performance since effective guidance and prompt attention are not guaranteed. Therefore, we aim to investigate the impact on ASR performance by modifying the functionality of the soft tokens (also known as the prompts in

prompt-tuning). Specifically, we propose reducing reliance on prompt attention conditions for adaptation tuning and instead using them as feature-modulating filters. Details of this approach are provided in the following subsection.

4.3.2 Deep Filter Tuning (DFT)

To address the challenge of insufficient a priori knowledge—which hinders the effective learning of soft tokens during fine-tuning—we propose a targeted use of soft tokens to enhance noisy speech recognition. As shown in Figure 4.2, our method draws on the dual-process theory of cognition, which distinguishes between fast and slow thinking in human decision-making [156]. We derive a static bias from soft tokens and utilize it in a manner similar to the Feature-wise Linear Modulation (FiLM) mechanism commonly applied in speaker extraction [157, 158], to scale intermediate representations and suppress noise. This mechanism is conceptually aligned with fast thinking, wherein prior biases shape immediate perceptions to filter out learned noise. Our design functions as a fast inference pathway by employing domain-informed, fixed heuristics to efficiently suppress irrelevant signals. Concretely, the soft tokens produce a filtering mask that remains fixed during inference, enabling selective attenuation of noise within frozen feature representations

In Figure 4.2, we initialize a set of m soft tokens $\mathbf{S} \in \mathbb{R}^{m \times d}$ that act as static filtering tokens in handling noisy latent speech features. Each soft token functions as a slightly different masking filter, which is broadcast to match the sequential length T of the frozen representations. To weigh the importance of each masking token on every frame, we introduce a frame-level linear module that computes the temporal weights of the token conditional on the input, denoted as $W = \delta(\mathbf{X}\mathbf{W}_s)^\intercal \in \mathbb{R}^{m \times T}$, where $\mathbf{W}_s \in \mathbb{R}^{d \times m}$, and δ represents tanh. The weighted-static filter is calculated as $W^\intercal S \in \mathbb{R}^{T \times d}$, which serves as the scaling mask. Soft tokens, which are trainable parameters, are optimized through the noise adaptation objective of ASR. This forces them to learn a masking pattern that extracts speech content from noisy signals, mirroring the scaling factor γ in FiLM of speaker extraction.

Then, for "slow" thinking, we have introduced a branch specifically designed for slow cognitive assessment, which features a bottleneck linear architecture to facilitate rapid assimilation of sampled utterances. This process is deliberate and

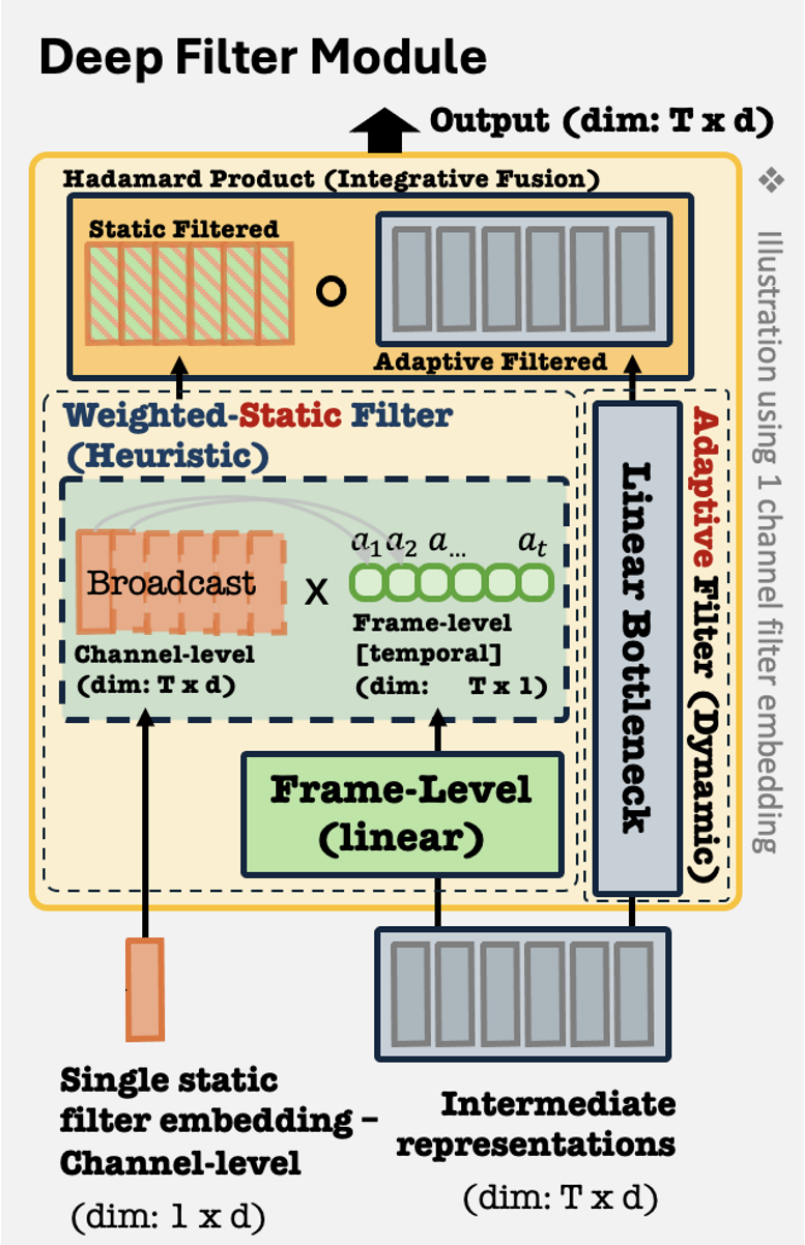


FIGURE 4.2: An illustration of the Deep Filter Tuning module operating on a single functional static filter embedding (initialized from external soft-tokens). The static filter embedding of channel-level is broadcast (i.e., repeating the token to match the size of T frames) and multiplied by the temporal weight to determine its influence on the changes in speech variations over time. The resultant weighted static filters are then fused with the adaptive filter through a Hadamard product (indicated by $\circ$). The output is the result of the modulated latent speech features, with a dimensional shape of $T \times d$. Such DFT is inserted as an adapting module, as demonstrated in Figure 4.3.

incorporates conscious (dynamic) analysis. The modulated output is then represented by the Hadamard product of the heuristic bias and the bottleneck-adapted

representations. As a whole, the system emulates human cognitive biases by acting as an adaptive information bottleneck—selectively passing relevant information in a manner consistent with both "fast" and "slow" thinking principles.

Besides, we emphasize the distinction between DFT and methods like LLama-Adapter [159]. LLama-Adapter treats soft tokens or adapter layers as instruction vectors inserted into frozen LLMs, modulating attention outputs through residual injection for task adaptation—primarily in discrete language tasks. In contrast, DFT reinterprets soft tokens as domain-informed filtering masks, explicitly trained to suppress noise in continuous speech representations. This filtering-centric architecture enables DFT to address the temporal and spectral complexity of speech—something LLama-Adapter does not explicitly model.

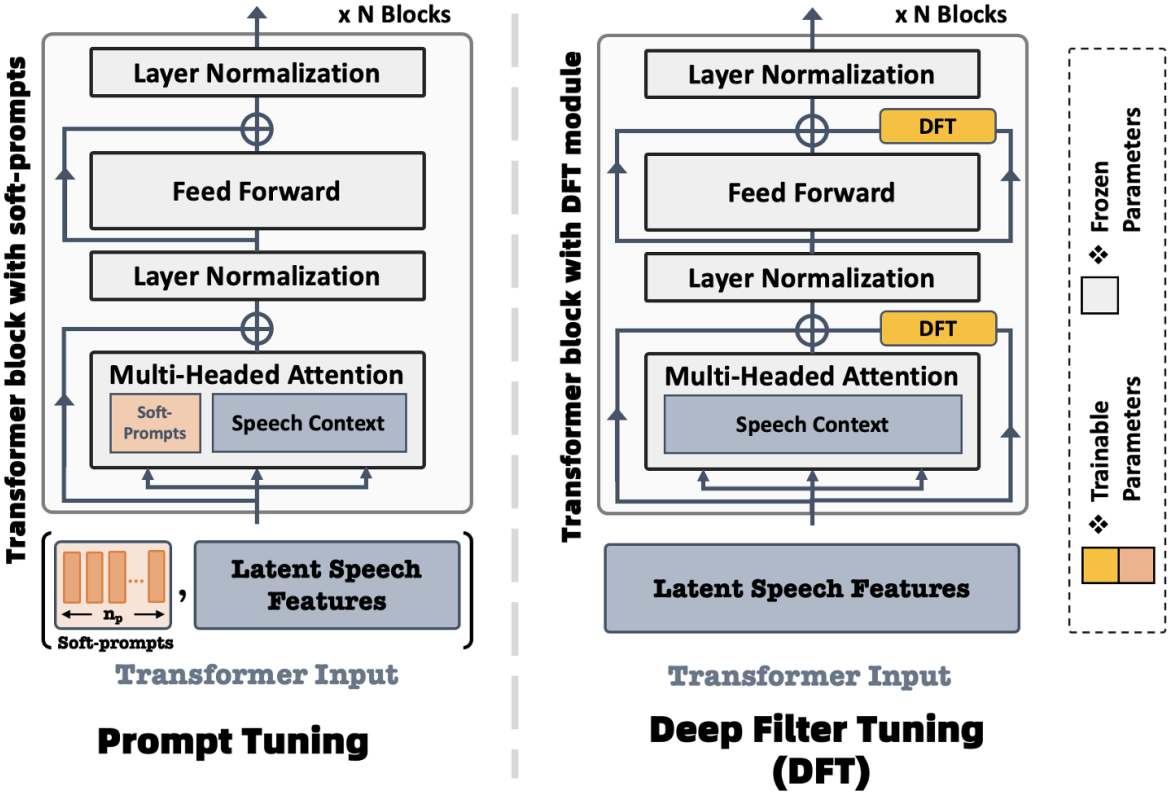


FIGURE 4.3: An illustration compares prompt tuning with Deep Filter Tuning (DFT) in transformer networks. DFT adopts the approach of Houlsby [4], incorporating a shared Deep Filtering block that provides residual adaptation tuning to the latent speech features.

Method Insight: How does Deep Filter Tuning enhances adaptation in SSL Models?

We provide more theoretical perspectives on how Deep Filter Tuning enhances adaptation in SSL models, as follows:

Information Theory Perspective: Deep Filter Tuning (DFT) can be understood through the lens of information theory. The frozen representations in SSL models contain both task-relevant and task-irrelevant information. DFT functions as an adaptive information bottleneck, selectively allowing relevant information to pass through while filtering out noise and irrelevant features, thereby enhancing the quality of the representations used for ASR [160].

Manifold Learning: SSL models learn a complex, high-dimensional manifold of speech representations. DFT refines this process by learning a task-specific submanifold that is better aligned with the ASR task. The static filters in DFT effectively project the SSL representations onto this submanifold, reducing the dimensionality of the problem space and making the model more efficient in handling the task-specific requirements [161].

Adaptive Noise Cancellation: The static filters in DFT can be likened to adaptive filters in signal processing. These filters dynamically learn to identify and suppress noise patterns within the latent space, akin to adaptive noise cancellation in the time domain. This adaptive filtering ensures that the model focuses on the most relevant features while minimizing the impact of noise and distortion.

Regularization Effect: DFT's filtering mechanism also serves as a form of regularization. By controlling the flow of information, DFT prevents the model from overfitting to task-irrelevant features within the SSL representations. This regularization is particularly beneficial in low-resource scenarios, where the risk of overfitting is higher due to limited data [162].

4.4 Experiments

4.4.1 Architectural Setup for DFT

The Deep Filter Tuning (DFT) module is a lightweight plug-in filtering block designed to work alongside the main architecture, distinct from the self-attention mechanism in both purpose and operation. To specifically evaluate DFT's effectiveness as an advanced soft-token-based tuning method, we initially exclude prompt attention from our experiments. This allows us to observe DFT's impact in isolation. We insert the DFT module in parallel with the main layers, adjacent to the multi-headed attention and feed-forward layers, following the architectural style outlined in [4]. This setup enables residual adaptation of the frozen intermediate representations. Importantly, the DFT layer is shared between the two parallel insertions within the same block, which minimizes the introduction of additional parameters and ensures efficient use of computational resources.

In a subsequent approach, we integrate DFT with the vanilla prompt attention mechanism, a combination we term DFT+. This hybrid method leverages the strengths of both approaches to improve ASR performance through prompt-based tuning. While attention mechanisms may struggle to provide precise instructional signals for filtering out irrelevant information in highly noisy or complex continuous input data, DFT lacks the capability to interact directly with the frame-level representations of the input context. By merging these two approaches, we aim to overcome their individual limitations. In this case, the filters act as instruction at the prompt attention stage, guiding the interactive refinement of the frozen features. This results in a more robust and adaptable PEFT strategy for various tasks while preserving network efficiency.

4.4.2 Pre-trained SSL Backbone: HuBERT and WavLM

For the PEFT applied to ASR, we mainly use HuBERT (Base), WavLM+ (Base) and WavLM (Large) as the pre-trained SSL backbone. WavLM+ represents the latest advancements in speech encoders, building on prior work, particularly HuBERT (based on the Transformer architecture [3]), with the addition of gated

relative positioning bias. It utilizes K-Means clustering to derive self-labels from continuous clean speech and incorporates a masked speech denoising framework along with noise conditioning during model training. This enables WavLM+ to effectively manage noise and signal distortion, gaining additional non-ASR-related knowledge essential for tasks such as diarization and separation, alongside traditional downstream contextual tasks.

With a model parameter size of 94.7 million, WavLM+ (Base) scales its training from 960 hours of unlabeled LibriSpeech to 94,000 hours of diverse compiled corpora [163–165], significantly enhancing the model's robustness and generalization capabilities across a wide array of speech-related tasks with a total of 1M steps updates.

Additionally, the WavLM (Large) model builds upon the base model with an even larger architecture and extended training regimen. It is trained with 700,000 steps, allowing for deeper learning and improved performance, particularly in more complex and noisy environments. This extensive training further bolsters the model's ability to generalize across various domains, making it a powerful backbone for ASR tasks, especially when combined with PEFT methods like DFT and DFT+.

We aimed to compare the effectiveness of Deep Filter Tuning (DFT) and prompt attention (i.e., prompt tuning) in the downstream task of automatic speech recognition (ASR) to assess their applicability and transferability from prior work in natural language processing (NLP) to real-world ASR scenarios. To simulate more realistic speech environments, we immersed our speech data in a broader domain setting, incorporating various background noises and speaker emotions. This involved collecting training datasets from LibriSpeech (LS) [166], the Emotional Speech Database (ESD) [167], and FreeSound [138, 140]. By doing so, we aimed to create a diverse training environment that closely mirrors real-world conditions, enabling our models to effectively handle background noise and emotional variations in human speech. The training dataset consists of a 100-hour subset from LS combined with the full 10-hour ESD for our experiments. To create a noisy corpus, we corrupted the speech data with the FreeSound noise dataset, which includes both stationary (Type A) and non-stationary (Type B) noises. Type A noises consist of car, metro, and traffic sounds, while Type B includes babble, airport/station, café,

and AC/vacuum noises. Each noise type comprises 10 audio streams for training and 8 for testing, totaling approximately 2 hours of noise data.

During testing, we evaluated performance across specific environments categorized into clean, noise, and emotional domains. Clean testing was conducted on the official LS testing set. For noisy ASR testing, we used the noise pre-mixed testing set [138], selecting 120 sub-files from the test-clean set of LibriSpeech and corrupting them with test noise at various signal-to-noise ratios (SNRs) ranging from 0 to 20 dB, resulting in 4,200 instances of noisy test data. Emotional domain testing utilized the provided testing set, consisting of five distinct categories: Neutral, Angry, Happy, Sad, and Surprise.

Additionally, we benchmarked the performance of prompt-based tuning against other commonly adopted PEFT methods to provide a more comprehensive understanding. These methods include Adapter-Tuning, LoRA Tuning, and fully frozen network tuning. Our implementation setup closely follows the approach outlined in [143] for PEFT optimization. In Adapter-Tuning, we use a reduction factor of 4 with one adapter module at the feed-forward layer, following Houlsby [4]. For Prompt-Tuning, we prepend 300 trainable prompts (as presented in Figure 4.3) to the input utterance—both setups matching the trainable size of DFT for comparable complexity. In the case of DFT, we employ 10 trainable tokens with the filtering modules. We use an RNN decoder [168], specifically from [169], to decode the features from the encoder output. No language model (LM) is employed in any of the experiments.

4.4.3 Experimental Results

In Table 4.1, we outline the ASR performance on the official clean LibriSpeech evaluation set using various PEFT approaches. It's noteworthy that the full model tuning of WavLM+ achieved slightly lower performance compared to the reported scores in [169]. This discrepancy may arise from the broader environmental domain present in the dataset, including a mixture of noisy and emotional speakers. Consequently, the model may struggle to learn effective speech representations for ASR in the presence of such noise. Moreover, we observe that nearly all PEFT approaches outperform the frozen model while updating only a small subset of

TABLE 4.1: WER on the official LibriSpeech evaluation set with 100 hours of train LS and ESD training data with FreeSound noise. The table shows the word error rate, WER (%) (↓) of the ASR system on clean speech recognition. Note that the first row in each block represents fully fine-tuned results.

Models	Params	Dev		Test	
	(M)	Clean	Other	Clean	Other
HuBERT (Base)	94.70	5.87	13.13	5.92	12.95
Frozen Model	0	6.21	17.11	6.52	17.29
Adapter Tuning	3.54	4.97	9.83	4.52	9.80
LoRA (R=16)	0.59	6.13	16.88	6.48	17.15
Prompt Tuning	0.23	5.87	15.85	6.20	15.97
DFT (Ours)	0.36	5.10	12.26	5.34	12.13
DFT+ (Ours)	0.36	4.95	11.83	5.15	11.78
WavLM+ (Full)	94.70	4.72	11.18	4.77	10.91
Frozen Model	0	5.37	12.16	5.40	12.04
Adapter Tuning	3.54	4.47	9.83	4.52	9.80
LoRA (R=16)	0.59	5.51	12.28	5.56	12.27
Prompt Tuning	0.23	4.83	11.51	5.08	11.44
DFT (Ours)	0.36	4.46	9.88	4.56	9.73
DFT+ (Ours)	0.36	4.38	9.82	4.45	9.61
WavLM (Large)	315	2.78	5.57	2.83	5.63
Frozen Model	0	3.51	6.93	3.54	6.89
Adapter Tuning	12.6	2.75	5.54	2.77	5.52
LoRA (R=16)	2.10	3.68	7.21	3.71	7.12
Prompt Tuning	0.31	3.14	6.38	3.12	6.43
DFT (Ours)	1.90	2.79	5.59	2.73	5.56
DFT+ (Ours)	1.90	2.75	5.50	2.74	5.54

model parameters. Particularly noteworthy is DFT among the prompt-based tuning variants, which stands out by delivering significantly superior results compared to the vanilla approach, all while maintaining the efficiency of previous methods by utilizing just 0.38% of the full model's weights. This empirically highlights the challenge (as discussed earlier) of replicating the success from the NLP domain to ASR, given the significant differences in context and network representations. However, we present a simple yet impactful workaround solution using DFT to regulate information flow, resulting in an average 11.7% relative reduction in Word Error Rate (WER). Additionally, DFT exhibits competitive performance compared to

Adapter Tuning while requiring almost 10 times fewer trainable parameters. Besides that, we achieve the best results with DFT+, which incorporates prompt attention from vanilla prompt tuning to mitigate limitations arising from the two computations, without introducing significantly more additional parameters.

Similarly, this trend is consistent across other SSL-based speech encoders, particularly HuBERT (Base) and the largest SSL of WavLM (Large), where empirical challenges arise in matching the effectiveness of other PEFT methods, as seen in NLP, for speech ASR. We also observe that soft-token-based tuning methods (e.g., prompt tuning, prefix tuning) are difficult to optimize, with performance fluctuating non-monotonically as the number of trainable parameters increases, confirming similar findings in the original studies and as discussed in [113].

Table 4.2, 4.3 illustrate the recognition performance in a noisy environment, where we assess our trained PEFT architectures using a synthetic in-domain noisy corpus with noise levels ranging from 0 to 20dB. Our proposed DFT consistently outperforms the prompt tuning, reducing the WER by 13.7% under noisy conditions on the base WavLM+. This outcome underscores the critical role of the filtering units in regulating information flow and enhancing content representations generated by the soft-token filters, thereby effectively adapting latent representations to various noisy environments. Additionally, we observed an unexpected decline in the performance of LoRA, evidenced by an increase in the error rate compared to the frozen model. We hypothesize that the assumption underlying LoRA indicating that fine-tuned model weights remain low-rank may not hold true in this context. This discrepancy could stem from the continuous nature of speech representations and the cross-modality involved in the ASR task, which requires transitioning from processing wave signals to generating discrete text output–a process not learned during upstream pre-training. These factors may lead to fine-tuned parameter weights with higher ranks. However, further investigation into this matter was not conducted as it falls outside the scope of our study.

Furthermore, we performed out-of-domain testing on the CHiME-4 [170] real noisy ASR dataset to evaluate the generalizability of our method. While fine-tuning the full WavLM+ model yielded strong in-domain noise performance, it struggled to generalize to out-of-domain cases, unlike our DFT approach. This highlights a potential pitfall of fine-tuning the full model, as it is prone to catastrophic forgetting

TABLE 4.2: WER on the synthesized noisy in-domain LibriSpeech (FreeSound) and real noisy speech CHiME-4 (OOD) testing with 100 hours of train LS and ESD noisy training data. The table shows the word error rate, WER (%) (↓) of the ASR system on noisy speech recognition at SNRs of (0 - 20)dB for the synthesized noisy test set. Note that the first row in each block represents fully fine-tuned results.

Models	Params (M)	Stationary (Type-A) Noise			CHiME 4 (Real)
		Traffic	Metro	Car	
HuBERT (Full Base)	94.70	13.58	13.22	8.83	31.22
Frozen	0	23.15	20.83	11.53	48.56
Adapter	3.54	15.19	15.87	8.17	38.43
LoRA (R=16)	0.59	24.51	23.17	11.14	54.08
Prompt Tuning	0.23	20.64	19.45	9.62	44.38
DFT (Ours)	0.36	15.04	14.85	7.80	38.62
DFT+ (Ours)	0.36	14.52	13.97	7.23	37.83
WavLM+ (Full)	94.70	9.01	8.64	6.01	20.58
Frozen	0	12.63	11.90	7.38	26.77
Adapter	3.54	8.73	8.88	6.04	19.46
LoRA (R=16)	0.59	14.28	14.04	7.86	30.98
Prompt Tuning	0.23	10.17	9.49	6.62	22.57
DFT (Ours)	0.36	8.74	8.72	5.94	19.67
DFT+ (Ours)	0.36	8.65	8.61	5.86	19.49
WavLM (Large)	315	5.28	4.96	3.65	12.13
Frozen	0	6.94	6.83	4.28	16.24
Adapter	12.6	5.23	5.03	3.71	11.59
LoRA (R=16)	2.10	8.77	8.42	6.53	20.87
Prompt Tuning	0.31	5.90	5.39	3.82	13.07
DFT (Ours)	1.90	5.14	4.81	3.49	11.87
DFT+ (Ours)	1.90	5.07	4.59	3.48	11.56

and overfitting, particularly in low-resource setups—a common challenge when using SSL models. In contrast, DFT+ outperformed all other PEFT methods, demonstrating the highest efficacy. Consistent trends were also observed in other SSL models, as reported in Table 4.2, 4.3.

Table 4.4 presents the recognition performance on emotional speech, showing results for the five different emotions included in the training and testing of the ESD dataset. Overall, all models struggle with recognizing emotional speech, particularly when the speaker expresses surprise. This difficulty may arise from occasional

TABLE 4.3: (*Continued..*) WER on the synthesized noisy in-domain LibriSpeech (FreeSound) and real noisy speech CHiME-4 (OOD) testing with 100 hours of train LS and ESD noisy training data. The table shows the word error rate, WER (%) (↓) of the ASR system on noisy speech recognition at SNRs of (0 - 20)dB for the synthesized noisy test set. Note that the first row in each block represents fully fine-tuned results.

Models	Params (M)	Non-stationary (Type-B) Noise				Avg. (Noisy)
		Babble	Airport/ Station	AC/ Vacuum	Cafe	
HuBERT (Full Base)	94.70	25.87	19.93	17.78	14.17	16.20
Frozen	0	49.98	39.32	31.98	26.49	29.04
Adapter	3.54	33.74	25.23	23.57	16.58	19.76
LoRA (R=16)	0.59	54.43	42.94	36.88	28.87	31.71
Prompt Tuning	0.23	47.51	37.45	30.32	24.33	27.05
DFT (Ours)	0.36	34.07	25.42	22.77	14.91	19.27
DFT+ (Ours)	0.36	33.28	23.89	22.05	13.88	18.40
WavLM+ (Full)	94.70	15.13	11.35	11.29	8.71	10.02
Frozen	0	26.65	18.39	16.70	13.27	15.27
Adapter	3.54	16.00	11.60	11.80	8.95	10.29
LoRA (R=16)	0.59	35.71	26.17	21.59	15.58	19.32
Prompt Tuning	0.23	18.32	13.68	13.26	10.01	11.65
DFT (Ours)	0.36	15.70	11.28	11.33	8.65	10.05
DFT+ (Ours)	0.36	15.39	11.13	11.19	8.46	9.90
WavLM (Large)	315	7.97	6.38	6.30	5.01	5.65
Frozen	0	13.72	10.98	9.21	7.14	8.44
Adapter	12.6	8.51	6.55	6.47	5.12	5.80
LoRA (R=16)	2.10	16.59	12.68	11.81	8.83	10.52
Prompt Tuning	0.31	9.32	7.18	7.20	5.82	6.38
DFT (Ours)	1.90	8.70	6.41	6.38	5.05	5.71
DFT+ (Ours)	1.90	8.51	6.28	6.07	4.99	5.57

spikes in timbral energy levels, leading to a loss of generality, especially with this specific emotion. Additionally, we observe that prompt tuning performs worse than the fully frozen network. It is likely that the learned prompts struggle with the emotional aspect, possibly due to an imbalance in the emotional data compared to neutral LS speech. The network may prioritize other factors over emotions and might be unable to cue effective instructional signals to direct prompt attention for adaptation, as described in the pitfalls of low-resource training setups in the preliminaries subsection, unlike DFT, resulting in less robust representations.

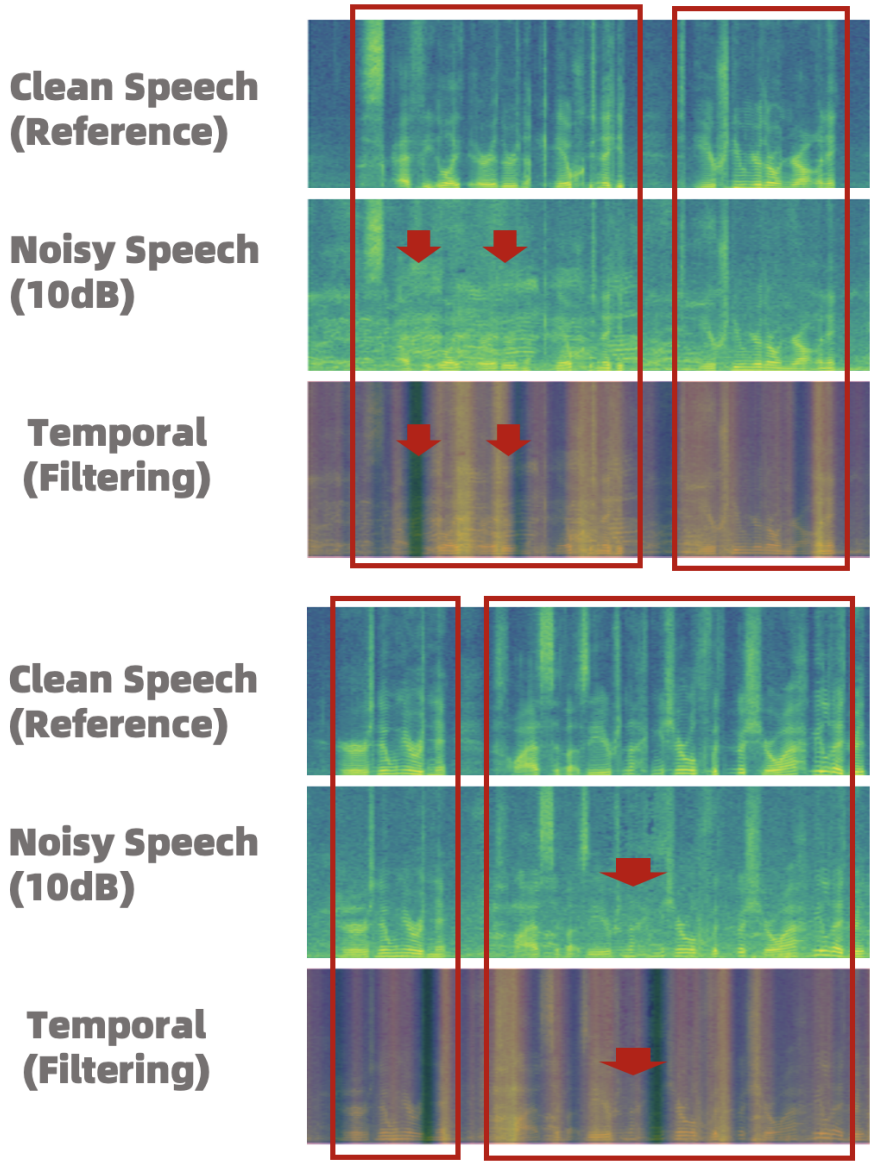


FIGURE 4.4: An illustration of the mean activation weight derived from the fram-level temporal weights of static filters presented with the heat map overlay on the noisy spectrogram. Clean speech is randomly corrupted by 10dB of babble noise.

To visually assess the filtering operation in DFT, we present a heatmap of static filter weights overlaid on the spectrogram of a randomly selected speech utterance from the Test Clean (LS) set (Figure 4.4). Background noise (specifically, 10 dB of babble, simulating another person speaking in the background) was introduced into the speech. We then extracted the mean activation weight of the static filters from DFT across all layers. The channel-wise filter map is excluded to avoid confusion, as it represents high-dimensional latent space information that is not intuitive for

TABLE 4.4: WER on the ESD testing set with 100 hours of train LS and ESD training data with FreeSound noise. The table shows the word error rate, WER (%) (↓) of the ASR system on emotional speech recognition.

Models	Neutral	Angry	Happy	Sad	Surprise
WavLM+ (Full)	9.63	11.57	12.19	11.20	14.28
Frozen	10.53	12.47	13.90	13.05	15.37
Adapter Tuning	9.62	11.23	11.42	11.37	13.18
LoRA (R=16)	10.97	13.12	14.18	13.38	15.97
Prompt Tuning	10.79	12.43	13.41	13.30	16.15
DFT (Ours)	10.01	10.51	11.60	11.18	13.02
DFT+ (Ours)	9.92	10.43	11.53	11.07	12.98

interpreting channel activity.

In the heatmap, segments containing speech content are highlighted with boxes, while regions with significant noise distortion are manually indicated by red arrows to aid quick reference. Darker shades represent the negative end, bounded by -1, while brighter shades indicate the positive end, limited to 1 of the tanh weights. Both extremes reflect increased filter processing activity, as opposed to importance weight of 0. The figure suggests that the static filter weights effectively discern content from noise, adjusting appropriately across different frames.

Our observations indicate that DFT effectively detects contextual speech content, as shown by the positive activation in the heatmap when compared to a clean reference spectrogram. In contrast, regions with significant noise distortion exhibit a more negative response, reflecting reduced content processing. This visualization underscores DFT's ability at the temporal level to accurately identify active regions within speech utterances and utilize the learned static filter embeddings to adapt frozen representations. By optimizing the flow of instructional information, DFT enhances the robustness of speech recognition. Similarly, we anticipate that the soft-token embeddings for static filters will selectively refine information, contributing to the significant performance gains observed compared to vanilla prompt tuning.

Additionally, Figure 4.5 presents a t-SNE plot [171] of encoded representations from the self-supervised transformer's output, illustrating the distribution of latent representations post-ASR fine-tuning. We compare the t-SNE distributions of

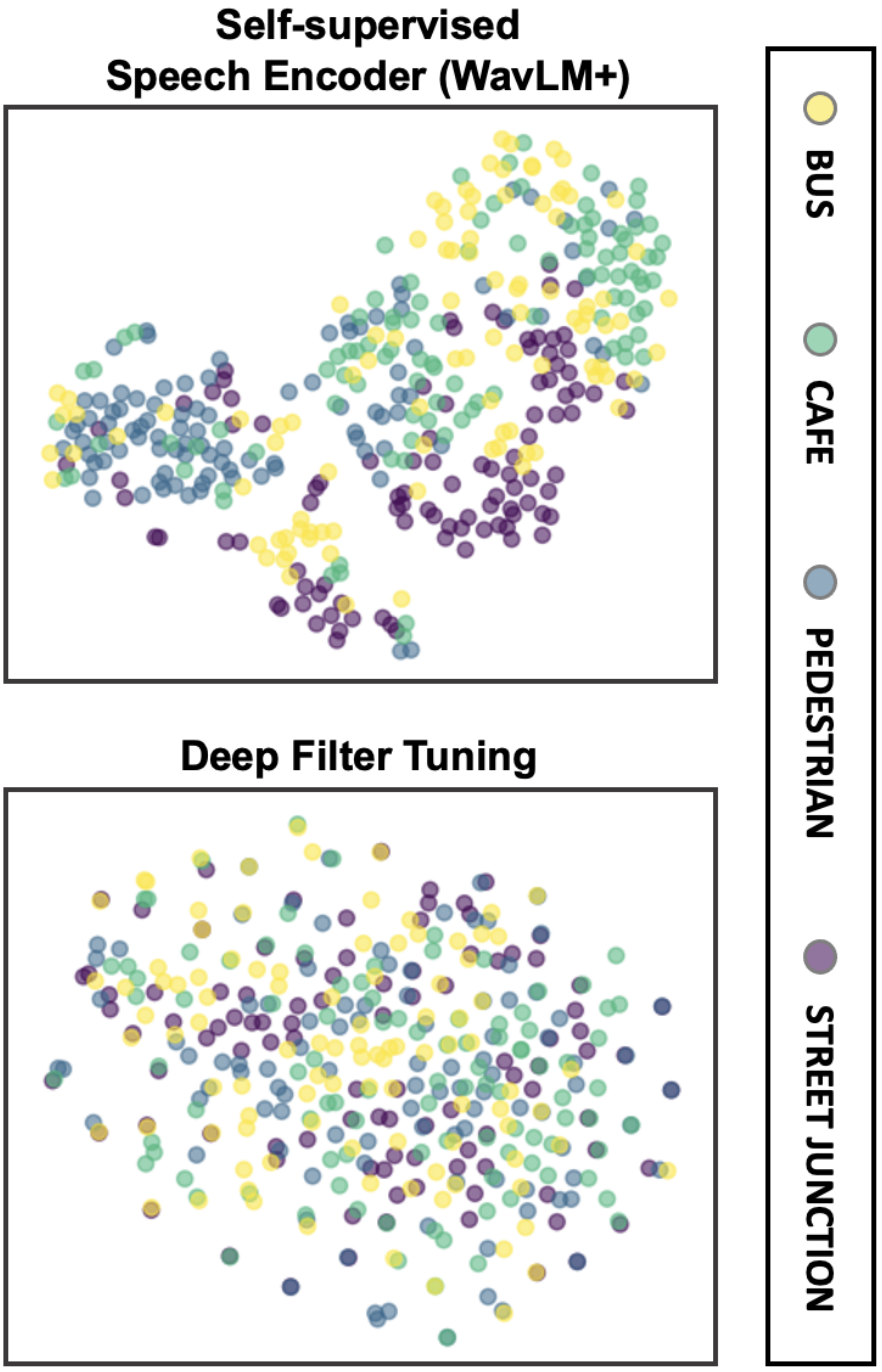


FIGURE 4.5: A t-SNE plot of the output transformer representations for out-of-domain CHiME4 Real Speech data. A total of 100 speech utterances are randomly sampled from each category and average pooled to obtain vector representations for the t-SNE distribution plot.

the vanilla WavLM+ model and our proposed DFT approach using out-of-domain CHiME 4 speech data, demonstrating DFT's robustness in handling real-world,

noisy, and unseen data. The latent representations are extracted from the transformer's output, with average pooling applied to obtain vector representations for t-SNE modeling.

The plot reveals that both models show less distinct clusters of noise types, reflecting WavLM+'s inherent resilience to noise [172], as evident in its improved ASR performance and lower WER compared to HuBERT. However, DFT further enhances the frozen representations of WavLM+ by modulating and effectively mitigating noise-induced distortions, thereby improving content representation. This is illustrated in the t-SNE plot, where the DFT model exhibits a more spread and uniform distribution with even less pronounced noise clusters, highlighting the effectiveness of deep filter tuning.

4.4.4 Ablation Studies of Deep Filter-Tuning

In this subsection, we conduct an ablation study to deconstruct the structure of DFT and examine the impact of its components on downstream performance. We will focus on evaluating the performance on real speech data from CHiME4 (OOD) and LibriSpeech Clean to assess real-world applicability in WavLM+.

Layer Non-sharing and ASR Performance

TABLE 4.5: WER of using non-sharing parameters and a single filter block in each transformer layer. MHA and FFN residual refer to DFTs added to specified layer.

Sub-models	Clean (LS)	Other (LS)	CHiME4
Sharing (Default)	4.56	9.73	19.67
Non-sharing	4.54	9.56	18.78
MHA Residual	4.73	9.96	19.98
FFN Residual	4.67	9.98	19.93

To optimize the efficiency of the proposed DFT, we implement module sharing within the same transformer block, specifically at both the multi-headed attention and feedforward layers, to facilitate residual adaptation. Table 4.5 indicates a general decrease in word error rate when using non-shared modules, despite the increase in trainable parameters. The advantages are particularly pronounced

for noisy and corrupted speech, showing a relative gain of 4.5% on the CHiME4 dataset. However, the results for clean speech are less conclusive, warranting further investigation. This suggests that noisy and corrupted speech may necessitate more information gating from the network. Consequently, with non-shared modules, each layer becomes more specialized, resulting in improved decision-making and more significant enhancements, especially in such scenarios.

In addition, we conducted experiments involving single residual adaptation. It is noteworthy that single residual adaptation at the multi-headed attention closely resembles prompt tuning. As shown in Table 4.5, firstly, we observed a slight increase in the error rate, likely caused by the absence of its counterpart in the default setup, although the change in performance appears to be marginal. Nonetheless, single adaptation still outperforms the conventional prompt attention, achieving demonstrably better recognition performance. Second, it remains inconclusive whether residual adaptation at the multi-headed attention is superior to the feedforward approach or vice versa, as their performances exhibit close differences.

Impact of the deep filtrs between Static *(Fast)* and Adaptive *(Slow)* filtering

TABLE 4.6: WER of experiments using only the sub-filtering module of deep filter tuning.

Sub-models	Clean (LS)	Other (LS)	CHiME4
DFT (Default)	4.56	9.73	19.67
Adaptive (Slow-Conscious)	4.73	10.47	21.13
Static (Fast-Heuristic)	4.85	10.12	20.01

For its intended aforementioned function, deep filters apply a fast and slow adaptive filters, regulating essential information to enhance the robustness of speech recognition. Here, we aim to independently study the importance of the two and evaluate their performance. From Table 4.6, we observe that the static filters seem to be just effective alone in performing robust domain adaptation, resulting in better calibration of frozen speech features with less degradation compared to using adaptive linear projection that resembles slow-conscious. This has demonstrated the efficacy of the approach by employing a filtering mechanism that has been well-motivated. Nevertheless, the decline in performance, as evidenced by the removal

of adaptive linear projection, underscores the significance of adapting latent speech features to interact effectively with the static filtering operation Furthermore, the performance on the test-clean dataset appears to degrade more noticeably with the static filters. This suggests that the filters may have become overly sensitive to modulating environmental information. This sensitivity could potentially lead to an over-filtering phenomenon, resulting in the removal of more than non-essential information. If the latent speech features are not adapted to the filtering mechanism, it can impact the quality of the representation. However, the combination of both functions, as proposed in our module, has shown to reduce the adverse impact of individual functions and maximize the potential of the filtering mechanisms.

Static Filters Size and ASR Performance

TABLE 4.7: WER of experiments using different number of static filters and their impact on ASR performance.

Dataset	Number of channel-filters			
	m=5	m=10	m=25	m=50
Clean (LibriSpeech)	4.62	4.56	4.53	4.49
Other (LibriSpeech)	9.81	9.73	9.67	9.59
CHiME4	19.89	19.67	19.50	19.29

To examine the performance improvement achieved by increasing the size of the soft tokens of static filtering embeddings, we present the performance using different numbers of static filters for PEFT. From Table 4.7, demonstrates that as the number of static filter embeddings increases, the ASR performance improves. This improvement is particularly noticeable in the case of real noisy speech, as seen in CHiME4. This is likely because with an increase in static filter embeddings, the network can allocate more capacity for the functional static filters to learn to process challenging noise interference, whereas the improvement in clean audio performance is less pronounced.

4.5 Summary

This study introduces Deep Filter Tuning (DFT), a novel approach designed to optimize the effectiveness of soft-tokens in Parameter-Efficient Fine-Tuning (PEFT)

for automatic speech recognition (ASR). By incorporating both static and adaptive filtering mechanisms, DFT effectively regulates the flow of information to adapt the frozen representations of pre-trained networks, enabling the model to handle adverse environmental distortions. This approach enhances the model's robustness in recognizing speech and improves content representation, even under challenging conditions.

Our comprehensive empirical evaluation reveals that DFT consistently delivers a relative gain of over 12% in ASR performance compared to conventional prompt tuning methods, while only adjusting 0.38% of the full model's parameters. This demonstrates the method's exceptional efficiency in improving ASR performance across various domain environments. Consequently, DFT offers a more parameter-efficient approach to prompt tuning, with the potential to greatly enhance the adaptability and accuracy of ASR systems in real-world applications.

Although prompt tuning has shown promise for PEFT in NLP tasks, our research corroborates the findings of [143], highlighting significant challenges when applying this approach to SSL pre-trained speech encoders for ASR. Our results indicate that prompt tuning often underperforms compared to other PEFT methods. Preliminary analysis suggests this gap stems from issues with the prompt attention mechanism, likely due to misalignment between soft tokens and continuous hidden speech representations. In contrast to NLP, where tuning operates at the discrete token level and hidden representations align closely with tokenized text, speech representations are influenced by complex factors like speaker identity and non-stationary noise, complicating the adaptation process. This emphasizes the need to either improve the representation of hidden speech states and prompt tokens or to enhance the delivery of instructional signals to frozen representations for downstream adaptation, with our focus on the latter. Enhancing these instructional signals could bridge the gap in prompt-based tuning, paving the way for more effective adaptation strategies in speech recognition.

While DFT incorporates a low-rank linear layer similar to adapter tuning, the two approaches serve different purposes. DFT adapts latent speech representations specifically to facilitate its filtering mechanism, with the gradient of the low-rank unit tied to the deep filters. This contrasts with vanilla adapter tuning, where adaptation is directly optimized for the downstream task. Additionally, our study

recognizes a limitation in PEFT research—specifically, the smaller scale of pre-trained model backbones in speech tasks compared to NLP. Currently, it's uncommon to see speech encoders with billions of parameters due to the vast data and resources required for training. Moreover, speech processing involves handling long sequences, and deploying such large-scale models could negatively impact latency in real-time applications like streaming ASR. Our work emphasizes practical deployment, aiming to enhance speech recognition robustness, prioritizing utility over scaling up model size.

Chapter 5

Alternative Speech Representations: Multi-Band Frequency Reconstruction in Psychoacoustic Neural Coding

5.1 Introduction

In this chapter, we focus on constructing discrete speech representations using a neural audio codec to achieve high-fidelity reconstruction. Our primary objective is to preserve information and learn meaningful, expressive speech representations throughout the tokenization process. This form of compression, which converts continuous speech sequences into subsampled spatial units, is inherently lossy. As a result, the introduction of noise and artifacts into the embedded representations due to missing information and misrepresentation presents an unavoidable challenge that we aim to address.

To this end, we propose a novel approach that leverages multi-band spectral decomposition for efficient audio compression and reconstruction. Our speech representation employs multi-band spectral residual vector quantization (MBS-RVQ) to partition and quantize distinct frequency bands. This technique aligns with psychoacoustic principles to enhance perceptual fidelity while optimizing bitrate

allocation. Furthermore, we introduce a high-compression Neural Audio Codec (NAC) variant designed to achieve a 12.5 kHz compression rate while preserving high audio fidelity. This reduction in sampling rate is aimed at enhancing inference efficiency within the transformers of LLMs. Note that this work lays the foundational groundwork in representation learning, with future efforts planned to integrate unit speech modality into LLMs for downstream tasks.

We explore the motivations behind our research, detail our methodology, and present an ablation study on our learned representations. This study assesses how well our approach preserves information and the expressiveness of the learned speech representation, crucial for reducing noise distortion in generative speech. These enhancements improve clarity and thereby boost speech recognition performance. This chapter seeks to address the following questions:

- What are the challenges of constructing speech representations with neural audio codec?
- What are the learning objectives in a neural audio codec?
- How can we enhance the preservation of information in NAC for more effective and efficient speech representations?
- What do we use to evaluate information preservation?
- How does our proposed work compare to the recent SOTA models?

5.2 Motivation

In recent years, the field of machine learning has undergone a significant transformation with the advent of large foundational models. Known for their vast scale and diverse training data, these models have dramatically reshaped our understanding of machine learning capabilities. Foundational models, such as the Generative Pre-trained Transformer (GPT) and other large language models (LLMs), have demonstrated remarkable proficiency in tasks that emulate human cognitive skills. These include reasoning and knowledge application, creativity and imagination, and linguistic fluency. Comparable to the abilities of a young adult, these models

excel in generating coherent text, performing accurate language translation, and engaging in meaningful and contextually appropriate dialogue.

Given their impressive performance, there has been a concerted effort to leverage the power of these foundational models across different domains. Specifically, adapting these models to new modalities such as speech aims to enhance tasks like speech recognition and spoken dialogue systems. This adaptation often involves borrowing foundational knowledge from LLMs. To effectively integrate speech with LLMs, it is crucial to discretize the speech into a tokenized form that aligns with the input format of these models. Consequently, learning an alternative speech representation that tokenizes speech into units using a neural audio codec (NAC) becomes essential.

Neural Audio Coding (NAC) has become a pivotal technology in speech and audio processing, renowned for its high-fidelity compression and reconstruction capabilities [58, 82, 129, 173–175]. This technology effectively reduces the size of audio files, optimizing storage, transmission, and playback while preserving high audio quality.

The robustness of NAC in capturing condensed and highly expressive semantic and acoustic details of speech has led to its widespread adoption in developing discrete tokens for speech-language models [59]. These tokens have further facilitated integration with large language models (LLMs), enabling an improved understanding of the speech modality. This integration has shown significant potential in advancing human-AI interactions by enhancing voice comprehension and enabling more natural and intuitive exchanges [176–178], establishing a promising connection between audio technology and language understanding.

NAC fundamentally relies on the neural autoencoder, a component engineered to capture essential audio features and compress them into compact, tokenized representations [179, 180]. To address the inherent challenges of lossy tokenization, many NAC systems incorporate Residual Vector Quantization (RVQ) [82, 181]. RVQ is crucial for improving information preservation, as it refines the quantization process through multiple stages. This progressive refinement reduces reconstruction errors and enhances fidelity to the original signal, ensuring higher quality audio outputs.

RVQ (illustrated in Figure 5.1) offers several advantages but also faces notable challenges, particularly in its reliance on hierarchical residual modeling [182]. This process involves capturing the residual details—those not quantized in earlier stages—through successive stages. However, as these layers progress, they frequently encounter complex or noisy residuals—errors that are carried forward from the previous quantization stages and accumulate. This complexity makes effective encoding challenging and can result in less meaningful speech unit representations of noises, leading to diminishing returns. Each additional stage tends to capture less information, which decreases the overall efficiency of the system. As such, we aim to improve the efficiency and preservation of information in RVQs in the following section.

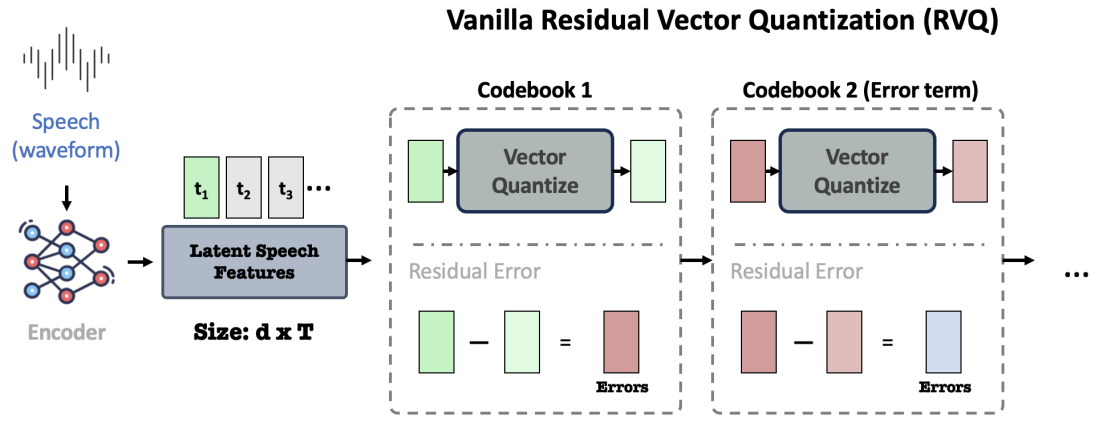


FIGURE 5.1: This diagram illustrates the process of Residual Vector Quantization (RVQ) applied to speech processing. Initially, speech waveforms are encoded into latent speech features. These features are then sequentially quantized using multiple codebooks in the RVQ framework. In Codebook 1, the latent features are vector quantized to produce a quantized output and a residual error. This residual error serves as the input to the next codebook, Codebook 2, which quantizes the error from the previous codebook to refine the approximation further. This process is repeated, with each subsequent codebook targeting the residual error from the previous quantization step, effectively stacking error terms to progressively minimize the overall quantization error.

5.3 Methodology

To address the inefficiency of RVQ in concentrating all information into the first codebook, we propose alternative approaches that distribute information more effectively across the codebooks, enabling each to specialize in distinct regions of

the feature space. In particular, we introduce a multi-band frequency quantization approach, where speech signals are segmented into multiple frequency bands. Each band is then assigned to a dedicated quantizer, allowing for focused and specialized processing of specific frequency components. This method not only enhances the representational capacity of the system but also significantly improves the overall efficiency and robustness of the quantization process.

5.3.1 Preliminaries: Multi-band Frequency Coding

Multi-band residual vector quantization. In traditional methods, a multi-band model divides the frequency domain of a discrete-time signal, $x[n]$, into different frequency bands using the Fast Fourier Transform (FFT). Mathematically, the FFT of the signal can be expressed as:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}, \quad k = 0, 1, \ldots, N-1 \tag{5.1}$$

where $X[k]$ represents the complex frequency components of the signal, N is the total number of samples, and k is the frequency index. We define K non-overlapping frequency bands, each corresponding to a specific range of frequencies, denoted as $\mathbf{B}_k$ can be defined as:

$$\mathbf{B}_k = \{f : f_{\min,k} \leq f < f_{\max,k}\}, \quad k = 1, 2, \ldots, K \tag{5.2}$$

where $f_{\min,k}$ and $f_{\max,k}$ denote the minimum and maximum frequencies of the k^{th} band, respectively. We first split the input signal into multiple frequency bands and apply an inverse FFT to reconstruct the time-domain waveform for each band. These reconstructed waveforms are then individually processed by a time-domain neural autoencoder, serving as the foundation for the multi-band neural tokenizer. The encoder within each autoencoder quantizes the signal to produce discrete representations for each frequency band. Although this approach is straightforward for processing the signal in separate bands, it increases latency, as each band-pass signal requires individual encoding steps through the autoencoder (i.e., scaling the computational burden in terms of FLOPs by the number of K bands) before being

recombined to reconstruct the original signal. This added computational complexity limits its suitability for real-time streaming applications.

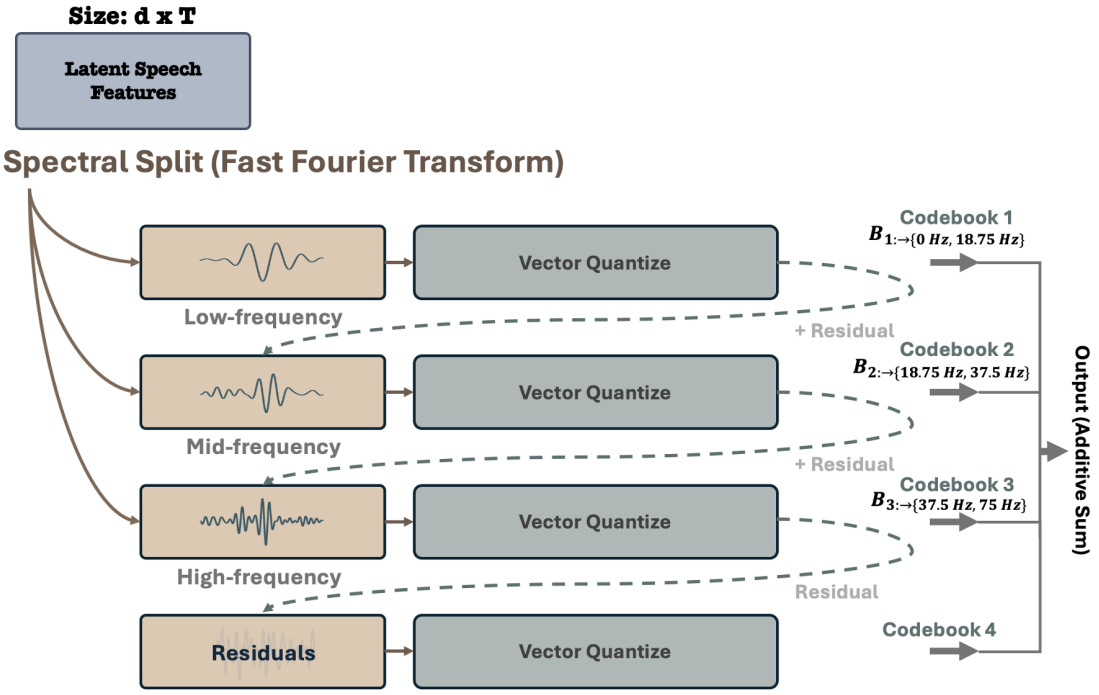


FIGURE 5.2: Illustration of the MBS-RVQ process: Fast Fourier Transform (FFT) is applied to the encoded latent representation to isolate specific frequency bands, capturing targeted spectral information for each codebook. The filtered representation is reconstructed using inverse FFT before undergoing quantization. The quantization residuals are then passed to the next codebook as RVQ features, creating a hierarchical and progressively refined representation across codebooks.

Therefore, rather than splitting the bands at the input, we propose operating the multi-band processing within the encoder's latent space using its latent features. The audio is first encoded into a compressed representation z, capturing both spectral and temporal features. The latent representation $z \in \mathbb{R}^{d \times T}$, where d is the channel dimensionality and T is the temporal length, is then decomposed into multiple frequency bands. Spectral band splitting is achieved using the FFT, as previously described. Subsequently, each band is quantized using the techniques from [58], which factorize and L2-normalize the codes to improve codebook utilization and bitrate efficiency. The quantization process is conducted sequentially, starting with the lowest frequency band and progressing through the mid-frequency band to the highest frequency band. Specifically, the frequencies of 18.75 Hz, 37.5

Hz, and 75 Hz are used, corresponding to scale factors of 4, 2, and 1, respectively. This setup aligns with the 75 Hz bandwidth of the latent representation, which is achieved by compressing 24 kHz audio by a factor of 320 using the encoder. Additionally, the residuals from each quantizer are forwarded to the following stages in a residual vector quantization framework, leveraging the principle of successive refinement. This method allows each stage to quantize the residual error from the previous stage, thereby improving the accuracy and detail of the audio representation. An illustration of the proposed multi-band spectral residual vector quantization (MBS-RVQ) is presented in Figure 5.2.

We note that adopting a multi-band split to enhance the meaningful expression of representations during the quantization process aligns with psychoacoustic studies demonstrating how different frequency ranges convey distinct types of information. Low-frequency bands contain concentrated energy crucial for speech intelligibility, while mid-frequency bands capture formant structures essential for content articulation. High-frequency bands, on the other hand, provide fine details such as speaker identity, pitch, and timbre, which are vital for achieving naturalness in speech. Additionally, these higher-frequency bands typically help to provide information to enhance the spatial and ambient qualities of audio recordings [183, 184]. As such, learning in such a manner helps optimize the allocation of encoding resources across different aspects of the audio spectrum, ensuring that each segment of data is processed in a way that maximizes the preservation and clarity of crucial auditory cues. This approach not only facilitates the natural quantization of perceptually meaningful expressions of quantized units but also improves the efficiency and effectiveness of data compression. Additionally, it significantly enhances the overall perceptual quality of the output.

Method Insight: Multi-band modeling improves generative quality by leveraging the perceptual entropy bound. Let $\mathbf{x}(t)$ be a continuous-time audio signal defined over $t \in \mathbb{R}$, and let $\hat{\mathbf{x}}(t)$ be its compressed approximation. Suppose the human auditory system is modeled by a set of perceptual filters that divide the frequency axis into K critical bands $B_1, B_2, \ldots, B_K$. Each band B_k corresponds to a region where the ear has distinct sensitivity levels (e.g., Bark or Mel scales). For each band B_k, let $P_k(\mathbf{x}(t))$ represent the perceptual threshold function indicating the maximum allowable distortion before artifacts become noticeable,

and let $D_k(\mathbf{x}(t), \hat{\mathbf{x}}(t))$ denote the band-specific distortion introduced by compression. The perceptual entropy E_p of $\mathbf{x}(t)$ is defined as the minimal bit rate needed so that $D_k(\mathbf{x}(t), \hat{\mathbf{x}}(t)) \leq P_k(\mathbf{x}(t))$ for all k, ensuring transparent audio compression:

$$E_p = \sum_{k=1}^{K} \min\{R_k : D_k(\mathbf{x}(t), \hat{\mathbf{x}}(t)) \leq P_k(\mathbf{x}(t))\}.$$

Theorem 5.1 (Perceptual Entropy and Masking Bounds)**.** *For the given audio signal $\mathbf{x}(t)$ and compressed representation $\hat{\mathbf{x}}(t)$, the perceptual entropy E_p satisfies the following lower bound when optimal multiband modeling is employed:*

$$E_p \;\geq\; \sum_{k=1}^{K} H(B_k \,|\, \mathbf{x}(t)) \;-\; \sum_{k=1}^{K} \Delta\big(B_k, \mathbf{x}(t)\big),$$

where $H(B_k \,|\, \mathbf{x}(t))$ is the Shannon entropy of the signal components within band B_k, and $\Delta(B_k, \mathbf{x}(t))$ is the perceptual masking effect that reduces the effective entropy by accounting for inaudible distortions in band B_k.

Theorem 5.1 characterizes how multiband modeling leverages psychoacoustic properties to achieve lower bit rates without sacrificing perceived audio quality. The first term, $\sum_{k=1}^{K} H(B_k \mid \mathbf{x}(t))$, represents the total intrinsic entropy over the K critical bands, analogous to the information-theoretic bound one would calculate if there were no masking effects. However, human hearing does not require perfect fidelity in all frequency regions; many distortions remain hidden under the masking threshold [185]. This phenomenon allows the codec to allocate fewer bits to masked regions without degrading the subjective audio experience.

In mathematical terms, $\Delta(B_k, \mathbf{x}(t))$ represents the "masking offset" that effectively reduces the bits needed for band B_k. Due to this offset, the total perceptual entropy E_p can be substantially smaller than the naive entropy sum, reflecting how frequency regions with strong masking require fewer bits for transparent encoding. This principle underlies the design of many perceptual audio codecs [186], where an FFT first decomposes the signal into subbands aligned with the human ear's critical bands, and then quantization is adapted based on psychoacoustic models [187].

Multiband Decomposition and Critical Bands. Breaking the signal into multiple bands $B_1, \ldots, B_K$ shows precisely how each region contributes differently to perceived audio quality. Low-frequency bands provide the foundational content and contribute significantly to intelligibility, especially for speech. Mid-frequency bands carry the bulk of formant information in speech signals and timbral elements in music [188], thus demanding higher precision where the ear is particularly sensitive. High-frequency bands, while less crucial to intelligibility, play an important role in conveying airiness, brightness, and certain aspects of speaker identity or timbre. By separately encoding these frequency regions, the codec can exploit masking more effectively and direct available bits to where they yield the greatest perceptual benefit.

Interpretation for Generative Models. Beyond compression, decomposing a signal into psychoacoustically meaningful subbands leads to more interpretable latent representations. This separability helps generative models or neural-based audio tokenizers isolate different attributes—such as speaker identity, timbral quality, or content—so that one can manipulate or transfer these features in a controlled manner. For instance, a style-transfer model might only adjust the higher bands responsible for speaker or instrument color while leaving low- and mid-frequency features intact, thus preserving intelligibility or musical pitch structure.

Validity of MBS-RVQ at latent representation space. We note the theoretical distinction in performing band splitting at the input space versus applying it to the latent representations of an autoencoder. This raises key considerations regarding the preservation of psychoacoustic properties within the latent representation space. However, we argue that latent representations retain the psychoacoustic properties of speech is supported by the Lipschitz continuity of the encoder.

Formally [189], if $f : \mathbb{R}^n \to \mathbb{R}^m$ is Lipschitz continuous with constant L, then for any two speech signals $\mathbf{x}_1, \mathbf{x}_2$:

$$\|f(\mathbf{x}_1) - f(\mathbf{x}_2)\| \leq L \|\mathbf{x}_1 - \mathbf{x}_2\| \tag{5.3}$$

Psychoacoustic cues such as formant positions, harmonic relationships, and energy distributions in critical frequency bands are primarily reflected in subtle variations of the speech signal's waveform. Lipschitz continuity ensures that these modest yet

perceptually crucial differences are neither excessively magnified nor erased when the signal is transformed into the latent domain. In other words, two psychoacoustically similar speech signals cannot become drastically separated in latent space [190, 191]. Furthermore, empirical evidence in representation learning supports this notion, demonstrating that neural networks constrained by Lipschitz continuity typically learn more structured representations, within which subtle perceptual attributes remain discernible [192, 193].

Consequently, if an autoencoder's encoder maintains Lipschitz continuity, the latent embeddings it generates for speech signals can be expected to closely reflect the psychoacoustic characteristics present in the original waveform. Minor spectral changes perceived by listeners, such as slight shifts in vowels or sibilants, correspond to small changes in latent space, thus helping to preserve the overall psychoacoustic signature of the speech. This argument provides the basis for the notion that, although a strict one-to-one psychoacoustic fidelity is not mathematically guaranteed, in practice, Lipschitz continuity significantly mitigates the risk of losing important auditory details in the encoder's output.

5.3.2 Model Architecture

Our NAC, called MUFFIN: Multi-band Frequency Reconstruction for Psychoacoustic Neural Coding, features an autoencoder architecture inspired by HiFi-Codec [127], utilizing a fully convolutional encoder-decoder network for temporal downscaling. We adopt the same striding configuration $(2, 4, 5, 8)$, optimized for 24kHz audio waveforms, achieving a total downsampling factor of 320 in the default configuration. A key component of our convolutional block is the multi-receptive field (MRF) fusion, adapted from [119], which aggregates outputs from residual blocks with varying dilated kernel sizes. This allows the model to capture dependencies across multiple temporal scales, enhancing its capacity to handle long-range sequential information.

To enhance the model's representational power, we integrate an inverted bottleneck layer with residual skip connections, inspired by ConvNeXt [130], which increases channel dimensions and adds neural complexity. This design aligns with SOTA

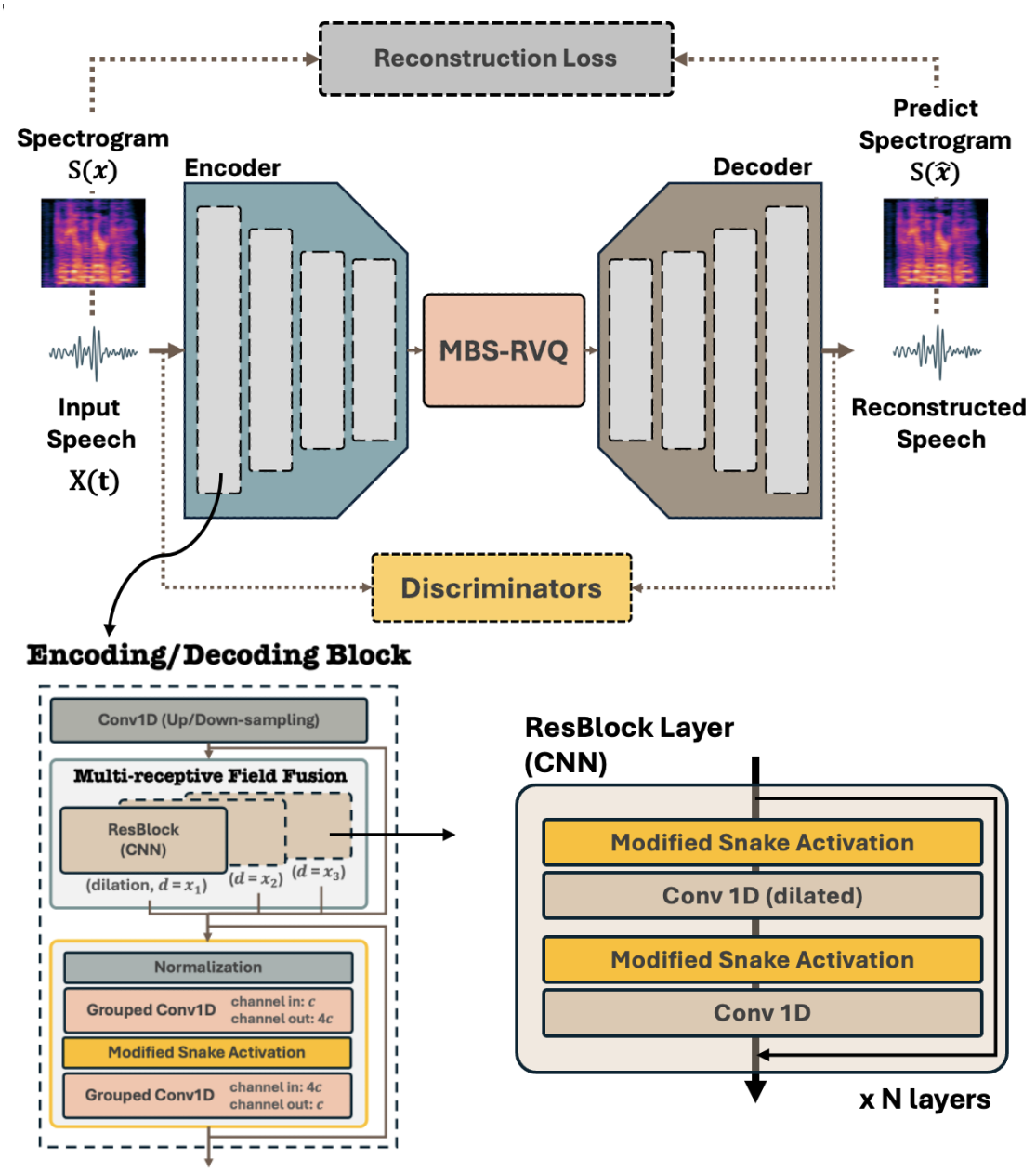


FIGURE 5.3: Architecture of MUFFIN incorporating a fully convolutional structure. The autoencoder blocks implement transformer-like operations through a (1) multi-receptive field communication layer for spatial dependency modeling, and (2) an inverted bottleneck layer for increased neural complexity. Besides, the layer block with a modified snake activation, as illustrated in the diagram, used to employ ReLU or LeakyReLU activations.

architectures [194–196], enabling the model to learn richer, more detailed representations, as shown in Figure 5.3. To reduce latency by improving the computational efficiency from the channel-upsampling layer, we employ grouped convolutions that upscale in groups of 32 channels, which significantly reduces the number of model

parameters and computations. The model consists of 46.1M parameters, with 34.2M in the encoder and 11.9M in the decoder. The Multiply-Accumulate Operations (MAC) reach 31.6G per second of audio sampled at 24 kHz, demonstrating enhanced performance efficiency compared to the HiFi-Codec's model, which has 61.5M parameters and taking 44.4G of MACs computational steps.

It is noteworthy that our convolutional block mirrors transformer functionality. The communication block from MRF mimics self-attention, enabling global temporal interactions, while the complexity block from the inverted bottleneck parallels the transformer's feed-forward layer, adding depth and expressiveness. This transformer-inspired architecture efficiently captures hierarchical temporal patterns, ensuring robust audio signal reconstruction performance.

5.3.3 Periodic activation function

To enhance periodic modeling for better spectral preservation and high-fidelity reconstruction, we draw inspiration from Kumar et al. [58] by replacing all Leaky ReLU activations with the snake activation function, $x + 1/\alpha \sin^2 \alpha x$, which better preserves high-frequency information [197] and maintain Lipschitz continuity, since the derivative of snake activation function is bounded by a constant of 1. We further enhanced this function by introducing amplitude and bias adjustments to improve overall performance. In the original formulation, the parameter α controls the frequency of the periodic component, while its reciprocal $1/\alpha$, attenuates the amplitude. To overcome this limitation, Evans et al. [198] introduced a learnable scaling factor β to adjust the amplitude independently. However, as shown in Figure 5.4, this adjustment can lead to increased variance due to the amplified periodic magnitude relative to the input x. To address this, we propose adding a bias term γ that learns to shift the output, ensuring better adaptation to the new scale while preserving consistency with the input range. This approach mitigates variance issues and offers greater flexibility in fitting data, significantly enhancing high-fidelity reconstruction.

In the course of modelling, Figure 5.4 shows four distinct data scenarios, where we observe that the vanilla activation function inadequately models regions exhibiting high-frequency patterns, likely due to trade-offs involving amplitude preservation.

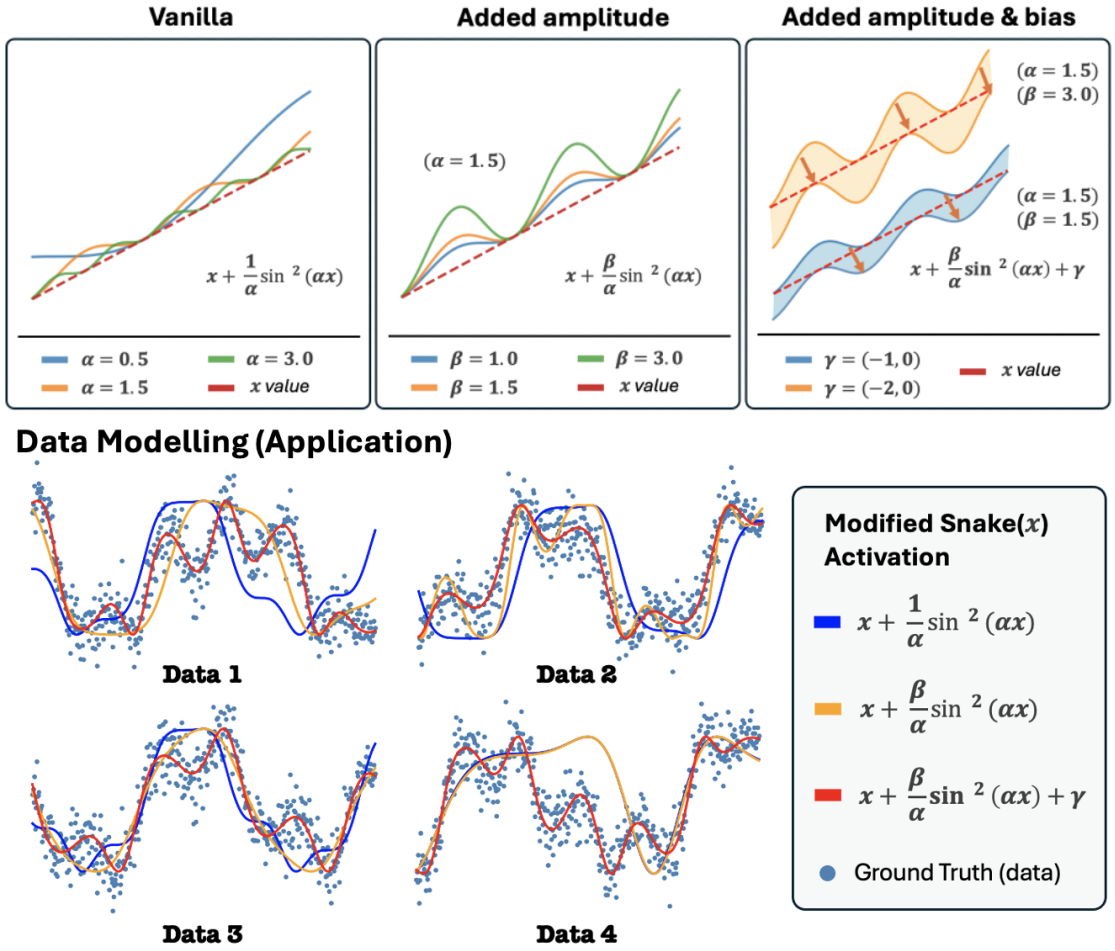


FIGURE 5.4: Illustration of the modifications to the vanilla snake activation and its behavior in actual modeling for sequential time datasets.

Although the introduction of the β parameter mitigates this issue to some extent, it simultaneously introduces regions of elevated variance, which compromises overall stability. In contrast, the inclusion of a bias term γ provides a more robust solution, effectively stabilizing the model and reducing the observed variance from over or underestimating outcome.

5.3.4 Training objectives

MUFFIN is trained on typical reconstruction tasks to minimize the error in recreating audio signals, with the aim of producing outputs that are perceptually indistinguishable from the original audio. Our primary goal is to ensure that the reconstructed audio sounds natural and maintains high fidelity to the original input. This evaluation indirectly assesses the effectiveness of quantized speech in

preserving information and capturing the expressivity of the input speech, which is crucial for high-fidelity reconstruction. Following principles from HiFi-Codec [127], our training leverages two key components in the loss function to achieve this.

Reconstruction Loss We employ a multi-scale mel-spectrogram reconstruction loss, calculated as the L1 distance between predicted and target mel-spectrograms over multiple time scales. This is represented as:

$$\mathcal{L}_{\text{mel}} = \sum_{i \in l} \|S_i(x) - S_i(\hat{x})\|_1 \tag{5.4}$$

where S_i is a 64-bins mel-spectrogram derived from an STFT, with a window size of 2^i and a hop length of $2^i/4$ for $l = 7, 8, 9, 10, 11$. Unlike prior work [127], i.e., using 80 mel-spectrogram bins, we opted for a lower bin count based on perceptual evaluations, which revealed improved naturalness in the reconstructed audio while preserving key spectral features. Although reducing the bin count compromises frequency resolution, our results indicate that stricter distance reconstruction criteria, while preserving more information, may slightly harm perceptual quality.

Discriminative Loss We use three discriminators: a multi-scale STFT discriminator (MS-STFT) Zeghidour et al. [82], a multi-period discriminator (MPD), and a multi-scale discriminator (MSD) [119] to enhance perceptual quality through adversarial learning. Each discriminator contributes a complementary inductive bias: MS-STFT enforces spectral consistency across frequency scales, MPD captures periodic structures such as pitch and harmonics and is particularly effective in reducing artifacts in the generated waveform, and MSD assesses overall signal quality across multiple temporal resolutions. Together, these discriminators guide the generator to produce waveforms that are not only statistically realistic but also perceptually coherent and artifact-free. Additionally, we also adopt the Hinge-GAN [199] adversarial loss formulation and incorporate the L1 feature matching loss [200].

RVQ Commitment Loss We employ the simple codebook and commitment losses with stop-gradients from the original VQ-VAE formulation [201], backpropagating gradients through the codebook lookup using the straight-through estimator [202]. Note that each codebook corresponds to a spectral band split, utilizing filtered information when learning to optimize the code searching process.

5.4 Experiments

5.4.1 Data sources

We train our model on a modest collection of 1,600 hours of speech, music, and environmental sounds. For speech, we use LibriTTS [203] and EARS [204] datasets with expressive anechoic recordings of speech (585 and 100 hours, respectively). For music, we utilize Music4All [205] (910 hours). For environmental sounds, we use ESC-50 [206] (3 hours, 50 semantical classes with 40 examples per class, loosely arranged into 5 major categories: animal, human, natural sounds, interior, and exterior sounds). Music and environmental sounds are utilized to enable MUFFIN to learn broader audio expression features, enhancing the foundational representations. All audio was resampled to 24 kHz.

Prior to training, all audio files are truncated to a maximum duration of 10 seconds. A text file containing the paths to the processed audio files is provided to the data loader. This ensures a balanced distribution of speech and music file samples, mitigating data imbalance and preventing under-performance skewing towards vocal or instrumental reconstruction. During training, we apply voice activity detection to remove non-audio segments, optimizing learning efficiency. For each batch, 1-second audio segments are randomly selected from each instance and are zero-padded if shorter than 1 second. No additional data augmentation is applied to maintain simplicity in the experiment.

5.4.2 Model and training details

In our experiments, models were trained on two A800 GPUs for 300K iterations with a learning rate of 2e-4 and a batch size of 20 per GPU. The residual block configuration mirrors that of HiFi-Codec, featuring a 10-bit code lookup from the factorized codebook [58] and a dimensionality of 64. Additionally, we developed a low token-rate compression variant of MUFFIN to leverage the model's efficiency in capturing non-redundant information across different frequency bands. The high-compression MUFFIN variant increases the downsampling rate by 960× and 1920× in the encoding layers, producing latent representations at 25 Hz and

12.5 Hz, respectively. This configuration achieves audio quantization at 150 and 100 tokens per second, utilizing additional residual codebooks with a total of six and eight vector quantization layers. The same MBS-RVQ configuration is used, maintaining a 4:2:1 frequency ratio relative to the sampled frequency of the latent embeddings (partitioned on a logarithmic scale) for the default band splits. MUFFIN, operating at 12.5 kHz, is trained on 2-second audio segments and requires four A800 GPUs to support a batch size of 10 per GPU. Using a lower learning rate is crucial to prevent gradient explosion. The high-compression MUFFIN model comprises approximately 50.6M parameters, with 36.5M in the encoder and 14.1M in the decoder (both models share the same architectural depth, differing only in downsampling rates). It achieves significantly lower MACs at 16.9G per second of audio, enabling faster inference.

Since our training setup closely adheres to Hifi-Codec, one of the leading codec model in the field, we establish it as our baseline by retraining the model with the same configuration, allowing for accurate performance comparison. Additionally, we benchmarked our method against other prominent codecs, including OPUS, Encodec and DAC, all reconfigured to a transmission rate of 3.0 kB/s to assess performance at the same transmission rate as our codec reconstruction model. Furthermore, we evaluate high-compression MUFFIN against Mimi, the recent state-of-the-art codec model for 12.5 Hz, to demonstrate the improved performance of our work.

5.4.3 Evaluation objectives

We utilize the objective evaluation metrics outlined in codec-SUPERB [131] to assess the perceptual quality of different audio domains. Specifically, we incorporate metrics such as the Perceptual Evaluation of Speech Quality (PESQ) [42], Short-Time Objective Intelligibility (STOI) [43], STFT distance [207], Mel distance [208], and F0CORR (F0 Pearson Correlation Coefficient) [209]. The selection of these metrics for the corresponding audio domain is justified by the inclusion-exclusion criteria discussed in codec-SUPERB. Additionally, we employ two automated Mean Opinion Score (MOS) evaluation metrics, UTMOS [44] and ViSQOL [132], to assess

the perceptual quality of the codecs. These metrics are designed to closely approximate subjective listening tests, providing a more accurate and robust evaluation of codec performance. For speech evaluation, we use the test-clean and test-other set from LibriTTS and evaluate emotional speech reconstruction with IEMOCAP [210]. For music, we employ the GTZAN dataset [211], while environmental sounds are evaluated using audio from the BBC sound effects [212].

5.4.4 Experimental Results

TABLE 5.1: Objective evaluation of reconstructed speech from the LibriTTS dataset using various neural audio codec models. *GT* refers to the abbreviation for ground truth and bandwidth corresponds to transmission rates in kilobytes per second (kB/s). Except for high-compression MUFFIN, which uses a compression rate of ▽ : ×960 (25.0 Hz) and ▲ : ×1920 (12.5 Hz), the others have the compression rate of ×320 (75 Hz).

Test-Clean (LibriTTS)								
Model	Bandwidth	Token/s	STFT	Mel	PESQ	STOI	UTMOS	ViSQOL
GT	-	-	-	-	-	-	4.041	-
OPUS	3.0	-	5.728	2.796	1.132	0.715	1.264	2.878
Encodec	3.0	300	1.956	1.051	2.042	0.903	2.269	4.078
DAC	3.0	300	1.759	0.849	2.370	0.915	2.951	4.143
Hifi-Codec	3.0	300	1.618	0.765	2.712	0.943	3.831	4.410
Mimi	1.0	100	2.488	1.706	1.715	0.620	2.966	3.791
MUFFIN	3.0	300	**1.555**	**0.692**	**2.986**	**0.954**	4.017	**4.516**
MUFFIN ▽	1.5	150	1.626	0.755	2.525	0.937	4.035	4.345
MUFFIN ▲	1.0	100	1.663	0.807	2.360	0.932	**4.074**	4.225
Test-Other (LibriTTS)								
GT	-	-	-	-	-	-	3.453	-
OPUS	3.0	-	5.390	2.703	1.143	0.695	1.271	2.815
Encodec	3.0	300	1.998	1.119	1.960	0.888	2.026	4.017
DAC	3.0	300	1.813	0.913	2.220	0.897	2.497	4.053
Hifi-Codec	3.0	300	1.681	0.840	2.419	0.919	3.216	4.296
Mimi	1.0	100	2.515	1.688	1.611	0.612	2.498	3.679
MUFFIN	3.0	300	**1.615**	**0.758**	**2.658**	**0.934**	3.444	**4.454**
MUFFIN ▽	1.5	150	1.681	0.817	2.232	0.914	3.516	4.276
MUFFIN ▲	1.0	100	1.725	0.875	2.086	0.904	**3.560**	4.129

Table 5.1 and 5.2 compare the speech reconstruction quality of our MUFFIN variants against existing top performing NACs. Evaluation metrics include bandwidth and 'tokens/s', representing the number of tokens per second into which the audio signal is encoded, providing a measure of the tokenization efficiency of the neural coding module. We use complete audio samples from the LibriTTS evaluation dataset, with 4,837 samples for test-clean and 5,120 for test-other. Notably, MUFFIN achieves superior reconstruction fidelity, achieving the lowest distance error and outperforming competing NACs across various objective metrics. This includes HiFi-Codec, which serves as a strong baseline, as we retrained it on the same dataset using its original optimal training hyperparameters, confirming the effectiveness of the proposed framework. Furthermore, MUFFIN ▽: 25.0 Hz and ▲: 12.5 Hz achieves better UTMOS scores despite reductions in bandwidth and token rates. It appears that increasing the number of codebooks in our NAC system enhances the preservation of information from the original speech, resulting in improved perceptual quality as reflected by higher UTMOS scores. This suggests that with more codebooks, the system better captures nuanced details that contribute to the naturalness of the audio. However, while this strategy enhances perceptual naturalness, it may not uniformly improve all objective metrics. In fact, as the number of codebooks increases, especially at higher compression rates, some objective measures experience a decline. This decline may be attributed to the introduction of noise or artifacts by additional codebooks, which, while capturing more detail, also amplify aspects that negatively impact certain evaluation metrics (ideally, one may wish to use fewer codebooks, while designing a learning system in which the codebook can learn robust representations that capture speech detailed variation). Thus, the relationship between increased codebooks and system performance exemplifies a trade-off between improved naturalness and the potential degradation of other audio quality metrics. Nevertheless, it is notable that MUFFIN ▲ remains competitive with leading NAC models and even outperforms Mimi in terms of naturalness and reconstruction fidelity, achieving significant gains in audio quality while maintaining efficient compression rates.

Table 5.2 showcases MUFFIN's zero-shot reconstruction capability on the full 12-hour IEMOCAP dataset, which contains expressive emotional speech that was not used during training.While reconstruction fidelity declines across the NACs,

TABLE 5.2: Objective evaluation of the reconstructed speech from the IEMOCAP dataset was conducted using various neural audio codec models, following the same setup as before. Note that bandwidth corresponds to transmission rates in kilobytes per second (kB/s)

Model	Bandwidth	Token/s	STFT	Mel	PESQ	STOI	UTMOS	ViSQOL
GT	-	-	-	-	-	-	1.859	-
OPUS	3.0	-	2.361	1.586	1.207	0.478	1.242	2.642
Encodec	3.0	300	2.150	1.290	1.649	0.746	1.321	3.501
DAC	3.0	300	1.553	0.781	1.867	0.763	1.316	3.774
Hifi-Codec	3.0	300	1.447	0.755	1.998	0.763	1.564	3.651
Mimi	1.0	100	2.112	0.755	1.433	0.494	1.427	2.801
MUFFIN	3.0	300	**1.399**	**0.675**	**2.178**	**0.806**	1.903	**4.000**
MUFFIN ▽	1.5	150	1.392	0.703	1.844	0.748	**2.026**	3.612
MUFFIN ▲	1.0	100	1.429	0.754	1.726	0.723	2.019	3.376

as shown in the objective metrics, both the default and high-compression variants of MUFFIN demonstrate superior robustness, achieving higher naturalness in human-perceived audio quality based on UTMOS scores compared to the ground truth references, despite the high compression. Nevertheless, the prominent drop in reconstruction fidelity for emotional content highlights a challenge in preserving emotional nuances, which could potentially impair emotional recognition in downstream tasks.

TABLE 5.3: Objective evaluation of the reconstructed speech from the GTZAN dataset was conducted using various neural audio codec models, following the same setup as before. Note that bandwidth corresponds to transmission rates in kilobytes per second (kB/s)

Model	Bandwidth	Token/s	STFT	Mel	PESQ	STOI	F0CORR	ViSQOL
OPUS	3.0	-	7.786	3.462	1.081	0.424	0.727	2.414
Encodec	3.0	300	2.712	1.016	1.684	0.756	0.882	4.247
DAC	3.0	300	2.493	0.928	1.709	0.741	0.867	4.220
Hifi-Codec	3.0	300	2.517	0.954	1.674	0.727	**0.899**	4.170
MUFFIN	3.0	300	**2.360**	**0.866**	**1.815**	**0.760**	0.896	**4.298**
MUFFIN ▽	1.5	150	2.492	0.928	1.474	0.674	0.879	4.273
MUFFIN ▲	1.0	100	2.550	0.987	1.409	0.642	0.872	4.223

Tables 5.3 and 5.4 present zero-shot reconstruction results on full data samples for music and audio, specifically from the GTZAN and BBC datasets. We observe a general decrease in fidelity for music reconstruction, as the task becomes more

challenging due to the need to reconstruct multiple instrumental audio channels. From the table, we observe that HiFi-Codec achieves a higher F0CORR, indicating superior pitch accuracy and suggesting that its model structure better preserves vocal quality compared to other NACs. However, the difference in F0CORR between MUFFIN and HiFi-Codec is minimal, down to the finer decimal places, and MUFFIN consistently outperforms other NACs across the remaining metrics. Moreover, while MUFFIN ▽▲ achieves a higher compression rate and learns to encode more efficiently with decent reconstruction fidelity for music, as confirmed by its competitive ViSQOL scores, we observed that PESQ and STOI were significantly lower, as reflected in Table 5.2. This suggests that highly compressed models face challenges in preserving fine information and are more vulnerable to reduced generalizability in zero-shot inference.

TABLE 5.4: Objective evaluation of the reconstructed speech from the BBC dataset was conducted using various neural audio codec models, following the same setup as before. Note that bandwidth corresponds to transmission rates in kilobytes per second (kB/s)

Model	Bandwidth	Token/sec	STFT	Mel	ViSQOL
OPUS	3.0	-	6.093	2.984	1.000
Encodec	3.0	300	1.998	1.011	3.852
DAC	3.0	300	1.846	0.784	3.995
Hifi-Codec	3.0	300	1.773	0.795	4.009
MUFFIN	3.0	300	**1.658**	**0.720**	**4.065**
MUFFIN ▽	1.5	150	1.700	0.748	4.010
MUFFIN ▲	1.0	100	1.706	0.777	3.997

Table 5.4 showcases the strong generalizability of zero-shot reconstruction on general audio from the BBC dataset, underscoring the robustness and efficiency of MUFFINs when compared to other NACs. The results consistently demonstrate the improved quality of our neural codec for general audio reconstruction.

5.4.5 Ablation Studies of Deconstructing MBS Codes

Auditory feature in codebook representations In this section, we analyze the information encoded in MUFFIN's learned codebooks, which correspond to different auditory frequency bands. We will demonstrate that MUFFIN's learning

model, which leverages spectral multi-band splits to isolate perceptual characteristics of speech attributes guided by psychoacoustic research, allows each codebook to specialize effectively in those attributes. This method not only more accurately preserves speech information by focusing on targeted tasks but also minimizes the noise generated by modeling complexities. We will validate these advantages through comprehensive empirical ablation studies. Moreover, to illustrate the practical impact, we will provide demonstrations of audio reconstructed from each codebook, enabling readers to directly perceive and evaluate the qualitative differences.

1. Low-frequency bands (Codebook 1, range: 0–18.75 Hz) contain fundamental frequencies and strong harmonic content, which are crucial for conveying core speech information. However, these bands primarily capture the coarse aspects of speech expression, without fully addressing the articulation nuances of speech content. This differs from previous NAC models, where full-band RVQ often consolidates most speech content information within the first codebook. To evaluate the preservation of semantic content in these low-frequency bands, we measure the STOI and the word error rate (WER) using the pre-trained Whisper-large V3 [146] model, which directly infers from the NAC model's reconstructed audio. This allows us to compare the semantic fidelity encoded by each codebook against other frequency bands. Since the ASR model is trained on 16 kHz audio, we use the compatible LibriSpeech [166] test-clean dataset, which consists of 2,620 samples, with resampling applied during reconstruction. For each sample, speech is decoded from individual codebooks and processed by the ASR model. Additionally, we evaluate other NAC models, including the vanilla RVQ of MUFFIN's autoencoder, to examine how their codebooks capture and contribute to semantic information in speech.

Table 5.5, 5.6 demonstrate the contextual role of Codebook 1, as recognizing content from any codebook other than Codebook 1 in MUFFIN proves notably challenging, with recognition errors exceeding 100. Unlike traditional Residual Vector Quantization (RVQ), which consolidates most speech content in the first codebook, MUFFIN distributes information across Codebooks 1 and 2: core expressions are assigned to Codebook 1, while articulation details are primarily captured in the mid-frequency range of Codebook 2, providing it with more contextual information than Codebook 3. Notably, MUFFIN's Codebooks 1 and 2 achieve an STOI

TABLE 5.5: The table presents the WER of ASR performance on reconstructed speech from each NAC's codebook using the Whisper-large V3 pre-trained model.

	MUFFIN		RVQ (Vanilla)		Hifi-Codec	
Ground Truth	WER: 2.408, STOI: NA					
	WER	STOI	WER	STOI	WER	STOI
Recons (All Codes)	2.670	0.940	2.719	0.930	3.000	0.919
Codebook 1	70.322	0.644	75.615	0.702	154.12	0.572
Codebook 2	113.97	0.379	141.08	0.426	138.81	0.454
Codebook 3	190.90	0.436	100.35	0.157	100.39	0.090
Codebook 4	106.63	0.082	100.95	0.086	112.18	0.129

TABLE 5.6: (*Continued..*) The table presents the WER of ASR performance on reconstructed speech from each NAC's codebook using the Whisper-large V3 pre-trained model.

	DAC		Encodec	
Ground Truth	WER: 2.408, STOI: NA			
	WER	STOI	WER	STOI
Reconstructed (All Codes)	3.529	0.901	3.151	0.900
Codebook 1	36.107	0.731	33.718	0.764
Codebook 2	131.69	0.148	159.24	0.199
Codebook 3	100.22	0.079	152.75	0.121
Codebook 4	100.04	0.049	147.36	0.094

of 0.729. In contrast, NACs using traditional RVQ exhibit declining semantic information across residual codebooks, along with a marked decrease in STOI.

Mid-frequency bands (Codebook 2, range: 18.75–37.5 Hz) are essential for encoding formant information that captures the articulation of vowels and consonants. We demonstrate this by visualizing spectrograms of randomly sampled speech in Figure 5.5 and 5.6, which shows spectrograms of randomly sampled speech and highlights the incremental contributions of each codebook. We observe that the frequency of formants is emphasized when combining Codebooks 1 and 2, revealing clearer articulation patterns that distinguish vowels and consonants. Nonetheless, the spectral content remains relatively flat, lacking distinct speaker information from Codebook 3. Additionally, we plot an elbow curve of WER as codebooks are added incrementally, illustrating each codebook's contribution to preserving speech

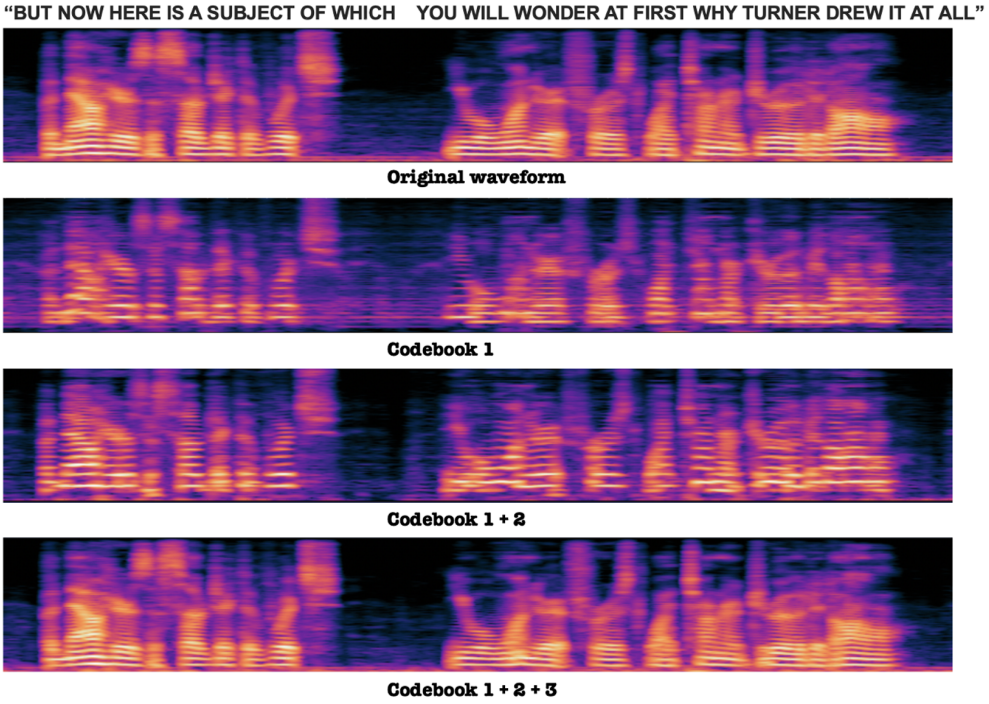


FIGURE 5.5: An illustration depicts a randomly sampled speech signal alongside its reconstruction using incremental codebooks.

content. A notable drop in error rate reflects improved clarity and speech intelligibility with enhanced articulation. We argue that Codebook 3 does not focus on contextual speech content, as using Codebook 3 alone results in high recognition errors, suggesting it does not contribute effectively to core speech intelligibility. This analysis is further supported by reconstructed audio samples and speaker classification results, both indicating that Codebook 2 primarily contributes to the articulation details of vowels and consonants.

High-frequency bands (Codebook 3, range: 37.5–75 Hz) capture essential auditory cues, including speaker identity, pitch, and timbre, which are important for distinguishing speakers and adding depth to reconstructed audio. To evaluate how well each codebook preserves speaker information, we randomly sample 600 speech files from VoxCeleb [213], representing six distinct speakers. Latent features are extracted from each codebook and average-pooled to create a vector representation for each sample. We then use t-SNE [171] to visualize these representations in 2D, aiming to identify speaker clusters across the codebooks. This analysis helps illustrate each codebook's role in preserving speaker characteristics in the encoded

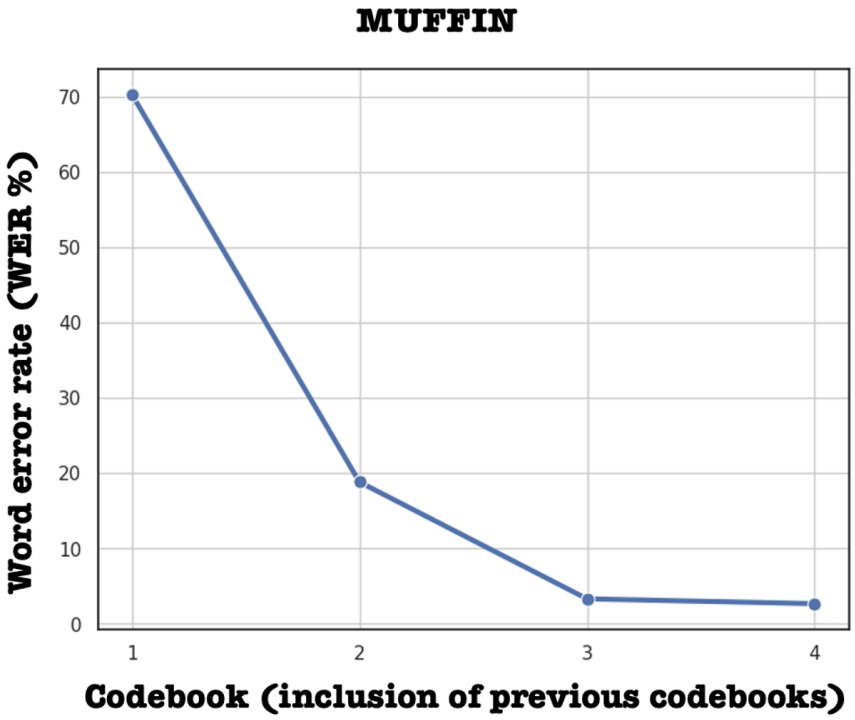


FIGURE 5.6: The elbow plot of the word error rate from whisper-large model, utilizing the same setup of incremental codebooks.

representations, revealed from Figure 5.7 the distinct clustering patterns across codebooks.

Codebook 1 exhibits a broad, dispersed distribution with some cluster overlap, suggesting it encodes foundational speech content with substantial variance across samples.

Codebook 2 forms a more concentrated central cluster with significant overlap between speakers, implying shared features likely related to vowel and consonant articulation.

Codebook 3, in contrast, shows well-separated clusters with the most distinct boundaries among speakers and minimal overlap, indicating its focus on capturing speaker-specific features. This distinct clustering should lead to a low distance error in speaker group classification, suggesting that Codebook 3 is highly effective at distinguishing between different speakers based on their unique vocal attributes.

FIGURE 5.7: A t-SNE plot showcasing each codebook, with speech randomly sampled from VoxCeleb, effectively represents six distinct speakers of the color code.

Lastly, Codebook 4 presents a compact distribution with considerable class overlap, suggesting it primarily encodes random or residual features that contribute minimally to core speech information. Additionally, the variations and the distributive patterns in the t-SNE plots suggest that each codebook captures slightly different information and likely serves a distinct purpose.

5.5 Summary

In conclusion, we have developed MUFFIN, a neural audio codec that significantly advances audio compression technology by incorporating multi-band spectral residual vector quantization (MBS-RVQ) within the latent space. By aligning the codec

architecture with psychoacoustic principles, MUFFIN not only optimizes the balance between compression rate and perceptual fidelity but also excels in preserving the detailed nuances of tokenized speech representations. This preservation enhances the clarity and distinctiveness of speech attributes in different specialized codebooks, essential for robust audio reconstruction across diverse audio domains, from human speech to complex environmental sounds.

Our comprehensive experimental evaluations across a wide range of audio datasets have validated MUFFIN's superior performance in foundational speech representation learning. Furthermore, we also show that MUFFIN substantially improves noise and artifact handling in reconstructed audio. This improvement is particularly beneficial for applications such as speech understanding and recognition, where maintaining clear, noise-minimized audio inputs is critical for accurate transcription.

Chapter 6

Conclusion and Future Work

In this thesis, we focus on advancing the speech representation learning framework to enhance its adaptability to ambient noise and improve the expressiveness of speech representations for more generalizable deep learning features. This effort is crucial as it directly influences the effectiveness of various downstream speech applications, particularly speech recognition systems in noisy, real-world settings. We have developed novel methodologies that significantly improve the noise robustness of the learned representations.

Furthermore, this work tackles the practical computational constraints associated with deploying these systems. We address the challenges of scaling foundational speech models, which often require substantial computational resources and extensive datasets for optimal adaptation. Our strategies are likely to be particularly beneficial for communities with limited budgets, providing cost-effective solutions while maintaining high performance levels.

Specifically, our research objectives are threefold. First, we explore modern, widely used neural speech representations by investigating recent self-supervised foundational speech representation learning frameworks. Our goal is to intrinsically achieve better noise robustness and more expressive latent embeddings with reduced channel information redundancy. Second, we refine domain adaptation strategies by implementing a more efficient, parameter-efficient fine-tuning approach for foundational speech models. Our strategy involves freezing a larger proportion of model parameters and selectively tuning a smaller, essential component.

This method is designed to prevent catastrophic forgetting during low-resource adaptation. Lastly, we explore alternative speech representations by tokenizing continuous speech into discrete units. This approach to speech discretization is designed to align with the input format of large-scale natural language models, thereby facilitating the integration of speech modalities into these more powerful systems. By doing so, we aim to establish a stronger connection between spoken language and large-scale natural language models, leveraging their prior knowledge to enhance the performance of downstream speech tasks. To conclude the work, Section 6.1 first summarizes the contributions proposed in this thesis. Finally, Section 6.2 discusses some of the future work to achieve high-performing speech LLMs that could listen and speak, integrating seamlessly with human-like conversational abilities. This will involve further refining the integration of speech processing models with advanced language understanding capabilities, aiming to create systems that can interact and respond in more dynamic, realistic scenarios.

6.1 Contributions

6.1.1 SSL Representations with Improved Noise Robustness and Redundancy Reduction

Chapter 3 explores how HuBERT, a self-supervised model based on the transformer architecture, addresses these challenges in speech representation learning. Despite its effectiveness, HuBERT lacks inherent noise robustness. To enhance this, we introduce deHuBERT, a training framework inspired by H. Barlow's redundancy reduction principle. This framework modifies HuBERT's training algorithm to include auxiliary losses that align the self- and cross-correlation matrices of noise-distorted embeddings with the identity matrix, improving the model's noise resilience and reducing redundancy.

Our findings show that deHuBERT outperforms both the original and noise conditioned HuBERT versions under noisy and clean speech conditions. Specifically, deHuBERT achieves Word Error Rates (WERs) of 6.3% and 13.2% on the test-clean and test-other datasets, respectively, matching baseline performances with

only noisy speech used for fine-tuning. Additionally, t-SNE visualizations confirm that deHuBERT embeddings are effectively noise-agnostic, unlike baseline HuBERT embeddings which retain noise features. These results underscore deHuBERT's superior noise robustness and overall efficacy in automatic speech recognition (ASR), even with limited training resources, highlighting its potential to significantly advance ASR technology in low-resource environments.

6.1.2 PEFT: Deep Filter Tuning

Chapter 4 investigates the adaptation of foundational-level self-supervised speech representation models. Adapting these fully-trained transformers to new tasks or domains is challenging due to the computational demands and risk of overfitting associated with fine-tuning extensive parameters, particularly in data-limited scenarios. To mitigate these issues, parameter-efficient fine-tuning (PEFT) has emerged as an effective strategy. This chapter examines various PEFT methods, highlighting their limitations in noise adaptation for downstream ASR tasks. We introduce a novel PEFT approach, deep filter-tuning (DFT), which draws inspiration from speech extraction techniques using feature-wise linear modulation (FiLM) for contextual content extraction.

Our empirical evaluation shows that DFT achieves a significant performance improvement, enhancing ASR accuracy by over 12% compared to traditional prompt tuning while adjusting only 0.38% of the total model parameters. This demonstrates DFT's efficiency and potential to significantly enhance transformer adaptability and accuracy in ASR applications across diverse environments.

6.1.3 Psychoacoustic Inspired Neural Audio Codec

Chapter 5 delves into the development of discrete speech representations via a neural audio codec, targeting high-fidelity audio reconstruction. Our primary aim is to effectively preserve information and craft meaningful, expressive speech representations through the tokenization process. Given that this compression method, which transforms continuous speech into subsampled spatial units, inherently incurs data

loss, it naturally introduces noise and artifacts. These distortions arise from incomplete information capture and misrepresentation, posing significant challenges that we address in our approach.

To tackle these issues, we propose an innovative method that employs multi-band spectral decomposition to optimize audio compression and reconstruction. Our technique, multi-band spectral residual vector quantization (MBS-RVQ), strategically partitions and quantizes distinct frequency bands. This aligns with psychoacoustic principles, enhancing perceptual fidelity while optimizing bitrate efficiency. Furthermore, we introduce a high-compression variant of our Neural Audio Codec (NAC) that achieves a compression rate of 12.5 kHz, maintaining high audio fidelity. This compression strategy is specifically designed to enhance the inference efficiency of transformers in large language models (LLMs). This work not only pioneers advancements in representation learning but also sets a foundational framework for integrating speech modalities into LLMs to improve performance in downstream applications.

6.2 Future Work

Integration of neural audio codec into LLMs for more powerful spoken speech applications. In this section, we discuss the implementation and functionality of a spoken dialogue framework designed to enhance human-computer interaction through advanced speech processing technologies. This framework [6, 115, 116, 214, 215] integrates a neural audio codec in chapter 5 with foundational Large Language Models (LLMs) [216–219] to efficiently process and generate spoken dialogue in real time. An overview of the framework diagram is presented in Figure 6.1.

The system begins by capturing spoken input, which is then converted into a waveform that is encoded into discrete speech tokens by the audio codec. These tokens are processed and formatted to include speech units and contextual prompts, enhancing the input's comprehensibility for transformer-based LLMs. The LLMs interpret the user's intent and generate appropriate textual and spoken responses. Notably, LLMs typically learns to generate response speech tokens that the audio

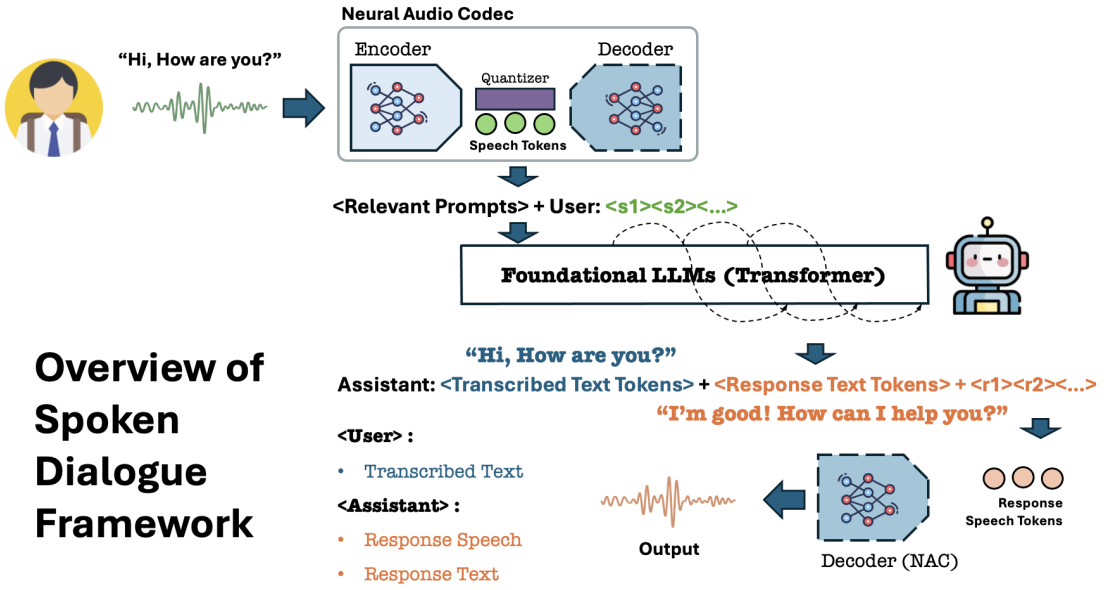


FIGURE 6.1: an overview of a Spoken Dialogue Framework utilizing a neural audio codec for speech processing and foundational Large Language Models (LLMs) for generating responses. The process begins with an input speech ("Hi, How are you?"), which is encoded into speech tokens and then decoded into text. These text tokens are combined with relevant prompts and processed by a transformer-based LLM, which generates appropriate textual and spoken responses, in this case, "I'm good! How can I help you?" This streamlined integration showcases how speech is transcribed, processed, and responded to within a sophisticated AI-driven dialogue system

codec decoder converts back into speech output. This innovative approach not only streamlines interactions but also significantly enhances the system's ability to understand and respond to natural language, making it more intuitive and effective for various applications, from customer service to interactive personal assistants.

However, current setups still confront several challenges. For instance, LLMs sometimes lose relevant contextual awareness [220, 221], leading to weak and unrelated responses. This underscores a significant deficiency in context memory, especially during the generation of long sequences. Additionally, these responses may exhibit biases and lack factual accuracy [222–224], posing potential risks to security and undermining social integrity. Moreover, despite advancements, LLMs still fall short of human emotional intelligence [225]. This deficiency often results in responses that fail to resonate with the emotional states of human users, further limiting their effectiveness in sensitive applications. As such, we identify several questions and propose directions for potential research to develop a more robust framework.

How do we ensure better contextual relevance? While Large Language Models (LLMs) are adept at generating coherent outputs, they often struggle with contextual preservation, primarily due to difficulties in recognizing and retaining key contextual elements. To tackle this limitation, recent research has introduced retrieval-augmented generation (RAG) [226, 227], an approach designed to enhance the model's ability to maintain context. This method dynamically retrieves relevant information from a knowledge base during the generation process and stores crucial knowledge and contextually pertinent data in a database. It continually refers to this stored data, incorporating relevant context as prompts to the input. Consequently, this approach not only improves the contextual awareness of the responses but also ensures that the outputs are relevant and informed by historical or background data, effectively narrowing the gap between generated content and the necessary contextual nuances. A potential research direction could focus on making RAG more effective and efficient, as the storage requirements are substantial, and accurately retrieving the appropriate context is critical for performance.

How do we ensure factual response? Ensuring factual responses from Large Language Models (LLMs) is crucial, especially in applications where precision and reliability are essential. To achieve this, various methods have been developed. Firstly, integrating structured knowledge bases during the training process allows LLMs to access verified factual data, reducing reliance on potentially inaccurate training patterns [228]. Secondly, incorporating dedicated fact-checking layers within the model enables automatic verification of generated content against trusted sources before finalization, enhancing accuracy. Another approach involves programming the model to cross-reference its responses with reliable external sources during generation, ensuring content accuracy and currency [229]. Additionally, implementing user feedback mechanisms enables the model to learn from inaccuracies and refine future outputs, gradually improving factual reliability [230, 231]. Lastly, training LLMs with diverse and rigorously fact-checked datasets can further prevent the generation of incorrect information [229]. Collectively, these strategies significantly boost the factual accuracy of responses produced by LLMs, ensuring their reliability and trustworthiness in various settings.

How do we teach LLMs to understand spoken sentimental value? Current LLMs struggle with recognizing and appropriately responding to varying speaking

styles, which is essential for effective spoken dialogue interactions. The objective is to improve LLMs' ability to understand how the same spoken content can provoke different responses depending on the speaker's style. To address this, [232] have introduced a novel dataset named StyleTalk, which comprises speech-to-speech data across diverse speaking styles. They also propose the Spoken-LLM framework that utilizes a two-stage training pipeline. This framework is designed to enable LLMs to capture both linguistic content and stylistic nuances. Looking ahead, future research could focus on incorporating multimodal inputs, such as visual cues or emotional tones, to enhance the model's contextual understanding in spoken dialogue. Additionally, exploring cross-linguistic capabilities to accommodate various languages and dialects, as well as developing real-time adaptation mechanisms that allow LLMs to dynamically learn from ongoing conversations, could further advance the field.

List of Publications and Awards

Conference Proceedings

- **Ng, Dianwen**, Yunqi Chen, Biao Tian, Qiang Fu, and Eng Siong Chng. "Convmixer: Feature interactive convolution with curriculum learning for small footprint and noisy far-field keyword spotting." In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3603-3607. IEEE, 2022.

- **Ng, Dianwen**, Jia Qi Yip, Tanmay Surana, Zhao Yang, Chong Zhang, Yukun Ma, Chongjia Ni, Eng Siong Chng, and Bin Ma. "I2CR: Improving noise robustness on keyword spotting using inter-intra contrastive regularization." In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 605-611. IEEE, 2022.

- **Ng, Dianwen**, Ruixi Zhang, Jia Qi Yip, Chong Zhang, Yukun Ma, Trung Hieu Nguyen, Chongjia Ni, Eng Siong Chng, and Bin Ma. "Contrastive speech mixup for low-resource keyword spotting." In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5. IEEE, 2023.

- **Ng, Dianwen**, Ruixi Zhang, Jia Qi Yip, Zhao Yang, Jinjie Ni, Chong Zhang, Yukun Ma, Chongjia Ni, Eng Siong Chng, and Bin Ma. "deHuBERT: Disentangling noise in a self-supervised model for robust speech recognition." In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5. IEEE, 2023.

- **Ng, Dianwen**, Yang Xiao, Jia Qi Yip, Zhao Yang, Biao Tian, Qiang Fu, Eng Siong Chng, and Bin Ma. "Small Footprint Multi-channel Network

for Keyword Spotting with Centroid Based Awareness." In *Proceedings of INTERSPEECH*. 2023.

- **Ng, Dianwen**, Chong Zhang, Ruixi Zhang, Yukun Ma, Trung Hieu Nguyen, Chongjia Ni, Shengkui Zhao, Qian Chen, Wen Wang, Eng Siong Chng, Bin Ma. "Adapter-tuning with Effective Token-dependent Representation Shift for Automatic Speech Recognition." In *Proceedings of INTERSPEECH*, pp. 1319-1323. 2023.

- **Ng, Dianwen**, Chong Zhang, Ruixi Zhang, Yukun Ma, Fabian Ritter-Gutierrez, Trung Hieu Nguyen, Chongjia Ni, Shengkui Zhao, Eng Siong Chng, and Bin Ma. "Are Soft Prompts Good Zero-Shot Learners for Speech Recognition?" In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10366-10370. IEEE, 2024.

- **Ng, Dianwen**, Kun Zhou, Yi-wen Chao, Zhiwei Xiong, Bin Ma, and Eng Siong Chng. "Multi-band Frequency Reconstruction for Neural Psychoacoustic Coding" In *Forty-Second International Conference on Machine Learning (ICML)*, 2025.

- **Ng, Dianwen**, Kun Zhou, Bin Ma, and Eng Siong Chng. "Thinking Fast and Slow: Robust Speech Recognition via Deep Filter-Tuning" In *Proceedings of INTERSPEECH*. 2025.

- Lan, Xiang, **Dianwen Ng**, Shenda Hong, and Mengling Feng. "Intra-inter subject self-supervised learning for multivariate cardiac signals." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, pp. 4532-4540. 2022.

- Yang, Zhao, **Dianwen Ng**, Xizhe Li, Chong Zhang, Rui Jiang, Wei Xi, Yukun Ma et al. "Dual-memory multi-modal learning for continual spoken keyword spotting with confidence selection and diversity enhancement." In *Proceedings of INTERSPEECH*. 2023.

- Yang, Zhao, **Dianwen Ng**, Xizhe Li, Chong Zhang, Rui Jiang, Wei Xi, Yukun Ma et al. "Dual Acoustic Linguistic Self-supervised Representation Learning for Cross-Domain Speech Recognition." In *Proceedings of INTERSPEECH*. 2023.

- Yang, Zhao, **Dianwen Ng**, Xizhe Li, Chong Zhang, Rui Jiang, Wei Xi, Yukun Ma et al. "A Unified Recognition and Correction Model under Noisy and Accent Speech Conditions." In *Proceedings of INTERSPEECH*. 2023.

- Ritter-Gutierrez, Fabian, Kuan-Po Huang, Jeremy HM Wong, **Dianwen Ng**, Hung-yi Lee, Nancy F. Chen, and Eng Siong Chng. "Dataset-Distillation Generative Model for Speech Emotion Recognition." In *Proceedings of INTERSPEECH*. 2024.

Awards

- **Best Student Nominee**, Ritter-Gutierrez, Fabian, Kuan-Po Huang, Jeremy HM Wong, **Dianwen Ng**, Hung-yi Lee, Nancy F. Chen, and Eng Siong Chng. "Dataset-Distillation Generative Model for Speech Emotion Recognition." In *Proceedings of INTERSPEECH*. 2024.

- **MISP Challenge ranked 6**, **Ng, Dianwen**, Jin Hui Pang, Yang Xiao, Eng Siong Chng. "MISP Challenge 2021: Multimodal Wakeup-Word Detection For Far-field and Noisy Environment—Technical Report". In *Signal Processing Grand Challenge (SPGC) of ICASSP 2022*. 2022

Bibliography

[1] Albert Zeyer, Parnia Bahar, Kazuki Irie, Ralf Schlüter, and Hermann Ney. A comparison of transformer and lstm encoder decoder models for asr. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 8–15. IEEE, 2019. xvii, 11

[2] Kanishka Rao, Haşim Sak, and Rohit Prabhavalkar. Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 193–199. IEEE, 2017. xvii, 11, 14, 16

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. xvii, 11, 17, 72

[4] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019. xx, 33, 64, 70, 72, 74

[5] Seyed Mahed Mousavi, Gabriel Roccabruna, Simone Alghisi, Massimo Rizzoli, Mirco Ravanelli, and Giuseppe Riccardi. Are llms robust for spoken dialogues? *arXiv preprint arXiv:2401.02297*, 2024. 1

[6] Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*, 2024. 1, 35, 41, 116

[7] Volodya Grancharov, David Yuheng Zhao, Jonas Lindblom, and W Bastiaan Kleijn. Low-complexity, nonintrusive speech quality assessment. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6):1948–1956, 2006. 1

[8] Kun Zhou, Shengkui Zhao, Yukun Ma, Chong Zhang, Hao Wang, Dianwen Ng, Chongjia Ni, Nguyen Trung Hieu, Jia Qi Yip, and Bin Ma. Phonetic enhanced language modeling for text-to-speech synthesis. In *Proc. INTERSPEECH*, 2024.

[9] Kun Zhou, Berrak Sisman, and Haizhou Li. Limited Data Emotional Voice Conversion Leveraging Text-to-Speech: Two-Stage Sequence-to-Sequence Training. In *Proc. Interspeech 2021*, pages 811–815, 2021. doi: 10.21437/Interspeech.2021-781. 1

[10] JM Tribolet, Peter Noll, B McDermott, and R Crochiere. A study of complexity and quality of speech waveform coders. In *ICASSP'78. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 586–590. IEEE, 1978. 1

[11] Barbara L Davis, Peter F MacNeilage, and Christine L Matyear. Acquisition of serial complexity in speech production: A comparison of phonetic and phonological approaches to first word production. *Phonetica*, 59(2-3):75–107, 2002.

[12] Jean Sawyer, HeeCheong Chon, and Nicoline G Ambrose. Influences of rate, length, and complexity on speech disfluency in a single-speech sample in preschool children who stutter. *Journal of fluency disorders*, 33(3):220–240, 2008.

[13] Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Emotional voice conversion: Theory, databases and esd. *Speech Communication*, 137:1–18, 2022. ISSN 0167-6393. 1

[14] Kun Zhou, Berrak Sisman, and Haizhou Li. Vaw-gan for disentanglement and recomposition of emotional elements in speech. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 415–422. IEEE, 2021. 1

[15] Kun Zhou, Berrak Sisman, Mingyang Zhang, and Haizhou Li. Converting Anyone's Emotion: Towards Speaker-Independent Emotional Voice Conversion. In *Proc. INTERSPEECH*, pages 3416–3420, 2020. 1

[16] Dirk Van Compernolle. Noise adaptation in a hidden markov model speech recognition system. *Computer Speech & Language*, 3(2):151–167, 1989. 2

[17] HM Cung and Yves Normandin. Noise adaptation algorithms for robust speech recognition. *Speech Communication*, 12(3):267–276, 1993.

[18] Dianwen Ng, Yunqi Chen, Biao Tian, Qiang Fu, and Eng Siong Chng. Convmixer: Feature interactive convolution with curriculum learning for small footprint and noisy far-field keyword spotting. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3603–3607, 2022. doi: 10.1109/ICASSP43922.2022.9747025. 2

[19] Ozlem Kalinli, Michael L Seltzer, Jasha Droppo, and Alex Acero. Noise adaptive training for robust automatic speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):1889–1901, 2010. 2

[20] Jinyu Li, Li Deng, Yifan Gong, and Reinhold Haeb-Umbach. An overview of noise-robust automatic speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(4):745–777, 2014.

[21] Steven Rennie, Trausti Kristjansson, Peder Olsen, and Ramesh Gopinath. Dynamic noise adaptation. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages I–I. IEEE, 2006. 2

[22] Siddique Latif, Rajib Rana, Sara Khalifa, Raja Jurdak, and Julien Epps. Direct modelling of speech emotion from raw speech. *arXiv preprint arXiv:1904.03833*, 2019. 2

[23] ZHOU KUN. *EMOTION MODELLING FOR SPEECH GENERATION*. Phd thesis, National University of Singapore, 2022.

[24] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*, 2019. 20

[25] Kun Zhou, Berrak Sisman, and Haizhou Li. Transforming Spectrum and Prosody for Emotional Voice Conversion with Non-Parallel Training Data. In *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, pages 230–237, 2020.

[26] Alexei Baevski, Steffen Schneider, and Michael Auli. vq-wav2vec: Self-supervised learning of discrete speech representations. *arXiv preprint arXiv:1910.05453*, 2019.

[27] Huaizhen Tang, Xulong Zhang, Jianzong Wang, Ning Cheng, and Jing Xiao. Avqvc: One-shot voice conversion by vector quantization with applying contrastive learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4613–4617. IEEE, 2022. 2

[28] Kun Zhou, Berrak Sisman, Rajib Rana, Björn W. Schuller, and Haizhou Li. Speech synthesis with mixed emotions. *IEEE Transactions on Affective Computing*, 14(4):3120–3134, 2023. doi: 10.1109/TAFFC.2022.3233324. 2

[29] Kun Zhou, Berrak Sisman, Rajib Rana, Björn W. Schuller, and Haizhou Li. Emotion intensity and its control for emotional voice conversion. *IEEE Transactions on Affective Computing*, 14(1):31–48, 2023. doi: 10.1109/TAFFC.2022.3175578.

[30] Katsuki Inoue, Sunao Hara, Masanobu Abe, Nobukatsu Hojo, and Yusuke Ijima. Model architectures to extrapolate emotional expressions in dnn-based text-to-speech. *Speech Communication*, 126:35–43, 2021. 2

[31] Yingzhu Zhao, Chongjia Ni, Cheung-Chi Leung, Shafiq R Joty, Eng Siong Chng, and Bin Ma. Speech transformer with speaker aware persistent memory. In *INTERSPEECH*, pages 1261–1265, 2020. 2

[32] David Bonet, Antonio Ortega, Javier Ruiz-Hidalgo, and Sarath Shekkizhar. Channel redundancy and overlap in convolutional neural networks with channel-wise nnk graphs. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4328–4332. IEEE, 2022. 3

[33] Jiafeng Li, Ying Wen, and Lianghua He. Scconv: Spatial and channel reconstruction convolution for feature redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6153–6162, 2023. 4

[34] Dianwen Ng, Yang Xiao, Jia Qi Yip, Zhao Yang, Biao Tian, Qiang Fu, Eng Siong Chng, and Bin Ma. Small footprint multi-channel network for keyword spotting with centroid based awareness. *Proc. INTERSPEECH 2023*, 2023.

[35] Dianwen Ng, Ruixi Zhang, Jia Qi Yip, Chong Zhang, Yukun Ma, Trung Hieu Nguyen, Chongjia Ni, Eng Siong Chng, and Bin Ma. Contrastive speech mixup for low-resource keyword spotting. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 3

[36] Jinhua Liang, Tao Zhang, and Guoqing Feng. Channel compression: Rethinking information redundancy among channels in cnn architecture. *IEEE Access*, 8:147265–147274, 2020. 3

[37] Fahim Dalvi, Hassan Sajjad, Nadir Durrani, and Yonatan Belinkov. Analyzing redundancy in pretrained transformer models. *arXiv preprint arXiv:2004.04010*, 2020. 3

[38] Jiaxiong Qiu, Cai Chen, Shuaicheng Liu, Heng-Yu Zhang, and Bing Zeng. Slimconv: Reducing channel redundancy in convolutional neural networks by features recombining. *IEEE Transactions on Image Processing*, 30:6434–6445, 2021. 3

[39] Ville Pulkki and Matti Karjalainen. *Communication acoustics: an introduction to speech, audio and psychoacoustics*. John Wiley & Sons, 2015. 3

[40] Eberhard Zwicker and Ernst Terhardt. Analytical expressions for critical-band rate and critical bandwidth as a function of frequency. *The Journal of the Acoustical Society of America*, 68(5):1523–1525, 1980.

[41] Brian C Moore and Brian R Glasberg. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *The journal of the acoustical society of America*, 74(3):750–753, 1983. 3

[42] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 01CH37221)*, volume 2, pages 749–752. IEEE, 2001. 4, 46, 102

[43] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In *2010 IEEE international conference on acoustics, speech and signal processing*, pages 4214–4217. IEEE, 2010. 4, 7, 46, 102

[44] Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *arXiv preprint arXiv:2204.02152*, 2022. 4, 47, 102

[45] Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, et al. Seamless: Multilingual expressive and streaming speech translation. *arXiv preprint arXiv:2312.05187*, 2023. 4

[46] Hyungchan Song, Sanyuan Chen, Zhuo Chen, Yu Wu, Takuya Yoshioka, Min Tang, Jong Won Shin, and Shujie Liu. Exploring wavlm on speech enhancement. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 451–457. IEEE, 2023.

[47] Daria Diatlova, Anton Udalov, Vitalii Shutov, and Egor Spirin. Adapting wavlm for speech emotion recognition. *arXiv preprint arXiv:2405.04485*, 2024.

[48] Yinlin Guo, Haofan Huang, Xi Chen, He Zhao, and Yuehai Wang. Audio deepfake detection with self-supervised wavlm and multi-fusion attentive classifier. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12702–12706. IEEE, 2024.

[49] Dianwen Ng, Chong Zhang, Ruixi Zhang, Yukun Ma, Trung Hieu Nguyen, Chongjia Ni, Shengkui Zhao, Qian Chen, Wen Wang, Eng Siong Chng, et al. Adapter-tuning with effective token-dependent representation shift for automatic speech recognition. In *Proc. INTERSPEECH 2023*, pages 1319–1323, 2023. 4

[50] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021. 4, 51, 52, 57

[51] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017. 5

[52] Dongseong Hwang, Ananya Misra, Zhouyuan Huo, Nikhil Siddhartha, Shefali Garg, David Qiu, Khe Chai Sim, Trevor Strohman, Françoise Beaufays, and Yanzhang He. Large-scale asr domain adaptation using self-and semi-supervised learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6627–6631. IEEE, 2022. 5

[53] Dianwen Ng, Chong Zhang, Ruixi Zhang, Yukun Ma, Fabian Ritter-Gutierrez, Trung Hieu Nguyen, Chongjia Ni, Shengkui Zhao, Eng Siong Chng, and Bin Ma. Are soft prompts good zero-shot learners for speech recognition? In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10366–10370. IEEE, 2024. 5

[54] Samet Oymak, Ankit Singh Rawat, Mahdi Soltanolkotabi, and Christos Thrampoulidis. On the role of attention in prompt-tuning. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023. 6, 67

[55] Sawyer Birnbaum, Volodymyr Kuleshov, Zayd Enam, Pang Wei W Koh, and Stefano Ermon. Temporal film: Capturing long-range sequence dependencies with feature-wise modulations. *Advances in Neural Information Processing Systems*, 32, 2019. 6

[56] Mehmet Ozgur Turkoglu, Alexander Becker, Hüseyin Anil Gündüz, Mina Rezaei, Bernd Bischl, Rodrigo Caye Daudt, Stefano D'Aronco, Jan Wegner, and Konrad Schindler. Film-ensemble: Probabilistic deep learning via feature-wise linear modulation. *Advances in neural information processing systems*, 35:22229–22242, 2022. 6

[57] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022. 6

[58] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. *Advances in Neural Information Processing Systems*, 36, 2024. 6, 37, 89, 92, 98, 101

[59] Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations*, 2024. 6, 35, 40, 89

[60] Lonce Wyse. Audio spectrogram representations for processing with convolutional neural networks. *arXiv preprint arXiv:1706.09559*, 2017. 10

[61] Fang Zheng, Guoliang Zhang, and Zhanjiang Song. Comparison of different implementations of mfcc. *Journal of Computer science and Technology*, 16: 582–589, 2001. 10

[62] Emiru Tsunoo, Yosuke Kashiwagi, Toshiyuki Kumakura, and Shinji Watanabe. Transformer asr with contextual block processing. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 427–433. IEEE, 2019. 12

[63] Abdelrahman Mohamed, Dmytro Okhonko, and Luke Zettlemoyer. Transformers with convolutional context for asr. *arXiv preprint arXiv:1904.11660*, 2019. 12

[64] Yuya Fujita, Aswin Shanmugam Subramanian, Motoi Omachi, and Shinji Watanabe. Attention-based asr with lightweight and dynamic convolutions. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7034–7038. IEEE, 2020. 13

[65] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International conference on machine learning*, pages 2342–2350. PMLR, 2015. 13

[66] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4945–4949. IEEE, 2016. 13

[67] Alex Graves and Alex Graves. Connectionist temporal classification. *Supervised sequence labelling with recurrent neural networks*, pages 61–93, 2012. 13

[68] Shuai Li, Wanqing Li, Chris Cook, Ce Zhu, and Yanbo Gao. Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5457–5466, 2018. 13

[69] Niko Moritz, Takaaki Hori, and Jonathan Le. Streaming automatic speech recognition with the transformer model. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6074–6078. IEEE, 2020. 13

[70] Tom Bagby, Kanishka Rao, and Khe Chai Sim. Efficient implementation of recurrent neural network transducer in tensorflow. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 506–512. IEEE, 2018. 16

[71] Jyh-Shing Roger Jang and Shiuan-Sung Lin. Optimization of viterbi beam search in speech recognition. In *International Symposium on Chinese Spoken Language Processing*, 2002. 16

[72] Hu Hu, Rui Zhao, Jinyu Li, Liang Lu, and Yifan Gong. Exploring pre-training with alignments for rnn transducer based end-to-end speech recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7079–7083. IEEE, 2020. 16

[73] Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood, and Imran Makhdoom. Automatic speech recognition: a survey. *Multimedia Tools and Applications*, 80:9411–9457, 2021. 20

[74] Rohit Prabhavalkar, Takaaki Hori, Tara N Sainath, Ralf Schlüter, and Shinji Watanabe. End-to-end speech recognition: A survey. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023. 20

[75] Zrar Kh Abdul and Abdulbasit K Al-Talabani. Mel frequency cepstral coefficient and its applications: A review. *IEEE Access*, 10:122136–122158, 2022. 20

[76] Jee-weon Jung, You Jin Kim, Hee-Soo Heo, Bong-Jin Lee, Youngki Kwon, and Joon Son Chung. Pushing the limits of raw waveform speaker recognition. *arXiv preprint arXiv:2203.08488*, 2022. 20

[77] Jee-weon Jung, Hee-Soo Heo, Ju-ho Kim, Hye-jin Shim, and Ha-Jin Yu. Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification. *arXiv preprint arXiv:1904.08104*, 2019. 20

[78] Tara N Sainath, Ron J Weiss, Andrew W Senior, Kevin W Wilson, and Oriol Vinyals. Learning the speech front-end with raw waveform cldnns. In *INTERSPEECH*, pages 1–5. Dresden, Germany, 2015. 20

[79] Alexander H Liu, Yu-An Chung, and James Glass. Non-autoregressive predictive coding for learning speech representations from local dependencies. *arXiv preprint arXiv:2011.00406*, 2020. 21

[80] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 21, 50

[81] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the association for computational linguistics*, 8:64–77, 2020. 21

[82] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2022. 21, 35, 39, 89, 100

[83] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020. 21, 50

[84] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021. 21, 50, 51

[85] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning*, pages 1298–1312. PMLR, 2022. 21

[86] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 21, 22, 24

[87] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 22

[88] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*, 2019. 24, 45

[89] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al. Espnet: End-to-end speech processing toolkit. *arXiv preprint arXiv:1804.00015*, 2018. 45

[90] Hung-yi Lee, Abdelrahman Mohamed, Shinji Watanabe, Tara Sainath, Karen Livescu, Shang-Wen Li, Shu-wen Yang, and Katrin Kirchhoff. Self-supervised representation learning for speech processing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Tutorial Abstracts*, pages 8–13, 2022. 24

[91] Douglas Steinley. K-means clustering: a half-century synthesis. *British Journal of Mathematical and Statistical Psychology*, 59(1):1–34, 2006. 24

[92] Zheli Liu, Zhendong Wu, Tong Li, Jin Li, and Chao Shen. Gmm and cnn hybrid method for short utterance speaker recognition. *IEEE Transactions on Industrial informatics*, 14(7):3244–3252, 2018. 24

[93] Samik Sadhu, Di He, Che-Wei Huang, Sri Harish Mallidi, Minhua Wu, Ariya Rastrow, Andreas Stolcke, Jasha Droppo, and Roland Maas. Wav2vec-c: A self-supervised model for speech representation learning. *arXiv preprint arXiv:2103.08393*, 2021. 25

[94] Alexei Baevski, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. Unsupervised speech recognition. *Advances in Neural Information Processing Systems*, 34:27826–27839, 2021. 25

[95] Kazuya Kawakami, Luyu Wang, Chris Dyer, Phil Blunsom, and Aaron van den Oord. Learning robust and multilingual speech representations. *arXiv preprint arXiv:2001.11128*, 2020. 25

[96] Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, et al. Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training. *arXiv preprint arXiv:2104.01027*, 2021. 50

[97] Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Paden Tomasello, Jacob Kahn, Gilad Avidov, Ronan Collobert, and Gabriel Synnaeve. Rethinking evaluation in asr: Are our models robust enough? *arXiv preprint arXiv:2010.11745*, 2020. 25

[98] Kaizhi Qian, Yang Zhang, Heting Gao, Junrui Ni, Cheng-I Lai, David Cox, Mark Hasegawa-Johnson, and Shiyu Chang. Contentvec: An improved self-supervised speech representation by disentangling speakers. In *International Conference on Machine Learning*, pages 18003–18017. PMLR, 2022. 25, 30

[99] Yiming Wang, Jinyu Li, Heming Wang, Yao Qian, Chengyi Wang, and Yu Wu. Wav2vec-switch: Contrastive learning from original-noisy speech pairs for robust speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7097–7101. IEEE, 2022. 31, 50

[100] Wenyong Huang, Zhenhe Zhang, Yu Ting Yeung, Xin Jiang, and Qun Liu. Spiral: Self-supervised perturbation-invariant representation learning for speech pre-training. *arXiv preprint arXiv:2201.10207*, 2022.

[101] Xuankai Chang, Takashi Maekaku, Yuya Fujita, and Shinji Watanabe. End-to-end integration of speech recognition, speech enhancement, and self-supervised learning representation. *arXiv preprint arXiv:2204.00540*, 2022. 25

[102] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):3048–3068, 2021. 26

[103] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 26

[104] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 28

[105] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *Advances in Neural Information Processing Systems*, 35:28708–28720, 2022. 28

[106] Andrew McCallum, Chris Pal, Gregory Druck, and Xuerui Wang. Multi-conditional learning: Generative/discriminative training for clustering and classification. In *AAAI*, volume 1, page 6, 2006. 29

[107] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 30

[108] Shizhe Hu, Zhengzheng Lou, Xiaoqiang Yan, and Yangdong Ye. A survey on information bottleneck. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 30

[109] Ofir Press, Noah A Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*, 2021. 30

[110] Katrin Tomanek, Vicky Zayats, Dirk Padfield, Kara Vaillancourt, and Fadi Biadsy. Residual adapters for parameter-efficient asr adaptation to atypical and accented speech. *arXiv preprint arXiv:2109.06952*, 2021. 33

[111] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

[112] Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv preprint arXiv:2312.12148*, 2023. 33

[113] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations*, 2021. 33, 65, 76

[114] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, 2021. 34, 65, 67

[115] Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*, 2023. 35, 116

[116] Dong Zhang, Xin Zhang, Jun Zhan, Shimin Li, Yaqian Zhou, and Xipeng Qiu. Speechgpt-gen: Scaling chain-of-information speech generation. *arXiv preprint arXiv:2401.13527*, 2024. 35, 116

[117] Sunghwan Ahn, Beom Jun Woo, Min Hyun Han, Chanyeong Moon, and Nam Soo Kim. Hilcodec: High fidelity and lightweight neural audio codec. *arXiv preprint arXiv:2405.04752*, 2024. 35

[118] Ryan Langman, Ante Jukić, Kunal Dhawan, Nithin Rao Koluguri, and Boris Ginsburg. Spectral codecs: Spectrogram-based audio codecs for high quality speech synthesis. *arXiv preprint arXiv:2406.05298*, 2024.

[119] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033, 2020. 35, 37, 41, 96, 100

[120] Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *arXiv preprint arXiv:2406.07550*, 2024. 36

[121] Zeyu Xie, Xuenan Xu, Zhizheng Wu, and Mengyue Wu. Picoaudio: Enabling precise timestamp and frequency controllability of audio events in text-to-audio generation. *arXiv preprint arXiv:2407.02869*, 2024. 36

[122] Taejun Bak, Junmo Lee, Hanbin Bae, Jinhyeok Yang, Jae-Sung Bae, and Young-Sun Joo. Avocodo: Generative adversarial network for artifact-free vocoder. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12562–12570, 2023. 37

[123] Hubert Siuzdak. Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis. *arXiv preprint arXiv:2306.00814*, 2023. 37, 41

[124] Lei Zhu, Fangyun Wei, Yanye Lu, and Dong Chen. Scaling the codebook size of vqgan to 100,000 with a utilization rate of 99%. *arXiv preprint arXiv:2406.11837*, 2024. 38

[125] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[126] Jialun Peng, Dong Liu, Songcen Xu, and Houqiang Li. Generating diverse structure for image inpainting with hierarchical vq-vae. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10775–10784, 2021. 38

[127] Dongchao Yang, Songxiang Liu, Rongjie Huang, Jinchuan Tian, Chao Weng, and Yuexian Zou. Hifi-codec: Group-residual vector quantization for high fidelity audio codec. *arXiv preprint arXiv:2305.02765*, 2023. 39, 41, 96, 100

[128] Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, et al. Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models. *arXiv preprint arXiv:2403.03100*, 2024. 39

[129] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *Transactions on Machine Learning Research*, 2022. 41, 89

[130] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 41, 96

[131] Haibin Wu, Ho-Lam Chung, Yi-Cheng Lin, Yuan-Kuei Wu, Xuanjun Chen, Yu-Chi Pai, Hsiu-Hsuan Wang, Kai-Wei Chang, Alexander H Liu, and Hung-yi Lee. Codec-superb: An in-depth analysis of sound codec models. *arXiv preprint arXiv:2402.13071*, 2024. 47, 102

[132] Andrew Hines, Jan Skoglund, Anil C Kokaram, and Naomi Harte. Visqol: an objective speech quality model. *EURASIP Journal on Audio, Speech, and Music Processing*, 2015:1–18, 2015. 47, 102

[133] Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, et al. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*, 2021. 50

[134] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. An unsupervised autoregressive model for speech representation learning. *Proc. INTERSPEECH 2019*, pages 146–150, 2019. 50

[135] Dianwen Ng, Jia Qi Yip, Tanmay Surana, Zhao Yang, Chong Zhang, Yukun Ma, Chongjia Ni, Eng Siong Chng, and Bin Ma. I2cr: Improving noise robustness on keyword spotting using inter-intra contrastive regularization. *arXiv preprint arXiv:2209.06360*, 2022. 50

[136] Wenyong Huang, Zhenhe Zhang, Yu Ting Yeung, Xin Jiang, and Qun Liu. Spiral: Self-supervised perturbation-invariant representation learning for speech pre-training. In *International Conference on Learning Representations*, 2022. 50

[137] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010. 54

[138] Archiki Prasad, Preethi Jyothi, and Rajbabu Velmurugan. An investigation of end-to-end models for robust speech recognition. In *ICASSP 2021-2021*

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6893–6897. IEEE, 2021. 56, 58, 73, 74

[139] Qiu-Shi Zhu, Jie Zhang, Zi-Qiang Zhang, Ming-Hui Wu, Xin Fang, and Li-Rong Dai. A noise-robust self-supervised pre-training model based speech representation learning for automatic speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3174–3178. IEEE, 2022. 56

[140] Frederic Font, Gerard Roma, and Xavier Serra. Freesound technical demo. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 411–412, 2013. 56, 73

[141] François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Esteve. Ted-lium 3: twice as much data and corpus repartition for experiments on speaker adaptation. In *International conference on speech and computer*, pages 198–208. Springer, 2018. 60

[142] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852, 2021. 60

[143] Zih-Ching Chen, Chin-Lun Fu, Chih-Ying Liu, Shang-Wen Daniel Li, and Hung-yi Lee. Exploring efficient-tuning methods in self-supervised speech models. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 1120–1127, 2023. doi: 10.1109/SLT54892.2023.10023274. 64, 65, 74, 85

[144] Ruchao Fan, Yunzheng Zhu, Jinhan Wang, and Abeer Alwan. Towards better domain adaptation for self-supervised models: A case study of child asr. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1242–1252, 2022.

[145] Bethan Thomas, Samuel Kessler, and Salah Karout. Efficient adapter transfer of self-supervised speech models for automatic speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7102–7106. IEEE, 2022. 64

[146] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023. 107

[147] Dianwen Ng, Ruixi Zhang, Jia Qi Yip, Zhao Yang, Jinjie Ni, Chong Zhang, Yukun Ma, Chongjia Ni, Eng Siong Chng, and Bin Ma. De'hubert: Disentangling noise in a self-supervised model for robust speech recognition. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[148] Zhao Yang, Dianwen Ng, Chong Zhang, Xiao Fu, Rui Jiang, Wei Xi, Yukun Ma, Chongjia Ni, Eng Siong Chng, Bin Ma, et al. Dual acoustic linguistic

self-supervised representation learning for cross-domain speech recognition. *Proc. INTERSPEECH 2023*, 2023. 64

[149] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2021. 64

[150] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, 2021. 64

[151] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, 2021. 65, 67

[152] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 2022. 65

[153] Xuancheng Huang, Zijun Liu, Peng Li, Tao Li, Maosong Sun, and Yang Liu. An extensible plug-and-play method for multi-aspect controllable text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15233–15256, 2023. 65

[154] Jing Qian, Li Dong, Yelong Shen, Furu Wei, and Weizhu Chen. Controllable natural language generation with contrastive prefixes. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2912–2924, 2022.

[155] Yuxuan Gu, Xiaocheng Feng, Sicheng Ma, Lingyuan Zhang, Heng Gong, and Bing Qin. A distributional lens for multi-aspect controllable text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1023–1043, 2022. 65

[156] Grady Booch, Francesco Fabiano, Lior Horesh, Kiran Kate, Jonathan Lenchner, Nick Linck, Andreas Loreggia, Keerthiram Murgesan, Nicholas Mattei, Francesca Rossi, et al. Thinking fast and slow in ai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 15042–15046, 2021. 68

[157] Katerina Zmolikova, Marc Delcroix, Tsubasa Ochiai, Keisuke Kinoshita, Jan Černockỳ, and Dong Yu. Neural target speech extraction: An overview. *IEEE Signal Processing Magazine*, 40(3):8–29, 2023. 68

[158] Zexu Pan, Marvin Borsdorf, Siqi Cai, Tanja Schultz, and Haizhou Li. Neuroheed: Neuro-steered speaker extraction using eeg signals. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024. 68

[159] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 70

[160] Abbas El Gamal and Young-Han Kim. *Network information theory*. Cambridge university press, 2011. 71

[161] Ameet Talwalkar, Sanjiv Kumar, and Henry Rowley. Large-scale manifold learning. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 71

[162] Sunit Sivasankaran, Aditya Arie Nugraha, Emmanuel Vincent, Juan A Morales-Cordovilla, Siddharth Dalmia, Irina Illina, and Antoine Liutkus. Robust asr using neural network based speech enhancement and feature simulation. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 482–489. IEEE, 2015. 71

[163] Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 993–1003, 2021. 73

[164] Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, et al. GigaSpeech: An evolving, multi-domain ASR corpus with 10,000 hours of transcribed audio. In *Proc. INTERSPEECH 2021*, pages 4376–4380, 2021.

[165] Jacob Kahn, Morgane Rivière, Weiyi Zheng, Evgeny Kharitonov, Qiantong Xu, Pierre-Emmanuel Mazaré, Julien Karadayi, Vitaliy Liptchinsky, Ronan Collobert, Christian Fuegen, et al. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673. IEEE, 2020. 73

[166] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015. 73, 107

[167] Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset.

In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 920–924. IEEE, 2021. 73

[168] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404: 132306, 2020. 74

[169] Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. Superb: Speech processing universal performance benchmark. In *Proc. INTERSPEECH 2021*, 2021. 74

[170] Emmanuel Vincent, Shinji Watanabe, Jon Barker, and Ricard Marxer. The 4th chime speech separation and recognition challenge. *URL: http://spandh. dcs. shef. ac. uk/chime_challenge/(last accessed on 1 August, 2018)*, 2016. 76

[171] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 80, 109

[172] Kenichi Fujita, Hiroshi Sato, Takanori Ashihara, Hiroki Kanagawa, Marc Delcroix, Takafumi Moriya, and Yusuke Ijima. Noise-robust zero-shot text-to-speech synthesis conditioned on self-supervised speech-representation model with adapters. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11471–11475. IEEE, 2024. 82

[173] Zhihao Du, Shiliang Zhang, Kai Hu, and Siqi Zheng. Funcodec: A fundamental, reproducible and integrable open-source toolkit for neural speech codec. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 591–595. IEEE, 2024. 89

[174] Yi-Chiao Wu, Israel D Gebru, Dejan Marković, and Alexander Richard. Audiodec: An open-source streaming high-fidelity neural audio codec. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[175] Jia Qi Yip, Shengkui Zhao, Dianwen Ng, Eng Siong Chng, and Bin Ma. Towards audio codec-based speech separation. *arXiv preprint arXiv:2406.12434*, 2024. 89

[176] Qian Chen, Yunfei Chu, Zhifu Gao, Zerui Li, Kai Hu, Xiaohuan Zhou, Jin Xu, Ziyang Ma, Wen Wang, Siqi Zheng, et al. Lauragpt: Listen, attend, understand, and regenerate audio with gpt. *arXiv preprint arXiv:2310.04673*, 2023. 89

[177] Dongchao Yang, Haohan Guo, Yuanyuan Wang, Rongjie Huang, Xiang Li, Xu Tan, Xixin Wu, and Helen Meng. Uniaudio 1.5: Large language model-driven audio codec is a few-shot audio task learner. *arXiv preprint arXiv:2406.10056*, 2024.

[178] Yixuan Zhou, Xiaoyu Qin, Zeyu Jin, Shuoyi Zhou, Shun Lei, Songtao Zhou, Zhiyong Wu, and Jia Jia. Voxinstruct: Expressive human instruction-to-speech generation with unified multilingual codec language modelling. *arXiv preprint arXiv:2408.15676*, 2024. 89

[179] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018. 89

[180] Marco Tagliasacchi, Yunpeng Li, Karolis Misiunas, and Dominik Roblek. Seanet: A multi-modal speech enhancement network. *arXiv preprint arXiv:2009.02095*, 2020. 89

[181] A Vasuki and PT Vanathi. A review of vector quantization techniques. *IEEE Potentials*, 25(4):39–47, 2006. 89

[182] Yunkee Chae, Woosung Choi, Yuhta Takida, Junghyun Koo, Yukara Ikemiya, Zhi Zhong, Kin Wai Cheuk, Marco A Martínez-Ramírez, Kyogu Lee, Wei-Hsiang Liao, et al. Vrvq: Variable bitrate residual vector quantization for audio compression. *arXiv preprint arXiv:2410.06016*, 2024. 90

[183] Robin San Roman, Yossi Adi, Antoine Deleforge, Romain Serizel, Gabriel Synnaeve, and Alexandre Défossez. From discrete tokens to high-fidelity audio using multi-band diffusion. *Advances in Neural Information Processing Systems*, 36, 2024. 93

[184] Darius Petermann, Inseon Jang, and Minje Kim. Native multi-band audio coding within hyper-autoencoded reconstruction propagation networks. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 93

[185] Eberhard Zwicker and Hugo Fastl. *Psychoacoustics: Facts and models*, volume 22. Springer Science & Business Media, 2013. 94

[186] Karlheinz Brandenburg and Marina Bosi. Overview of mpeg audio: Current and future standards for low bit-rate audio coding. *Journal of the Audio Engineering Society*, 45(1/2):4–21, 1997. 94

[187] James D Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on selected areas in communications*, 6(2):314–323, 1988. 94

[188] Brian CJ Moore. Basic auditory processes involved in the analysis of speech sounds. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 363(1493):947–963, 2008. 95

[189] William W Hager. Lipschitz continuity for constrained processes. *SIAM Journal on Control and Optimization*, 17(3):321–338, 1979. 95

[190] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 96

[191] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002. 96

[192] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003. 96

[193] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017. 96

[194] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024. 97

[195] Weihao Yu and Xinchao Wang. Mambaout: Do we really need mamba for vision? *arXiv preprint arXiv:2405.07992*, 2024.

[196] Dongchen Han, Ziyi Wang, Zhuofan Xia, Yizeng Han, Yifan Pu, Chunjiang Ge, Jun Song, Shiji Song, Bo Zheng, and Gao Huang. Demystify mamba in vision: A linear attention perspective. *arXiv preprint arXiv:2405.16605*, 2024. 97

[197] Liu Ziyin, Tilman Hartwig, and Masahito Ueda. Neural networks fail to learn periodic functions and how to fix it. *Advances in Neural Information Processing Systems*, 33:1583–1594, 2020. 98

[198] Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Long-form music generation with latent diffusion. *arXiv preprint arXiv:2404.10301*, 2024. 98

[199] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017. 100

[200] Kundan Kumar, Rithesh Kumar, Thibault De Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre De Brebisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32, 2019. 100

[201] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 100

[202] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 100

[203] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019. 101

[204] Julius Richter, Yi-Chiao Wu, Steven Krenn, Simon Welker, Bunlong Lay, Shinji Watanabe, Alexander Richard, and Timo Gerkmann. Ears: An anechoic fullband speech dataset benchmarked for speech enhancement and dereverberation. *arXiv preprint arXiv:2406.06185*, 2024. 101

[205] Igor André Pegoraro Santana, Fabio Pinhelli, Juliano Donini, Leonardo Catharin, Rafael Biazus Mangolin, Valéria Delisandra Feltrim, Marcos Aurélio Domingues, et al. Music4all: A new music database and its applications. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 399–404. IEEE, 2020. 101

[206] Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018, 2015. 101

[207] Leigh D Alsteris and Kuldip K Paliwal. Short-time phase spectrum in speech processing: A review and some experimental results. *Digital signal processing*, 17(3):578–616, 2007. 102

[208] Robert Kubichek. Mel-cepstral distance measure for objective speech quality assessment. In *Proceedings of IEEE pacific rim conference on communications computers and signal processing*, volume 1, pages 125–128. IEEE, 1993. 102

[209] Yannick Jadoul, Bill Thompson, and Bart De Boer. Introducing parselmouth: A python interface to praat. *Journal of Phonetics*, 71:1–15, 2018. 102

[210] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42:335–359, 2008. 103

[211] Bob L Sturm. The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*, 2013. 103

[212] LAION-AI. BBC Sound Effects Library data card. `https://github.com/LAION-AI/audio-dataset/blob/main/data_card/BBC.md`, 2022. Accessed: [October, 7, 2024]. 103

[213] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*, 2017. 109

[214] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*, 2023. 116

[215] Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024. 116

[216] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 116

[217] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[218] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[219] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 116

[220] Yanming Liu, Xinyue Peng, Jiannan Cao, Shi Bo, Yanxin Shen, Xuhong Zhang, Sheng Cheng, Xun Wang, Jianwei Yin, and Tianyu Du. Bridging context gaps: Leveraging coreference resolution for long contextual understanding. *arXiv preprint arXiv:2410.01671*, 2024. 117

[221] Egor Lakomkin, Chunyang Wu, Yassir Fathullah, Ozlem Kalinli, Michael L Seltzer, and Christian Fuegen. End-to-end speech recognition contextualization with large language models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12406–12410. IEEE, 2024. 117

[222] Jinwen He, Yujia Gong, Zijin Lin, Yue Zhao, Kai Chen, et al. Llm factoscope: Uncovering llms' factual discernment through measuring inner states. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10218–10230, 2024. 117

[223] Philippe Laban, Wojciech Kryściński, Divyansh Agarwal, Alexander Richard Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. Summedits: measuring llm ability at factual reasoning through the lens of summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9662–9676, 2023.

[224] Liyan Xu, Zhenlin Su, Mo Yu, Jin Xu, Jinho D Choi, Jie Zhou, and Fei Liu. Identifying factual inconsistencies in summaries: Grounding llm inference via task taxonomy. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14626–14641, 2024. 117

[225] Guan-Ting Lin, Prashanth Gurunath Shivakumar, Ankur Gandhe, Chao-Han Huck Yang, Yile Gu, Shalini Ghosh, Andreas Stolcke, Hung-yi Lee, and Ivan Bulyko. Paralinguistics-enhanced large language modeling of spoken dialogue. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10316–10320. IEEE, 2024. 117

[226] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020. 118

[227] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR, 2020. 118

[228] Xiaoze Liu, Feijie Wu, Tianyang Xu, Zhuo Chen, Yichi Zhang, Xiaoqian Wang, and Jing Gao. Evaluating the factuality of large language models using large-scale knowledge graphs. *arXiv preprint arXiv:2404.00942*, 2024. 118

[229] Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, et al. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity. *arXiv preprint arXiv:2310.07521*, 2023. 118

[230] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Ren Lu, Thomas Mesnard, Johan Ferret, Colton Bishop, Ethan Hall, Victor Carbune, and Abhinav Rastogi. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2411.13410*, 2023. 118

[231] Shreyas Chaudhari, Pranjal Aggarwal, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, Karthik Narasimhan, Ameet Deshpande, and Bruno Castro da Silva. Rlhf deciphered: A critical analysis of reinforcement learning from human feedback for llms. *arXiv preprint arXiv:2404.08555*, 2024. 118

[232] Guan-Ting Lin, Cheng-Han Chiang, and Hung-yi Lee. Advancing large language models to capture varied speaking styles and respond properly in spoken conversations. *arXiv preprint arXiv:2402.12786*, 2024. 119