

# SECodec: Structural Entropy-based Compressive Speech Representation Codec for Speech Language Models

Linqin Wang<sup>1,2</sup>, Yaping Liu<sup>1,2</sup>, Zhengtao Yu<sup>1,2</sup>\*, Shengxiang Gao<sup>1,2</sup>, Cunli Mao<sup>1,2</sup>,  
Yuxin Huang<sup>1,2</sup>, Wenjun Wang<sup>1,2</sup>, Ling Dong<sup>1,2</sup>

<sup>1</sup> Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China

<sup>2</sup> Yunnan Key Laboratory of Artificial Intelligence, Kunming, China

linqinwang7767@163.com, {ztyu, gaoshengxiang.yn}@hotmail.com,

{maocunli, huangyuxin2004}@163.com, {20203104003, 20232204183, ling.dong}@kust.edu.cn

## Abstract

With the rapid advancement of large language models (LLMs), discrete speech representations have become crucial for integrating speech into LLMs. Existing methods for speech representation discretization rely on a predefined codebook size and Euclidean distance-based quantization. However, 1) the size of codebook is a critical parameter that affects both codec performance and downstream task training efficiency. 2) The Euclidean distance-based quantization may lead to audio distortion when the size of the codebook is controlled within a reasonable range. In fact, in the field of information compression, structural information and entropy guidance are crucial, but previous methods have largely overlooked these factors. Therefore, we address the above issues from an information-theoretic perspective, we present SECodec, a novel speech representation codec based on structural entropy (SE) for building speech language models. Specifically, we first model speech as a graph, clustering the speech features nodes within the graph and extracting the corresponding codebook by hierarchically and disentangledly minimizing 2D SE. Then, to address the issue of audio distortion, we propose a new quantization method. This method still adheres to the 2D SE minimization principle, adaptively selecting the most suitable token corresponding to the cluster for each incoming original speech node. Furthermore, we develop a Structural Entropy-based Speech Language Model (SESLM) that leverages SECodec. Experimental results demonstrate that SECodec performs comparably to EnCodec in speech reconstruction, and SESLM surpasses VALL-E in zero-shot text-to-speech tasks.

**Code** — <https://github.com/wlq2019/SECodec>

## 1 Introduction

Large language models (LLMs) (Achiam et al. 2023; Touvron et al. 2023) have exhibited exceptional capabilities in a wide range of natural language processing tasks. This success has spurred extensive research efforts in developing speech language models (Zhang et al. 2024; Huang, Meng, and Ko 2023; Borsos et al. 2023), leading to notable advancements in numerous speech processing applications (Wang et al. 2023;

Tu et al. 2024; Rubenstein et al. 2023; Dong et al. 2024; Tu et al. 2023). To bridge the gap between continuous speech and token-based language models, a crucial method called speech discretization is employed. This process transforms an audio signal into a finite set of tokens. By converting speech into discrete tokens, language models are able to predict future semantic content and generate coherent and realistic speech with long-term consistency (Nguyen, Sagot, and Dupoux 2022; Tu et al. 2022).

Current discrete speech representations for speech language models can be categorized into three types: semantic tokens, acoustic tokens, and hybrid/unified tokens (Borsos et al. 2023; Zhang et al. 2024). 1) Semantic tokens (Hsu et al. 2021; Baevski et al. 2020; Chung et al. 2021) are typically generated from self-supervised pre-trained models using masked language modeling as the training objective, which are derived through  $k$ -means clustering on representations from a specific intermediate layer, resulting in sequences with a one-dimensional structure. Speech language models that use semantic tokens (Lakhotia et al. 2021; Zhang et al. 2023; Hassid et al. 2023) can be externally connected to a vocoder for speech synthesis. While these models effectively capture semantically accurate content, the resulting speech generation often suffers from poor quality and a loss of acoustic details. 2) Acoustic tokens (Zeghidour et al. 2021; Défossez et al. 2024; Yang et al. 2023; Du et al. 2024) are extracted from neural audio codecs, which use reconstruction as the training objective. By employing residual vector quantization (RVQ) (Gray 1984; Vasuki and Vanathi 2006) with hierarchical quantizers for discretization, acoustic tokens are represented as matrices with two dimensions: timesteps and quantizers. VALL-E (Wang et al. 2023) is a representative model of speech language models that utilize acoustic tokens. Despite achieving impressive zero-shot text-to-speech (TTS) capabilities, it still faces issues such as inaccurate content, stemming from the complex information contained within acoustic tokens. 3) Hybrid or unified tokens (Borsos et al. 2023; Zhang et al. 2024) employ different strategies to combine semantic tokens and acoustic tokens. Hybrid tokens adopt a hierarchical approach, encompassing both semantic token language models and acoustic token language models, to capture content information and acoustic details, respectively (Dong et al. 2024; Borsos et al. 2023; Rubenstein et al.

\* Corresponding author

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

2023). Recently, unified tokens, exemplified by SpeechTokenizer (Zhang et al. 2024), have emerged. These tokens distill semantic information into acoustic tokens, effectively unifying semantic and acoustic representations. SpeechTokenizer has achieved superior results in downstream tasks such as speech synthesis. The ideal speech representation for speech language models should meet two key characteristics: i) Effective preservation of speech information; ii) Sufficient compressiveness for efficient training of speech language models. However, i) existing speech discretization methods rely on  $k$ -means to initialize the codebook space, the size of codebook is a critical parameter that significantly impacts the performance of the codec and the training efficiency of downstream tasks, yet its size is typically determined through empirical judgment. ii) Additionally, when attempting to control the size of the codebook within a reasonable range, the quantization process, which relies on Euclidean distance, may lead to substantial differences between codebook’s vector and original vector, resulting in audio distortion. These issues result in a loss of information and produce overly long tokens that are difficult to train, thereby impairing overall performance.

In this work, we address the aforementioned issues from an information-theoretic perspective, drawing inspiration from structural entropy (SE) (Li and Pan 2016; Cao et al. 2024b), a metric that assesses the amount of information contained in a graph (Yang et al. 2024a; Zeng et al. 2024; Yang et al. 2024d,c; Cao et al. 2024a; Peng et al. 2024; Zou et al. 2024). We present SECodec, a novel speech representation codec tokenizer based on structural entropy, which can automatically determine the appropriate codebook size and integrate structural information into the quantization process. Experiments demonstrate that these approaches effectively mitigate the information loss problem prevalent in existing speech discretization methods. Our main contributions are:

- We model the speech representation codec from an information-theoretic perspective. Compared to previous methods that use  $k$ -means, the proposed SECodec, by introducing structural information and entropy guidance, learns a more compressive and informative codebook without requiring a predetermined codebook size. To the best of our knowledge, we are the first to apply structural entropy (SE) minimization for a speech representation codec.
- To address the issue of audio distortion when controlling codebook size, we propose a new quantization method that iteratively selects appropriate clusters for the added original speech features using a SE heuristic function. This approach significantly enhances the quality of information in the speech tokens.
- Extensive experimental results demonstrate that SECodec performs comparably to EnCodec in speech reconstruction, while SESLM surpasses VALL-E in zero-shot text-to-speech tasks on a multi-speaker benchmark dataset.

## 2 Preliminary

Structural entropy (SE) (Li and Pan 2016) is defined as the minimum number of bits to encode the vertex that is acces-

sible with a step of random walk on a graph. SE is a measurement of graph complexity by encoding tree structures via characterizing the uncertainty of the hierarchical topology of graphs. The structural entropy of graph  $G$  is defined on an associated encoding tree  $\mathcal{T}$ , revealing the amount of uncertainty that remained in  $G$  after encoded by  $\mathcal{T}$ . Through structural entropy minimization, the optimized hierarchical clustering result of vertices in  $G$  is retained by  $\mathcal{T}$ . We present the formal definitions of encoding tree and SE as follows.

**Definition 1.** Let  $G = (V, E, W)$  be an undirected weighted graph, where  $V = \{v_1, \dots, v_n\}$  is the vertex set,  $E$  is the edge set, and  $W \in R^{n \times n}$  is the edge weight matrix.

1) The encoding tree  $\mathcal{T}$  of  $G$  is a hierarchical rooted tree where each tree node  $\alpha$  associates with a vertex set  $T_\alpha$ .

2) The root node  $\lambda$  of  $\mathcal{T}$  associates with  $T_\lambda = V$  and each leaf node  $v$  associates with  $T_v$  containing a vertex in  $V$ .

3) For each non-leaf node  $\alpha \in \mathcal{T}$ , the successors of  $\alpha$  are associated with disjoint vertex subsets, and the union of these subsets is  $T_\alpha$ .

**Definition 2.** The structural entropy of  $G$  given by  $\mathcal{T}$  is defined as follows:

$$\mathcal{H}^{\mathcal{T}}(G) = \sum_{\alpha \in \mathcal{T}, \alpha \neq \lambda} \mathcal{H}^{\mathcal{T}}(G; \alpha) = \sum_{\alpha \in \mathcal{T}, \alpha \neq \lambda} -\frac{g_\alpha}{\mathcal{V}_G} \log_2 \frac{\mathcal{V}_\alpha}{\mathcal{V}_{\alpha^-}}, \quad (1)$$

where  $\mathcal{H}^{\mathcal{T}}(G; \alpha)$  is the assigned structural entropy of  $\alpha$ ,  $g_\alpha$  is the cut, i.e., the sum of edge weights between vertices in and not in  $T_\alpha$ ,  $\mathcal{V}_\alpha$  and  $\mathcal{V}_G$  are the volumes, i.e., the sum of vertex degrees in  $T_\alpha$  and  $G$ , respectively. The structural entropy of  $G$  is defined as  $\mathcal{H}(G) = \min_{\mathcal{T}} \{\mathcal{H}^{\mathcal{T}}(G)\}$ , where  $\mathcal{T}$  ranges over all possible encoding trees. The vertex sets associated with tree nodes form a clustering of vertices in  $V$ .

**Definition 3.** The  $K$ -D structural entropy is the structural entropy given by the encoding trees with the height of at most  $K$ . When  $K = 2$ , the encoding tree represents graph partitioning, which can be used to perform partitioning clustering. A 2-D encoding tree  $\mathcal{T}$  can be formulated as a graph partitioning  $\mathcal{P} = \{\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_L\}$  of  $V$ , where  $\mathcal{X}_i$  is a vertex subset called module associated with the  $i$ -th children of root  $\lambda$ . The structural entropy of  $G$  given by  $\mathcal{P}$  is defined as:

$$\mathcal{H}^{\mathcal{P}}(G) = - \sum_{\mathcal{X} \in \mathcal{P}} \sum_{v_i \in \mathcal{X}} \frac{g_i}{\mathcal{V}_G} \log_2 \frac{d_i}{\mathcal{V}_{\mathcal{X}}} - \sum_{\mathcal{X} \in \mathcal{P}} \frac{g_{\mathcal{X}}}{\mathcal{V}_G} \log_2 \frac{\mathcal{V}_{\mathcal{X}}}{\mathcal{V}_G}, \quad (2)$$

where  $d_i$  is the degree of vertex  $v_i$ ,  $g_i$  is the cut, i.e., the sum of edge weights connecting  $v_i$  and other vertices,  $\mathcal{V}_{\mathcal{X}}$  and  $\mathcal{V}_G$  are the volumes, i.e., the sum of vertex degrees in module  $\mathcal{X}$  and graph  $G$ , respectively, and  $g_{\mathcal{X}}$  is the cut, i.e., the sum of edge weights between vertices in and not in module  $\mathcal{X}$ .

## 3 The Method

Figure 1 presents an overview of SECodec. Our model is based on the RVQ-GANs (Du et al. 2024) framework, akin to

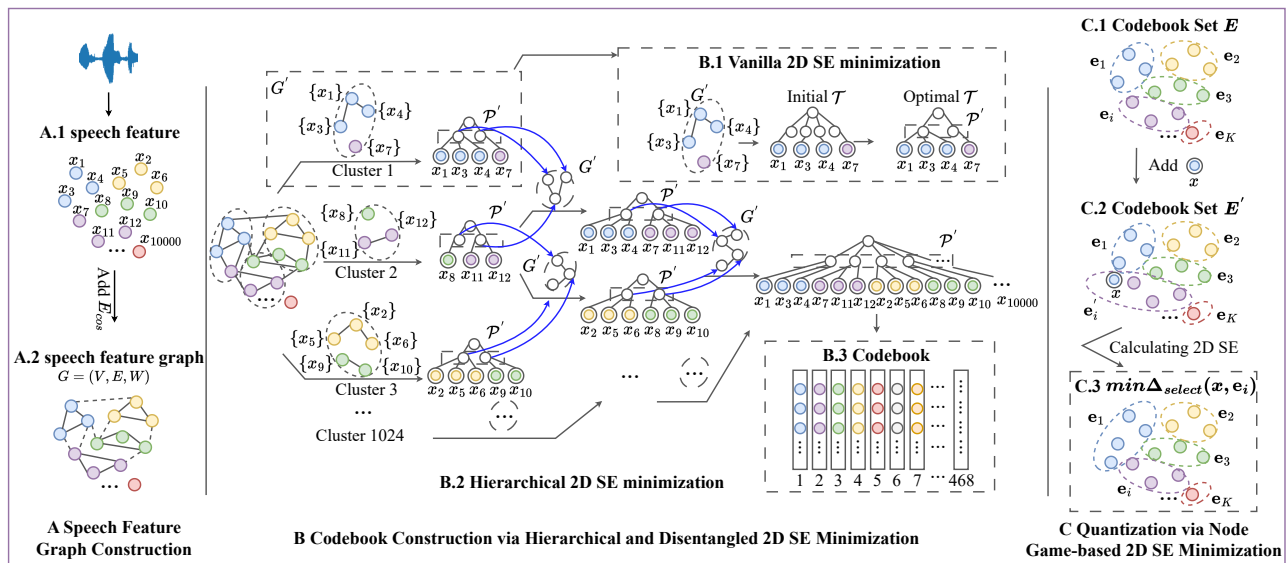


Figure 1: Framework of proposed model.

SoundStream (Zeghidour et al. 2021) and EnCodec (Défossez et al. 2024). However, we employ 2D structural entropy to optimize both the codebook initialization and the quantization process, resulting in more compressive codebook and more informative tokens. We begin by formalizing the task. Subsequently, we propose a novel structural entropy-based approach for codebook construction. We then present our informative quantization process. Finally, we introduce the training objective and design the SESLM.

### 3.1 Problem Formalization

Considering the input speech feature  $X = [x_1, \dots, x_T] \in R^{H \times T}$  from a pre-trained convolutional network, where  $H$  is the dimension of the speech representation and  $T$  is the length of the sequence. we construct a speech feature graph  $G = (V, E, W)$ . Here  $V = \{v_1, v_2, \dots, v_n\}$  is the set of vertices corresponding to speech features in  $X$ ,  $E$  represents the set of edges connecting the vertices, and  $W$  represents the set of edge weights measuring the similarities between every frame of speech feature. For two frames of speech feature  $x_i, x_j \in X$ , we measure their cosine similarity. Partitioning  $G$  results in  $\{e_1, \dots, e_i, \dots, e_j, \dots, e_K\}$ ,  $e_i \subset V, e_i \cap e_j = \emptyset$ , which represents a partition of  $V$  containing  $K$  clusters (sets) of speech features. These clusters correspond to the codebook  $\mathcal{E}_{codebook} = [e_1, \dots, e_K]$ .

### 3.2 Codebook Construction via Hierarchical and Disentangled 2D SE Minimization

Speech feature graph partitioning decodes  $G$  into  $\mathcal{P}$ , which defines the size of the codebook in the form of speech feature clusters. A faithful decoding of the speech feature correlations in  $G$  assigns related speech features to the same cluster and unrelated ones to different clusters. Previous RVQ-based speech codec methods use  $k$ -means to initialize the codebook space. These empirically defined codebooks, which must be

predetermined, lead to a loss of information and result in overly long tokens that are difficult to train, consequently impairing overall performance. To address this issue, SECodec conducts codebook partitioning under the guidance of 2D structural entropy (SE) minimization. This approach reveals the essential second-order (cluster-wise) structure inherent in the raw graph without prior knowledge of the number of speech feature clusters.

Li and Pan (2016) propose a vanilla greedy 2D structural entropy (SE) minimization algorithm that repeatedly merges any two nodes in the encoding tree  $\mathcal{T}$  resulting in the largest decrease in 2D SE until reaching the minimum possible value. This process partitions a graph without supervision or a predetermined total number of clusters. However, this vanilla 2D SE minimization algorithm has a time complexity of  $O(|V|^3)$ , making it prohibitively slow for large and complex graphs. Furthermore, the ultimate goal of our clustering is to construct codebook. The column vectors in the codebook need to be spatially distributed as far apart as possible and avoid overlapping to ensure effective representation and diversity of the speech features. To address these challenges, we propose to minimize 2D SE for construct codebook in a hierarchical and disentangled manner, shown in Algorithm 1. Specifically, each speech feature  $x_1, \dots, x_T$  is placed in its own cluster (line 1). These clusters are then divided into subsets of size  $n$  (line 3-5), and within each subset, the vanilla greedy algorithm is used to merge the clusters into new ones (lines 6-16). The newly formed clusters proceed to the next iteration (line 17). This iterative process continues until all speech feature clusters are considered simultaneously (lines 18-19). If no clusters within a subset can be merged at any point, the subset size  $n$  is increased to allow more clusters to be considered together for potential merging (lines 20-21). Finally, we extract the corresponding codebook by minimizing the mutual information between each vector within the codebook (lines 22-34). Figure 1A shows the speech feature graph construc-

tion on nodes  $x_1$  to  $x_{10000}$ . Figure 1B illustrates codebook construction process: initially  $x_1$  to  $x_{10000}$  are in separate clusters. Clusters of size  $n = 1024$  are considered at a time to form a subgraph  $G'$ . Clusters in each  $G'$  are merged using the vanilla 2D SE minimization to form  $\mathcal{P}'$  (Figure 1B.1). The partitions from the previous iteration are carried over to the next, as shown by the blue curved arrows in Figure 1B.2. The process concludes when a  $\mathcal{P}'$  that encompasses all the speech features is achieved. To further enhance the codebooks' ability to represent the diversity of speech, we introduce a mutual information learning algorithm to disentangle the central features of each cluster in  $\mathcal{P}'$ . For the disentanglement between  $\mathbf{e}_i$  and  $\mathbf{e}_j$ . The variational contrastive log-ratio upper bound (vCLUB) (Cheng et al. 2020) is used to compute the upper bound of mutual information (MI) for irrelevant information of the  $\mathbf{e}$ , decreasing the correlation among different clusters' representation:

$$\hat{\mathcal{I}}(\mathbf{e}_i, \mathbf{e}_j) = \frac{1}{N^2} \sum_{\mathcal{M}=1}^N \sum_{\mathcal{J}=1}^N [\log f_\psi(\mathbf{e}_{i\mathcal{M}}|\mathbf{e}_{j\mathcal{M}}) - \log f_\psi(\mathbf{e}_{j\mathcal{J}}|\mathbf{e}_{i\mathcal{M}})], \quad (3)$$

where  $\{\mathbf{e}_i, \mathbf{e}_j\} \in \mathbf{e}$ ,  $\mathcal{N}$  represents the samples from  $\mathbf{e}_i$  and  $\mathbf{e}_j$ .  $f_\psi(\mathbf{e}_i|\mathbf{e}_j)$  is a variational distribution with parameter  $\psi$  to approximate  $f(\mathbf{e}_i|\mathbf{e}_j)$ .  $\hat{\mathcal{I}}$  is the unbiased estimator for vCLUB with samples  $\{\mathbf{e}_{i\mathcal{M}}, \mathbf{e}_{j\mathcal{J}}\}$ . The indexes  $\mathcal{M}$  and  $\mathcal{J}$  are the samples of  $\mathbf{e}_i$  and  $\mathbf{e}_j$ . By minimizing Eq. 3, we can decrease the correlation among accent features  $\mathbf{e}_i$  and speech features  $\mathbf{e}_j$ . Finally, the central features of each cluster in  $\mathcal{P}'$  are concatenated to form the codebook  $\mathcal{E}_{codebook}$  columns in Figure 1B.3. In summary, SECodec constructs a compressive codebook from complex speech feature graphs in an unsupervised and disentangled manner.

### 3.3 Quantization via Node Game-based 2D SE Minimization

The quantization of previous RVQ-based speech codec methods is to compare the input vectors with the vectors in the codebook and extract the indexes of the most similar vectors (Défossez et al. 2024; Zeghidour et al. 2021). However, the comparison is performed by simply calculating the Euclidean distance between vectors. In high-dimensional space, the Euclidean distance tends to be uniformly distributed, which can distort the results and affect the quality of the quantized tokens. Consequently, the tokens obtained after quantization may lack sufficient real information when the size of the codebook is controlled within a reasonable range. To address this issue, SECodec views quantization as a process where graph nodes dynamically categorize subgraphs (Zeng, Peng, and Li 2024; Yang et al. 2024b; Sun et al. 2024) as shown in Figure 1C.1, treating new input features as added nodes in Figure 1C.2. In this approach, added nodes iteratively select the appropriate clusters through a structural entropy heuristic function, the selected clusters then correspond to the speech tokens as illustrated in Figure 1C.3. Specifically, given a speech feature graph and its corresponding set of codebooks  $E = [\mathbf{e}_1, \dots, \mathbf{e}_i, \dots, \mathbf{e}_K]$ . The added speech feature node  $x$  selects the current codebook  $\mathbf{e}_i$ , changing the

codebook set to  $E' = [\mathbf{e}_1, \dots, \mathbf{e}'_i, \dots, \mathbf{e}_K, \{x\}]$  ( $\mathbf{e}_i = \mathbf{e}'_i \cup \{x\}$ ). At this point, the change of the speech feature graph's 2D SE is formalized as:

$$\begin{aligned} \Delta_{select}(x, \mathbf{e}_i) &= \mathcal{H}^T(G) - \mathcal{H}^{\mathcal{T}'}(G) \\ &= \sum_{n=1}^{|E|} H^{(2)}(\mathbf{e}_n) - \sum_{n=1}^{|E'|} H^{(2)}(\mathbf{e}'_n) \\ &= -\frac{g_{\mathbf{e}_i}}{\mathcal{V}_G} \log \frac{\mathcal{V}_{\mathbf{e}_i}}{\mathcal{V}_G} + \frac{g_{\mathbf{e}'_i}}{\mathcal{V}_G} \log \frac{\mathcal{V}_{\mathbf{e}'_i}}{\mathcal{V}_G} \\ &\quad - \frac{g_{\mathbf{e}'_i}}{\mathcal{V}_G} \log \frac{\mathcal{V}_{\mathbf{e}'_i}}{\mathcal{V}_{\mathbf{e}_i}} - \frac{d_x}{\mathcal{V}_G} \log \frac{\mathcal{V}_G}{\mathcal{V}_{\mathbf{e}_i}}, \end{aligned} \quad (4)$$

where  $\Delta_{select}(x, \mathbf{e}_i)$  represents the change of the 2D SE when node  $x$  selects cluster  $\mathbf{e}_i$ , and  $\mathcal{T}'$  denotes the encoding tree corresponding to the codebooks set  $\mathcal{E}'$ .  $\mathcal{H}^{\mathcal{T}'}(G)$  and  $\mathcal{H}^T(G)$  represent the 2D SE of the graph under  $E$  and  $E'$ , respectively.  $\mathcal{V}_G$ ,  $\mathcal{V}_{\mathbf{e}_i}$ , and  $\mathcal{V}_{\mathbf{e}'_i}$  denote the volumes of the graph, cluster  $\mathbf{e}_i$ , and cluster  $\mathbf{e}'_i$ , respectively.  $g_{\mathbf{e}_i}$  and  $g_{\mathbf{e}'_i}$  correspond to the total cut edge weights of  $\mathbf{e}_i$  and  $\mathbf{e}'_i$ , respectively. The added node only selects and joins the codebooks cluster with the smallest change value of 2D SE, which is formalized as:

$$t = \text{Min}(\Delta_{select}(x, \mathbf{e}_i)), \quad (5)$$

where  $t$  represents the index of the target cluster, and the  $\text{Min}$  operation finds the codebooks index corresponding to the smallest 2D SE change value.

### 3.4 Training Objective

In terms of training objective, we follow the setup of (Du et al. 2024). The training objective consists of three components: reconstruction loss terms, adversarial loss terms, and RVQ commit losses. In the time domain, the L1 distance between the original speech and the reconstructed speech is minimized. In the frequency domain, both L1 and L2 distances are minimized across multiple Mel and magnitude spectra. For adversarial losses, SECodec incorporates several discriminators, including a multi-scale discriminator (MSD), a multi-period discriminator (MPD), and a multi-scale STFT-based (MSTFTD) discriminator.

### 3.5 SESLM

We build a structural entropy-based speech language model upon SECodec. Consisting of autoregressive and non-autoregressive models, it can hierarchically model information in speech. Compared to VALL-E (Wang et al. 2023), we include the input of speech tokens in the autoregressive model part. We believe that the speech tokens extracted by SECodec are richer in speech information, which benefits speech language model training. The model learns to perform conditional generation of a neural code sequence, denoted as  $\mathcal{O}$ , based on two input prompts: textual prompt  $u$  and acoustic prompt  $\mathcal{S}$ . The training objective is formulated as:

$$\begin{aligned} \mathcal{L}_{AR} &= - \sum_{t=1}^N \log P(\mathcal{O}_{t,1}|u, \mathcal{S}, \mathcal{O}_{<t,1}; \theta_{AR}), \\ \mathcal{L}_{NAR} &= - \sum_{l=2}^8 \log P(\mathcal{O}_{:,l}|u, \mathcal{S}, \mathcal{O}_{:,<l}; \theta_{NAR}), \end{aligned} \quad (6)$$

Algorithm 1: Codebook construction via hierarchical and disentangled 2D SE minimization.

---

**Input:** Speech feature graph  $G = (V, E, W)$ , sub-graph size  $n$ .

**Output:**  $\mathcal{E}_{codebook} = [\mathbf{e}_1, \dots, \mathbf{e}_K]$ ;

- 1  $\mathcal{P} \leftarrow (x|x \in V)$
- 2 **while** *True* **do**
- 3    $\{\mathcal{P}_s\} \leftarrow$  consecutively remove the first,
- 4    $\min(n, \text{size of the remaining part of } \mathcal{P})$  clusters,
- 5   from  $\mathcal{P}$  that form a set  $\mathcal{P}_s$ ;
- 6   **for**  $\mathcal{P} \in \{\mathcal{P}_s\}$  **do**
- 7      $V' \leftarrow$  combine all the clusters in  $\mathcal{P}_s$ ;
- 8      $E' \leftarrow \{e \in E, \text{ both endpoints of } e \in V'\}$ ;
- 9      $G' \leftarrow (V', E')$ ;
- 10     $\mathcal{T}' \leftarrow$  add a root tree node  $\lambda$ ;
- 11    **for** cluster  $\mathcal{C} \in \mathcal{P}_s$  **do**
- 12     Add a tree node  $\alpha$  to  $\mathcal{T}'$ ,  $\alpha^- = \lambda, \mathcal{T}_\alpha = \mathcal{C}$ ;
- 13     **for** speech feature  $x \in \mathcal{C}$  **do**
- 14      Add a tree node  $\gamma$  to  $\mathcal{T}'$ ;
- 15       $\gamma^- = \alpha, \mathcal{T}_\gamma = \{x\}$ ;
- 16      $\mathcal{P}' \leftarrow$  run vanilla 2D SE minimization;
- 17     Append  $\mathcal{P}'$  to  $\mathcal{P}$ ;
- 18    **if**  $|\{V'\}| = 1$  **then**
- 19     Break;
- 20    **if**  $\mathcal{P}$  is the same as at the end of last iteration **then**
- 21      $n \leftarrow 2n$ ;
- 22     $\mathcal{P} = \{\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_K\}$
- 23     $\mathbf{e} \leftarrow []$
- 24    **for**  $\mathcal{X}_i \in \mathcal{P}$  **do**
- 25      $\mathbf{e}_i \leftarrow 0$ , count  $\leftarrow 0$ ;
- 26     **for**  $x \in \mathcal{X}_i$  **do**
- 27       $\mathbf{e}_i \leftarrow \mathbf{e}_i + x$ , count  $\leftarrow$  count + 1;
- 28     Append  $\mathbf{e}_i$  to  $\mathbf{e}$ ;
- 29     $\mathbf{e} = [\mathbf{e}_1, \dots, \mathbf{e}_K]$
- 30    Minimize the mutual information in  $\mathbf{e}$  via Eq. 3
- 31     $\mathcal{E}_{codebook} = [\mathbf{e}_1, \dots, \mathbf{e}_K]$
- 32 **return**  $\mathcal{E}_{codebook}$ .

---

where  $\mathcal{O}_{<t,1} = [\mathcal{O}_{1,1}, \dots, \mathcal{O}_{t-1,1}]$ , while  $\theta_{AR}$  represents the AR Transformer model parameters.  $\theta_{NAR}$  represents the NAR model parameters, while  $\mathcal{O}_{:,l}$  denotes the entire sequence of  $\mathcal{O}_{t,l}$  for the  $l$ th layer, and  $\mathcal{O}_{:,<l} = [\mathcal{O}_{:,1}, \dots, \mathcal{O}_{:,l-1}]$  for  $l = 2, \dots, 8$ . Note that the AR model in SESLM is conditioned on the concatenated embeddings of both the acoustic and textual prompts. This formulation differs from that of VALL-E, where the AR model is only conditioned on the textual prompt and the past acoustic history. We validate the effectiveness of the structural entropy-based speech language model on the zero-shot TTS task. During inference, text input is converted to a phoneme sequence and the speech prompt to speech tokens. These are concatenated to form the prompts for both the AR and NAR models. The tokens generated by the AR and NAR models are then concatenated to construct the speech token matrix. Finally, the SECodec decoder is used to generate the waveform conditioned on the complete token matrix.

## 4 Experiment

### 4.1 Experimental Setup

**Data** For SECodec training, we use the LibriSpeech (Panayotov et al. 2015) dataset. At each training iteration, a 3.2 second segment is randomly cropped from the speech samples. For zero-shot TTS, we train AR and NAR models on the English subset of the Multilingual LibriSpeech dataset (Pratap et al. 2020), which contains 44,000 hours of transcribed speech data derived from LibriVox audiobooks. We select speech samples with durations ranging from 3 to 14 seconds for the training data. All speech data is sampled at a rate of 16 kHz.

**Model** SECodec is built on the framework of RVQ-GANs, following the same pattern as FunCodec (Du et al. 2024). SECodec uses the convolutional-based encoder-decoder network from EnCodec, which performs temporal downscaling with a chosen striding factor. For zero-shot TTS experiments, AR model and NAR model are both 12-layer Transformer decoders with 16 attention heads, an attention dimension of 1024 and the FFN dimension of 4096.

**Training** During the training stage, we randomly clip a continuous segment of 3.2 seconds from an utterance, which is considered as a training sample. Before being fed into the encoder, the segment undergoes root-mean-square (RMS) normalization. The reconstructed output is rescaled using inverse normalization to calculate losses. We train the models on single 3090Ti GPUs with a total batch size of 16. Under the adversarial training framework, we update the codec model 300,000 times. To prevent the discriminator from becoming too dominant, we only update it when its loss exceeds that of the codec model.

**Baselines** For SECodec, we consider two baselines: 1) EnCodec (Défossez et al. 2024), and 2) SpeechTokenizer (Zhang et al. 2024), a state-of-the-art speech tokenizer for speech language models. For SESLM, we consider two baselines: 1) VALL-E (Wang et al. 2023), and 2) ULSM (Zhang et al. 2024).

### 4.2 Metrics

For speech reconstruction evaluation, we randomly sampled 300 speech samples from the LibriSpeech test set, considering both subjective and objective evaluation metrics. For objective metrics, we used mel-cepstrum distortion (MCD) (Toda, Black, and Tokuda 2007), root mean square errors (RMSE) (Luo et al. 2017), and ViSQOL (Hines et al. 2015) to assess speech quality. Additionally, we evaluated content accuracy through Word Error Rate (WER) by transcribing the speech using the Whisper en-medium model (Radford et al. 2023). For subjective metrics, we use MUSHRA following SpeechTokenizer (Zhang et al. 2024).

For evaluating SESLM, we perform zero-shot text-to-speech assessments using the VCTK (Veaux et al. 2016) dataset, which includes recordings from 108 speakers with no overlap between the training data and the VCTK dataset. For each speaker, we randomly select a 3-second utterance as the prompt and use the text from a different utterance as the input. Objective metrics include evaluating speaker similarity

Methods	$K$ in codebook	WER ↓	MCD ↓	RMSE ↓	ViSQOL ↑	MUSHRA ↑	Param
Ground Truth	-	4.58	-	-	-	91.46	-
EnCodec	468	8.87	7.21	36.32	4.03	71.32	19.19M
EnCodec	1024	5.01	5.89	32.59	4.37	79.86	23.76M
EnCodec	2048	4.98	6.79	35.12	4.11	75.11	32.18M
EnCodec	4096	91.23	10.34	49.72	-	-	82.71M
SpeechTokenizer	1024	5.04	6.23	34.84	4.30	90.55	120M
<b>SECodec (Nodes:10000, Edges:&gt;0.2)</b>	468	<b>4.63</b>	<b>5.12</b>	<b>31.97</b>	<b>4.40</b>	<b>90.79</b>	<b>19.19M</b>
<i>Nodes Ablations (Edges:&gt;0.2)</i>							
SECodec (Nodes:100000)	1212	4.91	6.37	32.79	4.31	88.21	25.31M
SECodec (Nodes:50000)	768	4.82	5.57	32.45	4.37	90.12	22.27M
<i>Edges Ablations (Nodes:10000)</i>							
SECodec (Edges:>0.5)	1212	4.87	6.19	31.81	4.29	86.33	25.31M
<i>Methods Ablations</i>							
SECodec w/o SE Codebook	468	4.89	6.03	32.37	4.32	89.35	19.19M
SECodec w/o SE Quantization	468	4.71	5.67	33.89	4.39	87.62	19.19M

Table 1: Comparison of WER, MCD, RMSE, ViSQOL and MUSHRA of speech reconstruction on the LibriSpeech datasets.

and Word Error Rate (WER). To assess speaker similarity, we utilize the WavLM-TDCNN (Chen et al. 2022) speaker embedding model to measure the similarity. We report the similarity with respect to the resynthesized audio context by its vocoder (SIM-r) and the similarity against the original audio context (SIM-o). As subjective metrics, we utilize Mean Opinion Score (MOS) for evaluating subjective audio quality (QMOS) and Similarity MOS (SMOS) for assessing subjective audio similarity.

### 4.3 Main Results

**Speech Reconstruction** Table 1 provides a detailed summary of the speech reconstruction experiment results. SECodec outperforms both SpeechTokenizer and EnCodec by achieving a lower WER, highlighting its superior capability to retain content information. Furthermore, SECodec surpasses SpeechTokenizer and EnCodec in both MCD, RMSE, VISQOL and MUSHRA scores, underscoring its enhanced proficiency in producing high-quality speech.

**Zero-shot TTS** Table 2 illustrates that our SESLM achieves a lower Word Error Rate (WER) compared to USLM and VALL-E. This finding underscores SECodec’s capability to enhance the precision of content information modeling. Moreover, SESLM exhibits superior speaker similarity, suggesting that the modeled speech’s structural information more effectively facilitates the extraction of paralinguistic features.

**Ablation Study** Furthermore, we conducted an ablation study to analyze the performance effects of different components in SECodec, with the results presented in Table 1. The findings show that when SE quantization is removed, the model still performs well and outperforms most of the baseline models, but there is a noticeable decrease in speech quality. Additionally, when the SE codebook component is excluded, the results are significantly poorer, highlighting its critical importance to the overall performance.

### 4.4 Analysis

**Choice of Nodes and Edges for SE** We first analyzed the impact of different codebook sizes on the performance of EnCodec. It is evident that selecting an appropriate codebook size is crucial; a small codebook size (e.g., 468) leads to performance degradation, while an excessively large codebook (e.g., 4096) results in the model’s failure to converge. Furthermore, we examined the influence of the number of nodes and edges in the speech feature graph during the SE minimization process. Table 1 presents the impact of varying the number of nodes and edges on the performance of SECodec. It can be observed that different numbers of nodes and edges, constructed with different similarity thresholds, lead to varying codebook sizes, which in turn affect the results. The best performance in speech reconstruction was achieved with 10,000 nodes and an edge threshold of 0.2. A similar conclusion is drawn from the ablation experiments on SESLM, as shown in Table 2. On the other hand, SECodec achieves the best performance while also having the smallest model parameter size, further highlighting the importance of automatically selecting the appropriate codebook size.

**Codebook and Quantized Output Visualization** To demonstrate the compressive and informative nature of the codebook learned by SECodec, we first visualized each column vector in the codebooks initialized by different methods using t-SNE. As illustrated in Figure 2(a) and Figure 2(b), the codebook initialized with k-means exhibits an uneven distribution in space, with cluster centers entangled and overlapping. In contrast, the codebook initialized with SECodec is more evenly distributed and discrete. To further illustrate the effectiveness of the codebook learned by SECodec, we visualized the quantized outputs of 1,000 speech samples. As shown in Figure 2(c) and Figure 2(d), the features of different speech samples obtained by SECodec are more disentangled and distinguishable compared to those obtained by EnCodec.

**Effectiveness of SE Quantization** Figure 3 displays the spectrogram and F0 (fundamental frequency) of both the reference speech and the resynthesized (synthesized) speech

Methods	Codec	WER ↓	SIM-o ↑	SIM-r ↑	QMOS ↑	SMOS ↑
Ground Truth	-	1.92	0.698	n/a	3.89(±0.18)	3.92(±0.16)
SECodec resynthesis	SECodec	2.02	0.682	n/a	3.82(±0.12)	3.88(±0.11)
VALL-E	EnCodec	7.09	0.501	0.412	3.08(±0.10)	3.31(±0.11)
USLM	SpeechTokenizer	5.79	0.602	0.587	3.50(±0.13)	3.41(±0.12)
<b>SESLM (Nodes:10000, Edges:&gt;0.2)</b>	<b>SECodec</b>	<b>4.97</b>	<b>0.634</b>	<b>0.611</b>	<b>3.63(±0.12)</b>	<b>3.45(±0.11)</b>
<i>Nodes Ablations (Edges:&gt;0.2)</i>						
SESLM (Nodes:100000)	SECodec	5.51	0.612	0.593	3.56(±0.11)	3.37(±0.13)
<i>Edges Ablations (Nodes:10000)</i>						
SESLM (Edges:>0.5)	SECodec	5.48	0.602	0.597	3.46(±0.10)	3.33(±0.11)

Table 2: Comparison of WER, SIM-o, SIM-r, QMOS and SMOS of the proposed model and baselines on VCTK datasets.

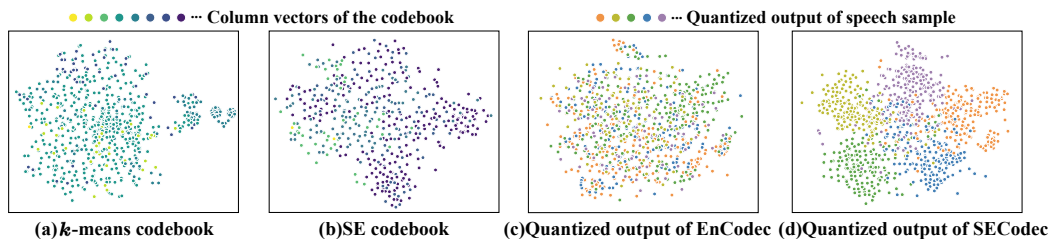


Figure 2: Codebook and quantized output visualization.

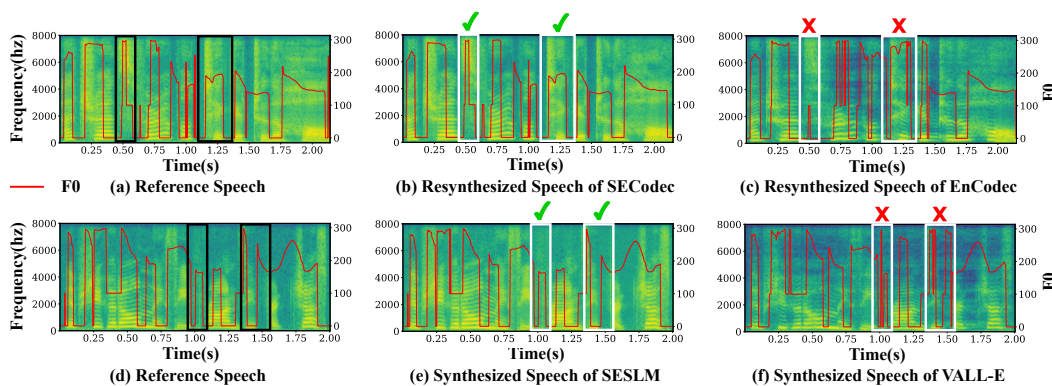


Figure 3: Effectiveness of SE quantization. The black box indicates the structural details in the reference speech, the white box with a green checkmark indicates the part that our method correctly predicts, and the white box with a red cross indicates the part that the baseline method incorrectly predicts.

with identical content. It is evident that SECodec preserves the detailed features of the audio more effectively than EnCodec, particularly in maintaining the intricacies of the wave peaks at the fundamental frequency line (F0), as shown in Figure 3(a), Figure 3(b), and Figure 3(c). This observation is consistent in the speech synthesized by SESLM compared to VALL-E, as shown in Figure 3(d), Figure 3(e), and Figure 3(f), demonstrating that quantization via node game-based 2D SE minimization significantly reduces speech distortion caused by Euclidean distance-based quantization. Consequently, our method retains more detailed information in the synthesized speech.

## 5 Conclusion

Interacting with LLMs through speech has led to an increased demand for effective speech representation discretization.

To address this, we propose SECodec, a novel speech representation codec designed to convert continuous speech waveforms into discretized tokens. SECodec automatically determines the appropriate codebook size and integrates structural information into the quantization process. Extensive experiments demonstrate that SECodec outperforms EnCodec in speech reconstruction. Furthermore, we developed a Structural Entropy-based Speech Language Model (SESLM), yielding superior results in terms of the quality of the generated speech. Furthermore, the experimental results demonstrate that SECodec is capable of learning a more compact and discrete codebook, thereby generating more informative speech tokens. SECodec exhibits comparable performance to EnCodec in speech reconstruction, while SESLM outperforms VALL-E in zero-shot text-to-speech tasks.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Nos. U24A20334, 62376111, U23A2038 and 61972186), Yunnan provincial major science and technology special plan projects (Nos. 202302AD080003, 202402AG050007), Yunnan Provincial Key Research and Development Plan (Nos. 202303AP140008), Yunnan Provincial Basic Research Major Project (Nos. 202401BC070021). The authors would like to thank anonymous reviewers for their comments.

## References

- Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Baevski, A.; Zhou, Y.; Mohamed, A.; and Auli, M. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proceedings of the NeurIPS Conference 2020*, 12449–12460.
- Borsos, Z.; Marinier, R.; Vincent, D.; Kharitonov, E.; Pietquin, O.; Sharifi, M.; Roblek, D.; Teboul, O.; Grangier, D.; Tagliasacchi, M.; et al. 2023. Audioldm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31: 2523–2533.
- Cao, Y.; Peng, H.; Li, A.; You, C.; Hao, Z.; and Yu, P. S. 2024a. Multi-Relational Structural Entropy. In *Proceedings of the UAI Conference 2024*, 1–xx.
- Cao, Y.; Peng, H.; Yu, Z.; and Philip, S. Y. 2024b. Hierarchical and incremental structural entropy minimization for unsupervised social event detection. In *Proceedings of the AAAI Conference 2024*, 8255–8264.
- Chen, S.; Wang, C.; Chen, Z.; Wu, Y.; Liu, S.; Chen, Z.; Li, J.; Kanda, N.; Yoshioka, T.; Xiao, X.; et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1505–1518.
- Cheng, P.; Hao, W.; Dai, S.; Liu, J.; Gan, Z.; and Carin, L. 2020. Club: A contrastive log-ratio upper bound of mutual information. In *Proceedings of the ICML Conference 2020*, 1779–1788.
- Chung, Y.-A.; Zhang, Y.; Han, W.; Chiu, C.-C.; Qin, J.; Pang, R.; and Wu, Y. 2021. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *Proceedings of the ASRU Conference 2021*, 244–250.
- Défossez, A.; Copet, J.; Synnaeve, G.; and Adi, Y. 2024. High fidelity neural audio compression. In *Proceeding of the ICLR Conference 2024*, 2835–8856.
- Dong, Q.; Huang, Z.; Tian, Q.; Xu, C.; Ko, T.; Zhao, Y.; Feng, S.; Li, T.; Wang, K.; Cheng, X.; et al. 2024. Polyvoice: Language models for speech to speech translation. In *Proceeding of the ICLR Conference 2024*, 1–xx.
- Du, Z.; Zhang, S.; Hu, K.; and Zheng, S. 2024. Funcodec: A fundamental, reproducible and integrable open-source toolkit for neural speech codec. In *Proceedings of the ICASSP Conference 2024*, 591–595.
- Gray, R. 1984. Vector quantization. *IEEE Assp Magazine*, 1(2): 4–29.
- Hassid, M.; Remez, T.; Nguyen, T. A.; Gat, I.; Conneau, A.; Kreuk, F.; Copet, J.; Defossez, A.; Synnaeve, G.; Dupoux, E.; et al. 2023. Textually pretrained speech language models. In *Proceedings of the NeurIPS Conference 2023*, 2771–2789.
- Hines, A.; Skoglund, J.; Kokaram, A. C.; and Harte, N. 2015. ViSQOL: an objective speech quality model. *EURASIP Journal on Audio, Speech, and Music Processing*, 2015: 1–18.
- Hsu, W.-N.; Bolte, B.; Tsai, Y.-H. H.; Lakhotia, K.; Salakhutdinov, R.; and Mohamed, A. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29: 3451–3460.
- Huang, Z.; Meng, C.; and Ko, T. 2023. Repproc: A speech representation codec for speech tokenization. *arXiv preprint arXiv:2309.00169*.
- Lakhotia, K.; Kharitonov, E.; Hsu, W.-N.; Adi, Y.; Polyak, A.; Bolte, B.; Nguyen, T.-A.; Copet, J.; Baevski, A.; Mohamed, A.; et al. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9: 1336–1354.
- Li, A.; and Pan, Y. 2016. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory*, 62(6): 3290–3339.
- Luo, Z.; Chen, J.; Takiguchi, T.; and Ariki, Y. 2017. Emotional voice conversion using neural networks with arbitrary scales F0 based on wavelet transform. *EURASIP Journal on Audio, Speech, and Music Processing*, 2017: 1–13.
- Nguyen, T. A.; Sagot, B.; and Dupoux, E. 2022. Are discrete units necessary for spoken language modeling? *IEEE Journal of Selected Topics in Signal Processing*, 16(6): 1415–1423.
- Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: an asr corpus based on public domain audio books. In *Proceedings of the ICASSP Conference 2015*, 5206–5210.
- Peng, H.; Zhang, J.; Huang, X.; Hao, Z.; Li, A.; Yu, Z.; and Yu, P. S. 2024. Unsupervised social bot detection via structural information theory. *ACM Transactions on Information Systems*.
- Pratap, V.; Xu, Q.; Sriram, A.; Synnaeve, G.; and Collobert, R. 2020. MIs: A large-scale multilingual dataset for speech research. In *Proceedings of the Interspeech Conference 2020*, 2757–2761.
- Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the ICML Conference 2023*, 28492–28518.
- Rubenstein, P. K.; Asawaroengchai, C.; Nguyen, D. D.; Bapna, A.; Borsos, Z.; Quiry, F. d. C.; Chen, P.; Badawy, D. E.; Han, W.; Kharitonov, E.; et al. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.
- Sun, L.; Huang, Z.; Peng, H.; Wang, Y.; Liu, C.; and Yu, P. S. 2024. LSEnet: Lorentz Structural Entropy Neural Network for Deep Graph Clustering. In *Proceedings of the ICML Conference 2024*, 1–xx.

- Toda, T.; Black, A. W.; and Tokuda, K. 2007. Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(8): 2222–2235.
- Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Tu, Y.; Li, L.; Su, L.; Du, J.; Lu, K.; and Huang, Q. 2023. viewpoint-Adaptive representation disentanglement network for change captioning. *IEEE Transactions on Image Processing*, 32: 2620–2635.
- Tu, Y.; Li, L.; Su, L.; Gao, S.; Yan, C.; Zha, Z.-J.; Yu, Z.; and Huang, Q. 2022. I2 Transformer: Intra-and inter-relation embedding transformer for TV show captioning. *IEEE Transactions on Image Processing*, 31: 3565–3577.
- Tu, Y.; Li, L.; Su, L.; Zha, Z.-J.; and Huang, Q. 2024. SMART: Syntax-Calibrated Multi-Aspect Relation Transformer for Change Captioning. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 46(07): 4926–4943.
- Vasuki, A.; and Vanathi, P. 2006. A review of vector quantization techniques. *IEEE Potentials*, 25(4): 39–47.
- Veaux, C.; Yamagishi, J.; MacDonald, K.; et al. 2016. Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit.
- Wang, C.; Chen, S.; Wu, Y.; Zhang, Z.; Zhou, L.; Liu, S.; Chen, Z.; Liu, Y.; Wang, H.; Li, J.; et al. 2023. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.
- Yang, D.; Liu, S.; Huang, R.; Tian, J.; Weng, C.; and Zou, Y. 2023. Hifi-codec: Group-residual vector quantization for high fidelity audio codec. *arXiv preprint arXiv:2305.02765*.
- Yang, R.; Peng, H.; Li, A.; Li, P.; Liu, C.; and Philip, S. Y. 2024a. Hierarchical Abstracting Graph Kernel. *IEEE Transactions on Knowledge and Data Engineering*.
- Yang, R.; Peng, H.; Liu, C.; and Li, A. 2024b. Incremental measurement of structural entropy for dynamic graphs. *Artificial Intelligence*, 334: 104175.
- Yang, Y.; Wu, Q.; He, B.; Peng, H.; Yang, R.; Hao, Z.; and Liao, Y. 2024c. SeBot: Structural Entropy Guided Multi-View Contrastive Learning for Social Bot Detection. In *Proceedings of the ACM SIGKDD Conference 2024*, 3841–3852.
- Yang, Z.; Wei, Y.; Li, H.; Li, Q.; Jiang, L.; Sun, L.; Yu, X.; Hu, C.; and Peng, H. 2024d. Adaptive Differentially Private Structural Entropy Minimization for Unsupervised Social Event Detection. In *Proceedings of the ACM CIKM Conference 2024*, 2950–2960.
- Zeghidour, N.; Luebs, A.; Omran, A.; Skoglund, J.; and Tagliasacchi, M. 2021. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30: 495–507.
- Zeng, G.; Peng, H.; Li, A.; Wu, J.; Liu, C.; and Philip, S. Y. 2024. Scalable Semi-Supervised Clustering Via Structural Entropy With Different Constraints. *IEEE Transactions on Knowledge and Data Engineering*.
- Zeng, X.; Peng, H.; and Li, A. 2024. Effective Exploration Based on the Structural Information Principles. In *Proceedings of the NeurIPS Conference 2024*, 1–xx.
- Zhang, D.; Li, S.; Zhang, X.; Zhan, J.; Wang, P.; Zhou, Y.; and Qiu, X. 2023. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. In *Proceedings of the Findings of EMNLP Conference 2023*, 15757–15773.
- Zhang, X.; Zhang, D.; Li, S.; Zhou, Y.; and Qiu, X. 2024. Spechtokenizer: Unified speech tokenizer for speech large language models. In *Proceedings of the ICLR Conference 2024*, 1–xx.
- Zou, D.; Wang, S.; Li, X.; Peng, H.; Wang, Y.; Liu, C.; Sheng, K.; and Zhang, B. 2024. Multispans: A multi-range spatial-temporal transformer network for traffic forecast via structural entropy optimization. In *Proceedings of the WSDM Conference 2024*, 1032–1041.