

# Data Side of the Moon

Decoding Pink Floyd's Legacy

2024 年 8 月 28 日

# 目录

<b>1 背景</b>	<b>1</b>
1.1 音乐风格 . . . . .	1
1.1.1 什么是音乐风格? . . . . .	1
1.1.2 我们是怎么认识风格的? . . . . .	1
1.1.3 音乐风格的形成 . . . . .	1
1.2 我们如何“听”音乐? . . . . .	2
1.3 计算机能否“听”音乐? . . . . .	2
<b>2 介绍</b>	<b>3</b>
<b>3 音频预处理</b>	<b>3</b>
3.1 分帧、加窗、快速傅立叶变换 . . . . .	4
3.2 梅尔滤波器组和梅尔频谱图 . . . . .	4
<b>4 编码器</b>	<b>5</b>
4.1 基于 MFCC 的编码器 . . . . .	5
4.1.1 MFCC (梅尔倒谱系数) . . . . .	5
4.1.2 引入 Phi-Moment 加权计算音乐特征向量 . . . . .	5
4.2 Echoes: 基于卷积自编码器 (CAE) 的编码器 . . . . .	6
4.2.1 自编码器 . . . . .	6
4.2.2 相关工作 . . . . .	6
4.2.3 模型结构 . . . . .	7
4.2.4 模型训练 . . . . .	7
4.2.5 模型效果评估 . . . . .	9
<b>5 基于音乐特征向量的系列任务</b>	<b>10</b>
5.1 Echoes 模型对音乐风格的捕捉 . . . . .	10
5.1.1 音乐特征向量聚类与可视化 . . . . .	10
5.1.2 无监督音乐分类 . . . . .	11
5.2 音乐特点与组成变化分析 . . . . .	11
5.3 基于音乐内容的歌曲推荐 . . . . .	13
5.3.1 Echoes 模型和乐迷眼中的“平均歌” . . . . .	13
5.3.2 曲库中的相似歌曲匹配/推荐 . . . . .	13
<b>6 未来工作与讨论</b>	<b>15</b>
6.1 未来工作 . . . . .	15
6.2 讨论 . . . . .	16

# 1 背景

## 1.1 音乐风格

### 1.1.1 什么是音乐风格？

音乐风格（音乐类型，Music Genres）是对于音乐作品归属的传统性分类 [1]。常见的音乐风格包括古典（Classical）、爵士（Jazz）、摇滚（Rock）、流行（Pop）、嘻哈（Hip-Hop）等。

### 1.1.2 我们是怎么认识风格的？

人们对音乐风格的认识是通过长期的听觉经验和文化影响形成的。听众在接触不同类型的音乐时，会逐渐积累对不同风格特征的理解和认知。例如，摇滚音乐通常具有强烈的节奏和电吉他的使用，而古典音乐则以复杂的乐曲结构和管弦乐器为特点。

### 1.1.3 音乐风格的形成

音乐风格的形成并不简单。它不仅涉及音乐本身的特征，还受文化、历史和社会因素的影响。不同的音乐风格在时间和地域上也有着显著的差异。

而对于音乐风格形成本身，其实某种程度上是一个聚类的过程。先有创作出来的音乐作品，在大量具有类似特点的作品基础上，将其聚类成不同的风格。

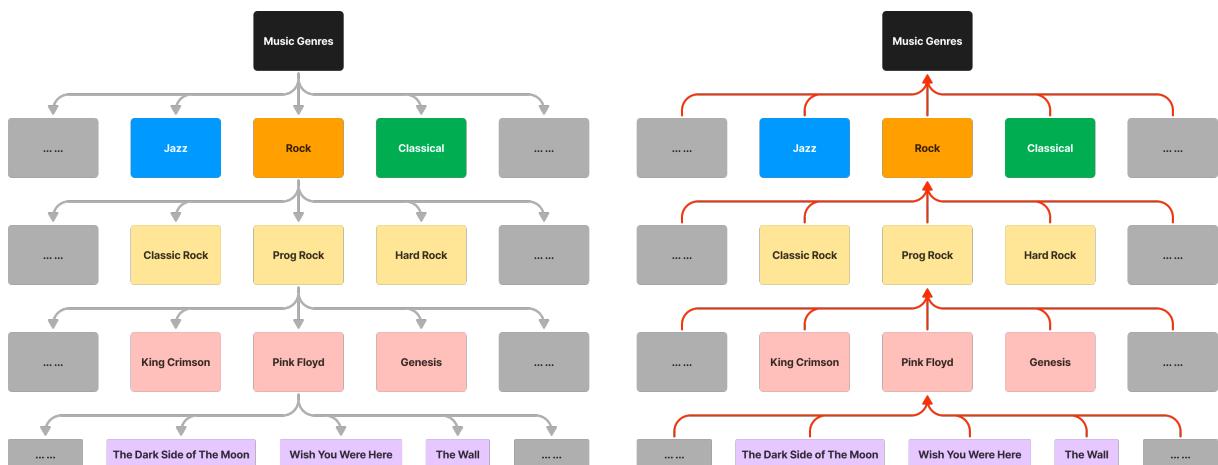


图1 音乐风格

图1中展示了音乐风格的层次结构和分类过程：顶层：音乐风格（Music Genres），包括各种主要音乐风格如爵士（Jazz）、摇滚（Rock）、古典（Classical）等。中层：子风格，在主要风格下进一步细分，如摇滚音乐可以分为经典摇滚（Classic Rock）、前卫摇滚（Prog Rock）、硬摇滚（Hard Rock）等。底层：具体乐队和作品，每个子风格下可以列出代表性的乐队和作品，如前卫摇滚下的 Pink Floyd，及其经典专辑《The Dark Side of The Moon》、《Wish You Were Here》、《The Wall》等。

左图代表先有音乐的音乐风格不断细分的过程，而右图在很大程度上代表的是从歌曲的创造到风格形成的过程，其天然就呈现出聚类的特点。

## 1.2 我们如何“听”音乐？

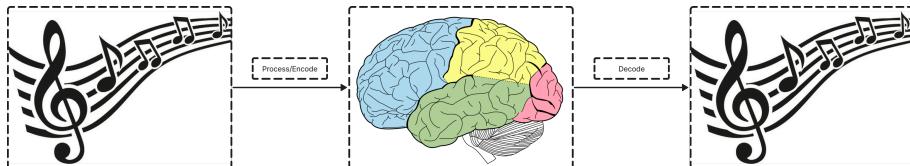


图 2

**人类听觉过程：**人类通过耳朵和骨传导听音乐。声音以声波的形式传到耳朵，通过鼓膜和听小骨传到内耳。内耳的耳蜗将这些机械振动转化为电信号，通过听神经传到大脑。

**音乐在大脑中的存在形式：**当音乐以电信号传到大脑，大脑又会以什么形式将其进行储存？这点目前尚不完全清楚，而每当人们想起一首音乐，脑海中浮现的可能是旋律、声音或节奏，这点也是因人而异的。

但是不可否认的是，大脑一定通过了某种方式将原始音频的特征（如节奏、独特的音色、器乐声等）进行了“编码”，并以某种方式存储在大脑里。

## 1.3 计算机能否“听”音乐？

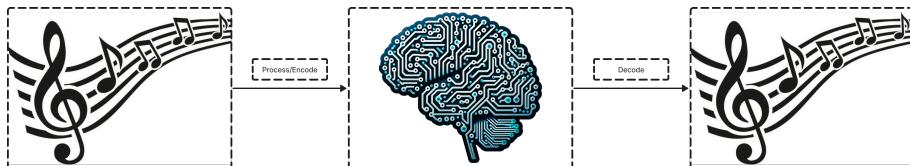


图 3

从计算机科学的视角来看，这一过程的核心属于音乐信息检索（Music Information Retrieval）任务：计算机通过分析音乐信号的频率、节奏、音色等特征，提取出代表音乐特征的数据。这些数据可以用于音乐分类、推荐、生成等任务。[\[2\]](#)

可以看出，人脑和计算机在音乐处理方面有着过程上的相似性。基于传统的信号处理方法已经在计算机音频领域取得了众多成果[\[3\]](#)，而随着人工神经网络，尤其是深度神经网络的发展，在 MIR 中应用神经网络已经取得了很多成果[\[4\]](#)。

## 2 介绍

受到人脑与计算机对音频处理的流程/模式上的相似性的启发，我们将基于音频文件的梅尔频谱图，分别设计了基于梅尔倒谱系数和卷积自编码器的两种编码模型，对音乐进行编码，得到特征向量。基于特征向量，与音乐元数据结合，我们进行了对音乐聚类、分析艺人曲风变化、基于音乐内容的歌曲推荐等工作。

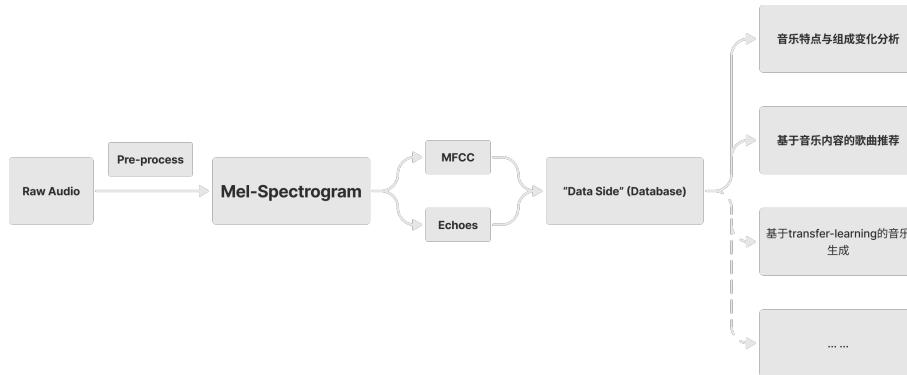


图 4 项目流程图

本文在后续工作中选择了乐队 Pink Floyd 的作品进行分析，主要考虑到笔者对 Pink Floyd 的音乐作品和乐队历史等内容有着深入了解，可以更好的评价模型效果。

## 3 音频预处理

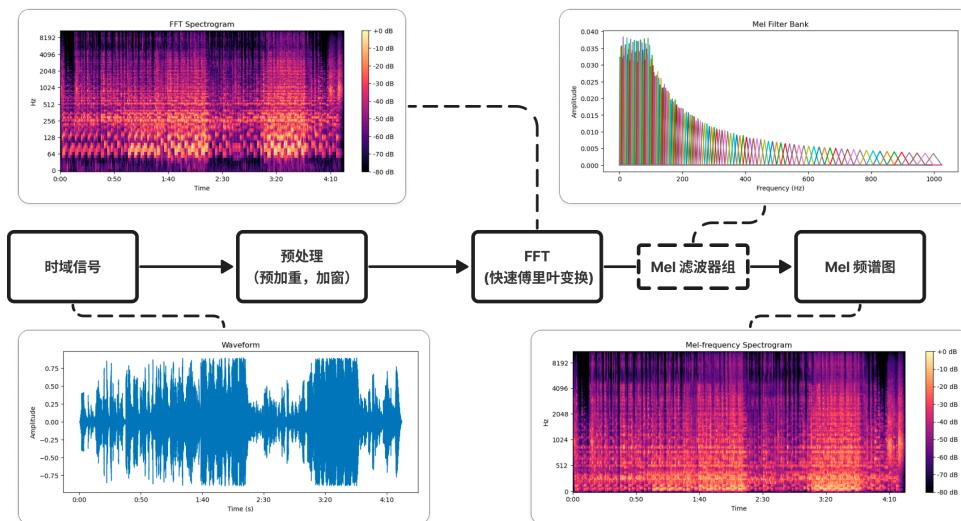


图 5 预处理流程图

### 3.1 分帧、加窗、快速傅立叶变换

分帧 (图10 A,B) 将音频信号分成短时帧, 以方便获得更加平稳的信号来进行傅立叶变换。加窗 (图10 C) 将对每个短时帧乘以一个窗函数, 以增加帧左端和右端的连续性。抵消 FFT 假设 (数据是无限的), 并减少频谱泄漏。快速傅里叶变换 (FFT, 图10 D) 将时域信号转换为频域信号, 由于信号在时域上的变换通常很难看出信号的特性, 因此通常会把它转换为频域上的能量分布来观察, 不同的能量分布, 就能代表不同语音的特性。

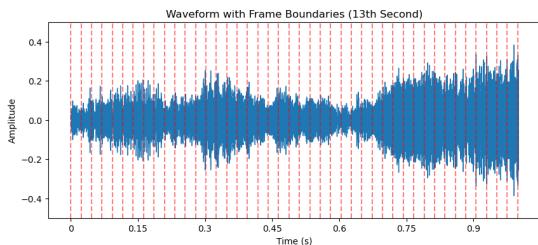


图 6 A

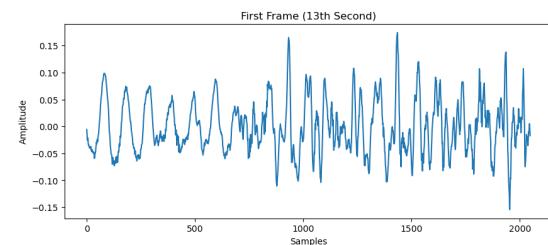


图 7 B

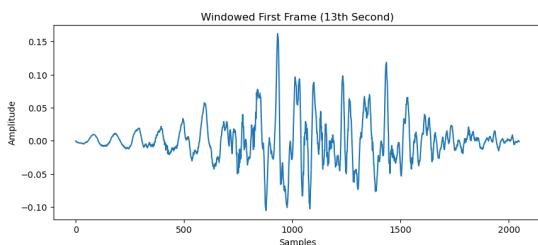


图 8 C

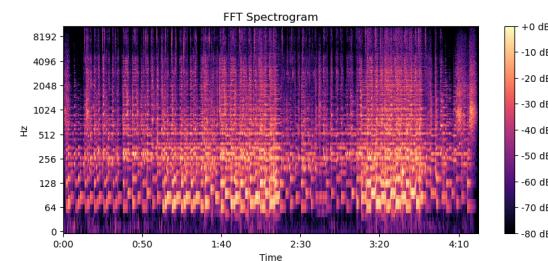


图 9 D

图 10 分帧、加窗、快速傅立叶变换

### 3.2 梅尔滤波器组和梅尔频谱图

根据梅尔刻度对频谱进行处理, 其转化为梅尔频谱, 这可以更好地模拟人类耳朵的频率感知, 将线性频谱映射到梅尔尺度会更加接近人耳听觉特性。后续的工作将基于梅尔频谱图进行

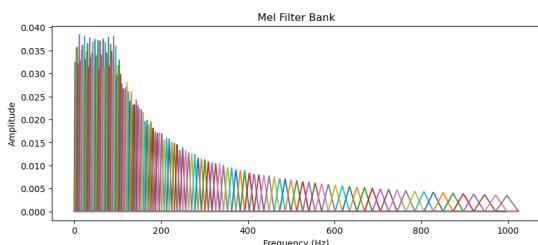


图 11 梅尔滤波器组

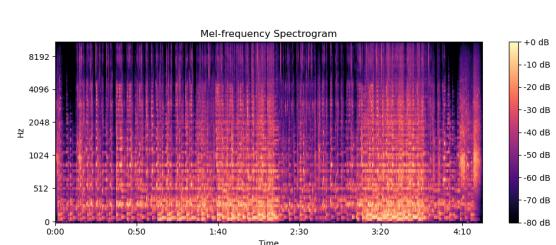


图 12 梅尔频谱图

## 4 编码器

### 4.1 基于 MFCC 的编码器

#### 4.1.1 MFCC (梅尔倒谱系数)

基于3得到的梅尔频谱图可以直接计算梅尔倒谱系数 (MFCC): 【research】介绍对数转换将梅尔频谱的幅度值取对数, 因为人耳对声音的感知是对数关系的, 这样可以更好地模拟人耳对不同声音强度的感知。离散余弦变换 (DCT) 将对数梅尔频谱转换到倒谱域, 以得到梅尔频率倒谱系数 (MFCC)。DCT 的作用是将特征集中在前几个系数上, 从而实现降维, 并减少频谱的相关性, 使特征更加稳定和易于处理。由上述流程, 我们对每首歌提取了 20 个 MFCC 特征, 取平均后获取一个 20 维的特征向量。

基于 MFCC 的模型在音乐风格分类等任务中可以取得不错的效果 [5, 6]

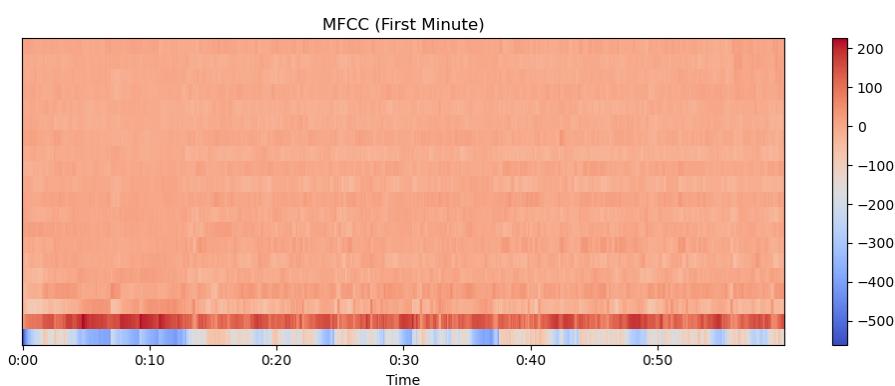


图 13 MFCC

#### 4.1.2 引入 Phi-Moment 加权计算音乐特征向量

因为音乐不同时刻包含的信息是不同的, 所以直接对 MFCC 系数取平均得到特征向量可能会有些失衡, 下面我们试图利用 Phi-moment 给出一种解决方案。Phi-moment 是音乐中处于黄金比例 (0.618) 位置的时刻, 这通常被认为是曲子最和谐、最平衡的部分 [7]。因此, 使用 MFCC 提取音频 phi-moment 处 30 秒的频谱特征, 能够有效捕捉这段音乐的关键频谱特征, 从而对整段音频具有较好的代表性。基于此理论, 可以通过两种方法来优化音频特征提取过程: 一是仅计算 phi-moment 处的特征向量, 这样可以减少计算时间, 同时保证特征的代表性; 二是在对整首歌进行 MFCC 计算时, 加大 phi-moment 处的权重, 从而放大这首歌的关键特征。这两种方法都能提高特征提取的效率和准确性, 使得分析和处理音乐更加精准有效。

## 4.2 Echoes：基于卷积自编码器（CAE）的编码器

使用 MFCC 提取音频特征能够使用较小的计算资源，得到不错的效果，且其在语音内容处理等任务中有着表现出色 [8]。但是由于音乐本身的复杂性，其在最基本的节奏、旋律、音色之外，仍有很多信息，因此 MFCC 模型在处理音乐时“上限”较低。因此，我们提出了基于卷积自编码器的 Echoes 模型，试图使编码后的特征向量包含更多的信息。

### 4.2.1 自编码器

自编码器（Autoencoder）是一种神经网络模型，常用于无监督学习任务，特别是在数据压缩和特征提取方面。自编码器由两个主要部分组成：编码器和解码器。编码器将输入数据映射到一个低维隐空间（Latent Space），生成一个紧凑的表示（编码）。解码器则将这个低维表示重新映射回原始数据空间，尽可能重建输入数据。通过最小化输入与重建数据之间的差异，自编码器能够学习到数据的关键特征。由于自编码器的这种特性，它在降维、去噪以及生成模型等领域有广泛应用。在音乐信息检索（MIR）项目中，自编码器可以用来处理复杂的音频信号，将其转化为低维特征表示，从而提高后续任务如分类或检索的效果。[9, 10]

### 4.2.2 相关工作

这里希望将 Echoes 模型与目前在音乐风格分类任务中取得 SOTA(State of the Art) 的深度学习模型（CNN、RNN、LSTM）[11, 12, 13] 进行对比——有监督 vs. 无监督。

虽然这些深度学习模型在音乐风格分类任务中能取得很好的效果，但笔者也在反思音乐分类任务本身的合理性：

- 训练/测试数据集中的风格标签是由人来标注的，且风格数量往往较少。在这样的分类任务中，如果增加风格数量，模型的分类效果则会下降。所以如果有新的分类，已训练好的模型并不能保持优秀的表现。并且，将神经网络限制在有限的人为分类风格下，在一定程度上并不能真正发挥神经网络的能力。
- 正如开篇问题所述，风格是从音乐出发来定义的。Echoes 模型的设计是试图从音乐出发，真正专注于音乐本身，通过无监督学习的方式，发掘音乐之间的关系。
- 音乐的艺术本质也决定了，这样的分类往往是专断且陈腐的。[1] 在很大程度上可以认为，风格永远是滞后于音乐的。

自编码器作为一种神经网络结构，在音乐处理中已经有较多应用 [14, 15, 16, 17]。受到1中关于人脑接受处理音乐过程的启发，笔者认为自编码器的训练过程在一定程度上能模拟人类学习一首歌的过程：耳朵、鼓膜等对音乐进行“编码”（可以看做编码器），以某种方式储存在大脑中（可以看做特征层），然后通过声带等器官将歌曲唱出来（可

以看做解码器），通过对唱出来的歌曲和原来的歌曲，不断练习尽可能缩小差距（优化，减小损失函数值）。

#### 4.2.3 模型结构

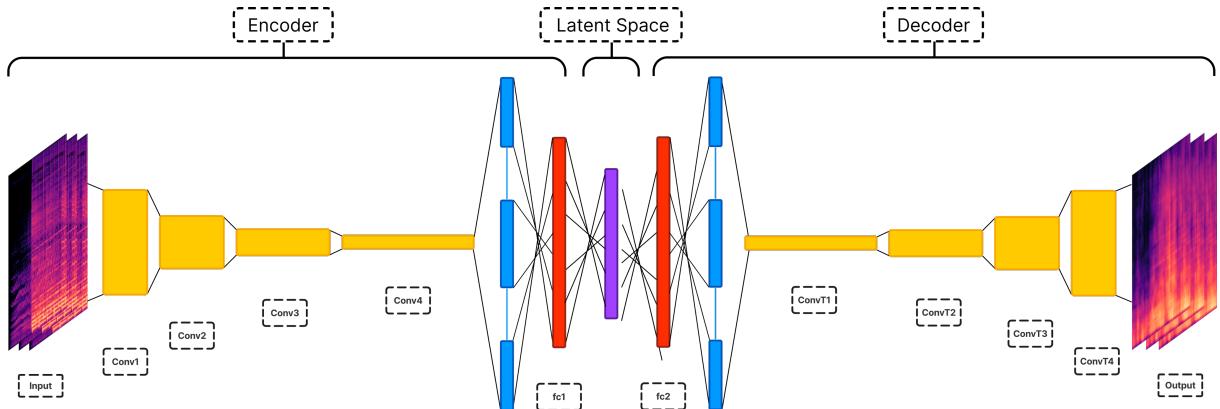


图 14 Enter Caption

**编码器 (Encoder)** 由卷积层 (CNN) 和全连接层 (FC) 组成，用于将输入的梅尔频谱图像编码为隐向量。四个卷积层均使用大小为  $3 \times 3$  的卷积核，步长设置为 2，每个卷积层后接批归一化和 ReLU 激活函数，Conv1-Conv4 的输入、输出通道数分别为  $(1, 16), (1, 32), (32, 64), (64, 128)$ 。之后通过 Flatten 层：将卷积层的输出展平为一维向量，经过两个全连接层 (FC) 得到隐向量，其中输出维度分别为 256, 128，后者即为特征向量维度。

**解码器 (Decoder)** 结构与编码器相反，使用全连接层和卷积转置层将隐向量解码为大小与原始输入相同的二维频谱图像。

#### 4.2.4 模型训练

**数据集：**模型训练数据集选取 GTZAN 数据集 [18]，GTZAN 是一个用于音频信号音乐流派分类的数据集。该数据集包含 1,000 条音频轨道，每条时长 30 秒。数据集中包含 10 种音乐流派，每种流派各有 100 条轨道。所有音轨都是 22,050Hz、单声道、16 位的 WAV 格式音频文件。这些音乐流派包括：布鲁斯 (blues)、古典 (classical)、乡村 (country)、迪斯科 (disco)、嘻哈 (hiphop)、爵士 (jazz)、金属 (metal)、流行 (pop)、雷鬼 (reggae) 和摇滚 (rock)。

**训练集划分：**训练集、验证集、测试集的数量分别为：630, 70, 300，其中，训练集和测试集包含来自 10 个风格数量基本相同的音乐片段。

#### 模型训练情况

**损失函数：**选择均方误差损失函数 (MSE Loss)，用于刻画输入和输出之间的误差。

**优化器：**选择 Adam 优化器来更新模型参数，进行训练。

图15中展示了模型在训练和验证过程中的损失变化曲线。横轴表示训练的轮数 (Epoch)，纵轴表示平均损失值 (Average Loss)。

训练损失 (Train Loss): 图15中蓝色曲线表示训练数据集的损失变化情况。可以观察到，随着训练轮数的增加，训练损失迅速下降并趋于稳定。在最初几轮训练中，损失值显著下降，表明模型在快速学习数据的特征。随着训练的进行，损失下降速度减慢，最终趋于平稳，表明模型在训练数据上的拟合效果良好。

验证损失 (Valid Loss): 图15中橙色曲线表示验证数据集的损失变化情况。验证损失在初始阶段也迅速下降，并在随后的训练过程中呈现出一定的波动。这种波动可能是由于验证数据集的复杂性或者模型在不同数据上的表现差异导致的。总体上，验证损失也呈现下降趋势，表明模型在未见过的数据上有较好的泛化能力。

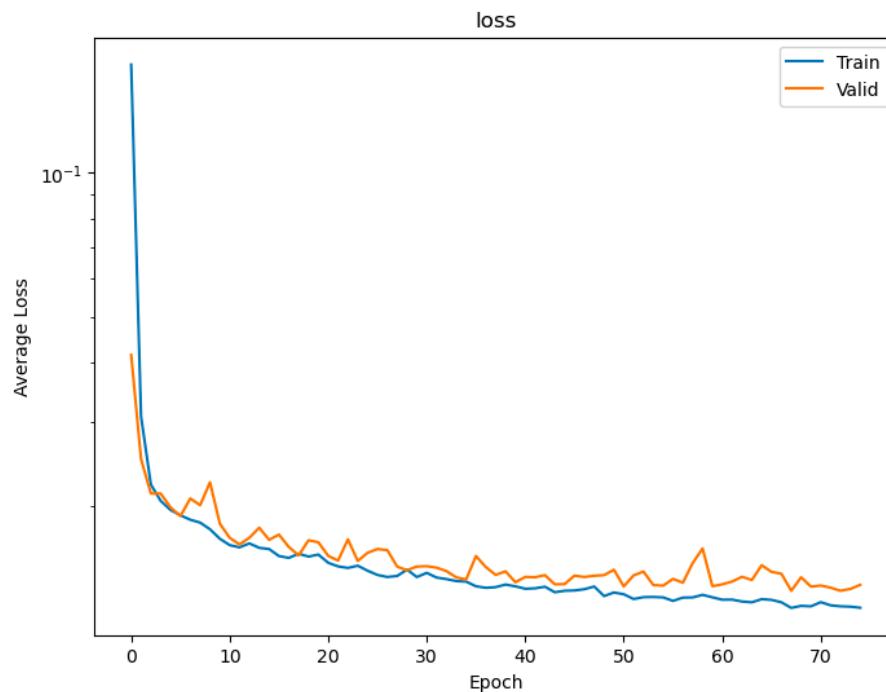


图 15 训练损失

从图15中可以看出，训练损失和验证损失在训练过程中都显著下降，且在训练轮数增加后趋于平稳。这表明模型在训练数据和验证数据上都取得了较好的表现，且没有明显的过拟合或欠拟合现象。

#### 4.2.5 模型效果评估

使用训练轮数不同的自编码器对梅尔频谱图进行还原，得到输出与原图进行对比。由于解码器通过隐向量对梅尔频谱图进行重建，所以还原效果可以体现隐向量对音频的代表性。

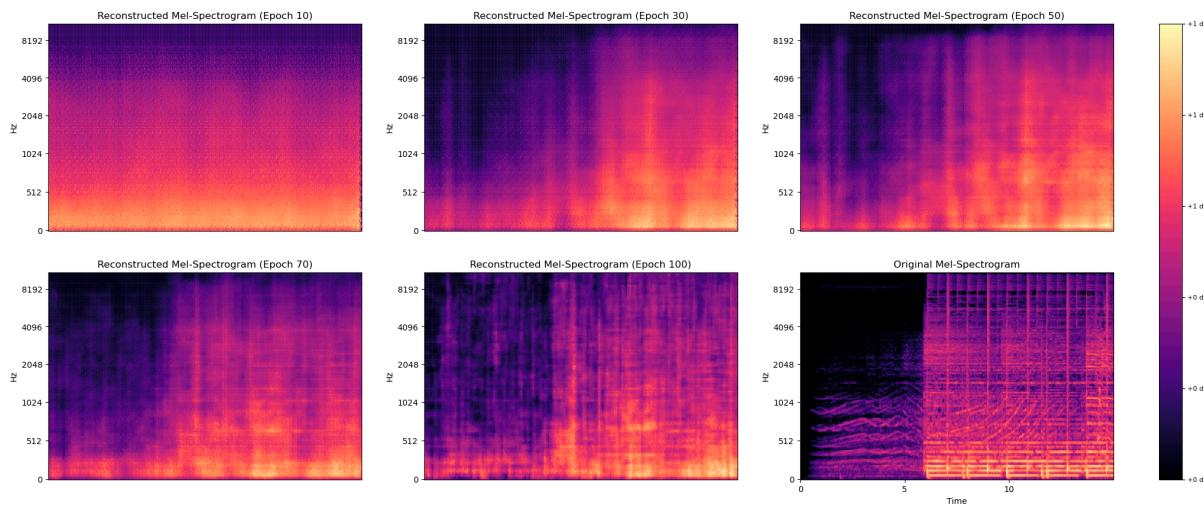


图 16 自编码器重建梅尔频谱图

图16中展示了自编码器在不同训练轮数下对输入梅尔频谱图的重建效果。随训练轮数的增加，重建的梅尔频谱图逐渐接近原始输入。

可以观察到：在训练初期 (epoch 10)，重建的梅尔频谱图仍然比较粗糙，细节较少。高频部分的重建较差，低频部分逐渐显现一些特征。随着训练的进行 (epoch 30-70)，重建效果显著改善。图像的整体结构开始显现，高频和低频部分的特征更加清晰。可以看到一些原始梅尔频谱图中的纹理和细节被捕捉到了。训练完成时 (epoch 100)，重建效果进一步提高，图像中的细节更加丰富。高频部分的细节逐渐清晰，低频部分的纹理特征也更加明显。重建的梅尔频谱图与原始输入图像较为相似。

通过图16中的系列图片可以看出，自编码器在训练过程中逐渐学习到输入梅尔频谱图的特征，并且能够越来越准确地进行重建。这表明模型在训练过程中不断优化，其重建能力得到了明显提高，也反映出特征向量对音频的代表性越来越好。

训练完成的编码器即作为我们的 Echoes 模型，用于后续工作

## 5 基于音乐特征向量的系列任务

### 5.1 Echoes 模型对音乐风格的捕捉

#### 5.1.1 音乐特征向量聚类与可视化

参考 [19] 中对隐藏层的处理，我们对4.2.4中划分的测试集进行编码后的特征向量进行了 K-Means 聚类，然后使用 PCA 进行可视化

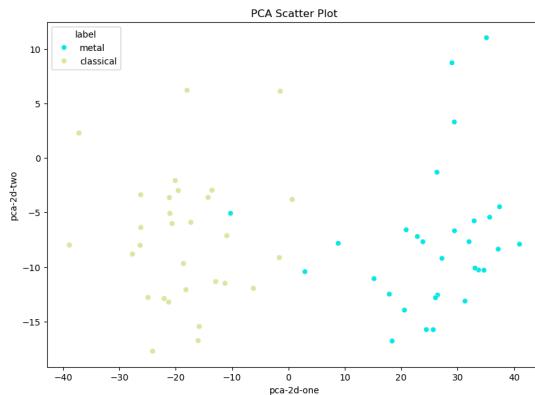


图 17 Metal - Classical

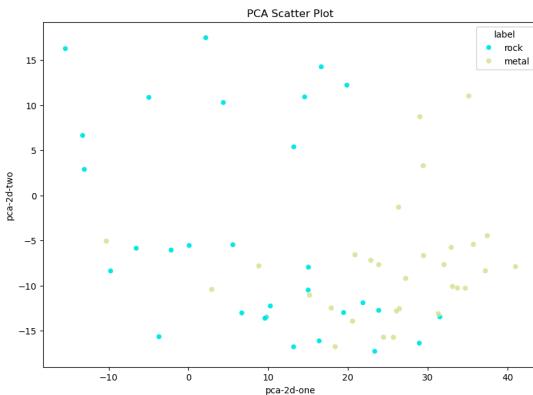


图 18 Metal - Rock

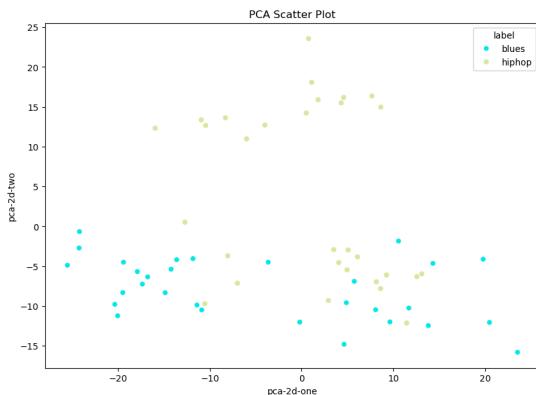


图 19 Blues - Hip-Hop

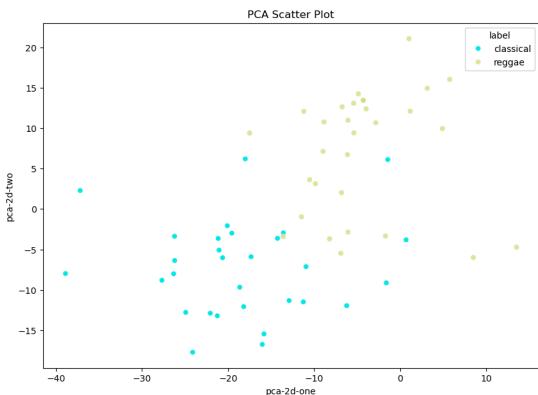


图 20 Classical - Reggae

PCA 将高维特征向量投影到二维空间，以便观察不同音乐风格之间的分布和区分情况。图中展示了四种不同风格组合的 PCA 散点图。

1. 图17-Metal 与 Classical, 图20-Classical 与 Reggae: 这两组风格在二维空间中分布较为分散，能够明显区分开来，说明模型在这两种风格上提取的特征具有较好的辨别能力。
2. 图18-Rock 与 Metal, 图19-Blues 与 Hip-hop: 总体上可以看到一定的区分度，但两组风格在某些区域有重叠。而根据对音乐风格的认知，Rock 和 Metal 本身就有很密切的关系，尤其是在 Hard Rock 等子风格下，由于其与 Metal 是同源的，所以差距很小；同样，Blues 音乐和 Hip-Hop 的密切关系也在图中得到了很好的反映。

从上述 PCA 散点图可以看出，Echoes 模型能够有效地提取音乐的特征，并在不同

音乐风格之间进行较好的区分。虽然在风格的“边界”处之间存在一定的重叠，但也能符合对音乐风格的传统认知。通过这些可视化结果，可以认为 Echoes 模型能够捕捉到音乐中具有辨识度的特征，为后续的音乐分析和聚类任务提供了有力支持。

### 5.1.2 无监督音乐分类

与直接预测音乐的风格标签不同，这里尝试直接对歌曲的特征向量使用 K-Means 聚类。对于所有风格，把它们分别赋予给含其比例最高的类。通过这种方式，可以得到一个完全基于音乐的“自发”的分类结果。图21的主对角线代表分类正确的情况，可以看到金属（Metal）和古典（Classical）有着不错的准确率。通过图22可以看到，迪斯科（Disco）和流行（Pop），蓝调（Blues）和嘻哈（Hip-Hop）、摇滚（Rock）和金属（Metal）交叉较为明显，这点与前文5.1.1中的讨论也是一致的。在这种任务中，Echoes 模型的“分类”效果虽不能与专门针对音乐风格分类的模型相比，但可以看到 Echoes 模型在音乐相关任务中的泛化性。

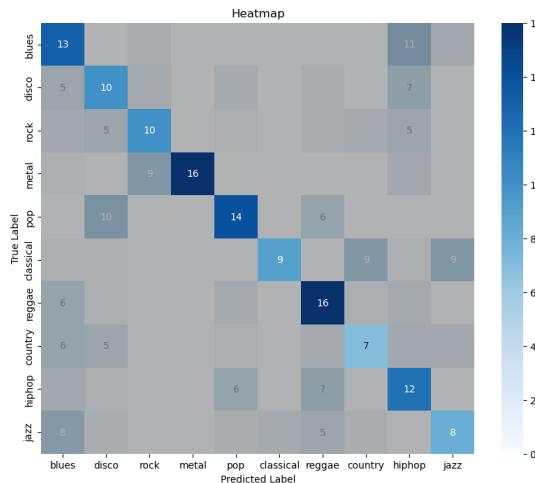


图 21 正确风格标签

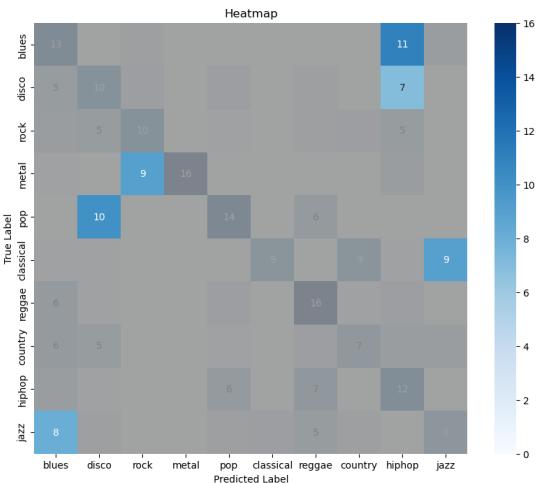


图 22 错误风格标签

同时，这一结果在某种程度上也印证了前文对“风格分类”的理解，即风格的形成与聚类的过程有着一定的相似性。

## 5.2 音乐特点与组成变化分析

针对一个艺术家或给定范围内的歌曲，使用 Echoes 模型对歌曲进行编码，基于编码后的歌曲，我们可以进行系列工作。下面将以乐队 Pink Floyd 的作品为例，展示基于编码后获得的特征向量对乐队曲风变化的分析。

数据集：Pink Floyd 1967-2014 年录音室专辑音频（数量：170，来源：购买自数字音乐平台 [HDTracks](#) 并遵循使用规则；格式：.mp3 音频文件）

对于长度不同的歌曲，我们以 15s 为一个单元，将其切分成多个片段（舍弃最后不足 15 秒的片段）。将每个片段输入 Echoes 模型，得到对应的特征向量，然后根据片段

的音量进行加权平均（假设整体音量越高，包含的信息越多）。进而我们可以得到包含有歌曲元数据以及特征（Echoes 模型的 128 维特征向量和 MFCC 的 20 维特征向量）的数据集，如图23。

Artist	Year	Album	Song	Cluster	Echoes_128	MFCC_20
Pink Floyd	1975	Wish You Were Here	Have A Cigar	0	[2.9370487 -0.146 -2.0348624e+02 1.5]	
Pink Floyd	1975	Wish You Were Here	Shine On You Crazy Diamond, Pts. 1-5	3	[2.2825460e+00 2 -2.7404300e+02 1.6]	
Pink Floyd	1975	Wish You Were Here	Wish You Were Here	3	[2.8893247e+00 5 -300.8528 140.20]	
Pink Floyd	1975	Wish You Were Here	Shine On You Crazy Diamond, Pts. 6-9	3	[3.0897217 0.356 -2.6079218e+02 1.7]	
Pink Floyd	1975	Wish You Were Here	Welcome To The Machine	0	[2.1391609 -0.356 -2.3477303e+02 1.5]	
Pink Floyd	1977	Animals	Sheep	6	[3.0341606e+00 6 -2.6910693e+02 1.7]	

图 23 编码后的数据集

在此基础上，我们使用 K-Means，针对Echoes\_128进行聚类，图24与图25展示了使用不同模型编码后的特征向量进行聚类的结果：可以观察到：Echoes 模型聚类后，数据点在二维主成分分析（PCA）空间中的分布相对均匀，形成明显的簇结构，说明 Echoes 模型特征能够较好地捕捉数据之间的相似性和差异性。而相比之下 MFCC 模型聚类后数据点较为集中，簇结构不明显。可见使用 Echoes 模型得到的特征向量在捕捉音乐数据特征和实现有效聚类方面展示了更好的性能。

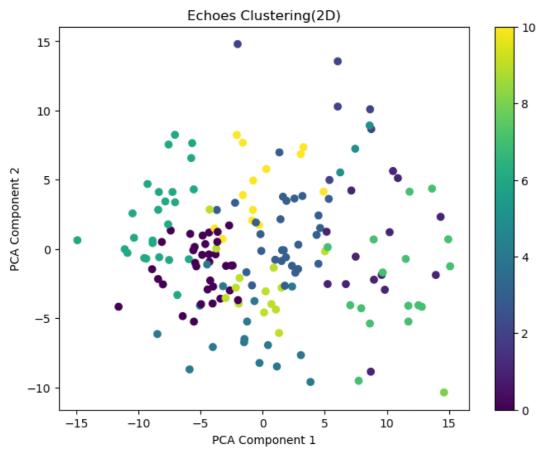


图 24 Echoes 模型编码特征向量

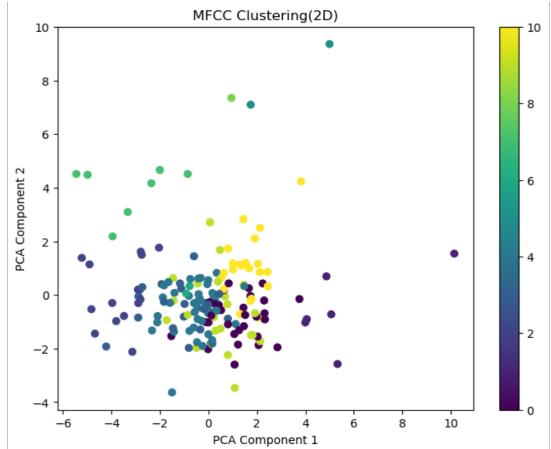


图 25 MFCC 编码特征向量

统计乐队不同年份属于各个类的歌曲占比，如图26。结合图23中的元数据可知，Cluster 0,9 代表了 Pink Floyd 成员 David Gilmour 的主要作品，Cluster 6 代表了乐队成员 Syd Barrett 的主要作品，Cluster 3,4 代表了乐队成员 Roger Waters 的主要作品。结合图26以及将乐队成员的类凸显后的图27,28,29，我们可以观察到在 1969 年、1983 年两个节点附近，专辑的类组成发生了较大的转变，这与乐队成员变动相吻合（1969 年 Syd Barrett 离开，同时期 David Gilmour 加入；1983 年 Roger Waters 离开）。

通过这个任务，我们可以认为 Echoes 模型对音乐内容有着较深入的挖掘，能够发现“风格中的风格”即统一乐队内部的“微风格”。这在未来可以成为想要系统了解一个乐队的乐迷或传记作者等的一个有力工具。

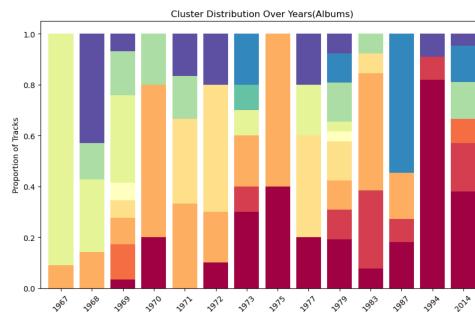


图 26 聚类-年份/专辑

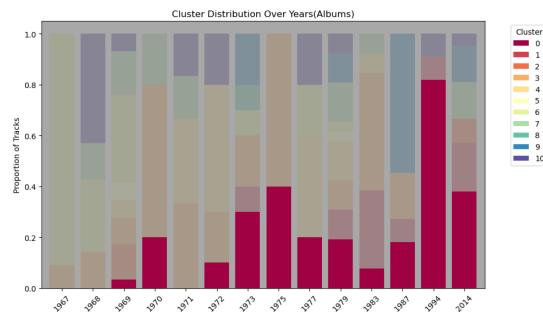


图 27 David Gilmour

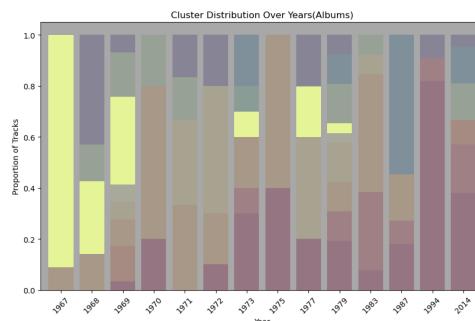


图 28 Syd Barrett

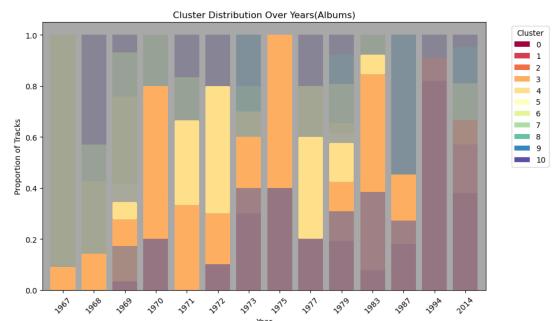


图 29 Roger Waters

### 5.3 基于音乐内容的歌曲推荐

前面针对 Echoes 模型编码后的歌曲进行 K-Means 聚类可以得到不错的效果，因此可以认为特征向量之间的欧氏距离可以刻画歌曲与歌曲之间的“相似度”。基于这点，我们还进行了“平均歌”的计算和简易的歌曲推荐系统。

#### 5.3.1 Echoes 模型和乐迷眼中的“平均歌”

“平均歌”，是指距离所有歌曲特征向量的均值向量最近的歌曲。我们认为这样的“平均歌”能够在一定程度上代表乐队的所有作品。使用 Echoes 模型对 Pink Floyd 的歌曲进行编码后，我们得到前十首平均歌，如图30。为了评价这个结果的效果，我们在有大量 Pink Floyd 乐迷的[论坛](#)上发送了[帖子](#)，询问乐迷们心目中的“平均歌”，收集到了 80 条评论，其中认同歌曲“Time”的回答得到了最高的支持（100+），认同歌曲“Echoes”，“Shine On You Crazy Diamond, Pts. 1-5”的其次，同时，歌曲“Mudmen”，“Wearing the Inside Out”，“Fearless”也均有被提及。可以认为，通过 Echoes 模型得到的“平均歌”与歌迷心目中的“平均歌”能达到一定程度的吻合的。

#### 5.3.2 曲库中的相似歌曲匹配/推荐

这一部分是“对 Echoes 模型的性能评估” 4.2.5 部分的延续，考虑到主观上的推荐并不好呈现，因此采用本身相似的歌曲进行测试，这里的测试主要基于 Pink Floyd 的音乐中有许多“系列”歌曲，如 Shine On You Crazy Diamond, Pts. 1-5, Pts. 6-10; Pigs

Song	Album	Year
Echoes	Meddle	1971
Shine On You Crazy Diamond, Pts. 1-5	Wish You Were Here	1975
Waiting for the Worms	The Wall	1979
Mudmen	Obscured by Clouds	1972
Wearing the Inside Out	The Division Bell	1994
Fearless	Meddle	1971
Shine On You Crazy Diamond, Pts. 6-9	Wish You Were Here	1975
The Trial	The Wall	1979
The Hero's Return	The Final Cut	1983
Time	The Dark Side Of The Moon	1973

图 30 “平均歌”

On The Wing (Part One), (Part Two), 等等；同时，还会对一些歌曲的 demo 版本和 live 版本进行测试（在曲库之外）。如果能够较好的匹配到原曲，我们则可以认为 Echoes 模型在推荐相似乐曲的任务中可以有较好的表现。

下面还展示了基于 MFCC 模型的推荐（同样基于欧氏距离），作为对比

### 匹配效果

如图31，选取 Pink Floyd 的四首有代表性的“系列”录音室版歌曲的一部分作为输入，可以看到 Echoes 模型可以很好的匹配到同一系列的歌曲（这里系列是指在音乐内容上具有很相似的特点，但是两首不同的歌曲），相比之下，MFCC 模型则略显逊色。

target_song	Echoes	MFCC
Another Brick in the Wall, Pt. 2		
	Another Brick in the Wall, Pt. 3	Eclipse
	Wearing the Inside Out	Hey You
	TBS9	Your Possible Pasts
	Run Like Hell	Two Suns In The Sunset
In the Flesh		
	In the Flesh	Dogs
	Have A Cigar	Welcome To The Machine
	The Nile Song	High Hopes
	Summer '68	In the Flesh
Shine On You Crazy Diamond, Pts. 1-5		
	Shine On You Crazy Diamond, Pts. 6-9	Mudmen
	The Fletcher Memorial Home	Shine On You Crazy Diamond, Pts. 6-9
	The Hero's Return	Echoes
	Alan's Psychedelic Breakfast	Cymbaline
Pigs On The Wing (Part One)		
	Pigs On The Wing (Part Two)	Pigs On The Wing (Part Two)
	The Gnome	A Spanish Piece
	Dogs	Alan's Psychedelic Breakfast
	Burning Bridges	The Grand Vizier_s Garden Party (Entertainment)

图 31 系列歌曲查找

如图32，选取 Pink Floyd 的三首现场版和一首 demo 版本进行测试，可以看到 Echoes 模型依然有着不错的表现，而 MFCC 模型则显出不足。

target_song	top	Echoes	MFCC
The Doctor (Comfortably Numb)			
	1	Childhood's End	Wot's...Uh the Deal
	2	Comfortably Numb	Side 4, Pt. 4 Louder Than Words
	3	Burning Bridges	When You're In
	4	TBS14	The Show Must Go On
	5	Wot's...Uh the Deal	Another Brick in the Wall, Pt. 3
Shine On You Crazy Diamond (Parts 1-5)			
	6	TBS14	Yet Another Movie
	7	Wearing the Inside Out	Money
	8	Shine On You Crazy Diamond, Pts. 1-5	A Great Day for Freedom
	9	Learning To Fly	Side 3, Pt. 6 Allons-y (2)
	10	Side 3, Pt. 7 Talkin' Hawkin'	Coming Back to Life
Comfortably Numb			
	6	On The Turning Away	Money
	7	Keep Talking	The Nile Song
	8	Side 3, Pt. 7 Talkin' Hawkin'	The Dogs Of War
	9	Comfortably Numb	Nervana
	10	TBS14	Side 3, Pt. 4 Allons-y (1)
Money			
	1	Side 3, Pt. 4 Allons-y (1)	Side 4, Pt. 3 Surfacing
	2	On The Turning Away	Keep Talking
	3	Money	Money
	4	Side 4, Pt. 3 Surfacing	Learning To Fly
	5	Side 4, Pt. 4 Louder Than Words	On The Turning Away

图 32 live/demo 歌曲查找

这里需要指出，对于大部分的现场版歌曲，两个模型均没有太好的表现，但是在部分歌曲上，可以看出 Echoes 模型明显强于 MFCC 模型。这点我们认为，由于乐队在进行现场演奏的时候会进行很多演绎，所以与原曲内容和形式上都有很多不同，并且现场版的录音中往往有很多人群的声音，会对于模型判断会产生很强的干扰。

## 6 未来工作与讨论

### 6.1 未来工作

1. 模型选择：和单纯的图像任务不同，音乐数据是具有天然的时序性的，因此，我们考虑在卷积自编码器之外，能否使用一些更善于序列数据的网络结构，比如循环神经网络（RNN）、长段记忆网络（LSTM）、变压器（Transformer）等，从而更好的提取歌曲的特征？
2. 音乐之间的“距离”：本次工作选取了与 K-Means 相同的刻画距离的方法，欧氏距离，在后续工作中，可以考虑其他的刻画“距离”的方法，如余弦距离，进行比较。同时，有了音乐之间的距离刻画，我们可以对距离与相似度的关系进行量化，可应用到曲库内容的网络分析，以及专辑与专辑、艺人与艺人的关系图谱搭建。
3. 变长数据处理：即针对不同的歌曲的特征向量处理应采用什么方式？本次工作中 MFCC 模型采取了针对特定位置附近的音乐片段向量的加权平均；Echoes 模型则

是对音量进行加权平均。这样的处理其实略显草率，在后续工作中将继续探索不同的卷积核对效果的影响。

4. 音乐之外：人们“听”音乐其实不只是听音乐本身，很多时候歌词、专辑封面等因素都会影响人们对一首歌曲的认识。因此计算机想要“听”音乐，也应结合这些非音乐数据，未来能否结合更多的模型实现一个多模态编码器？

## 6.2 讨论

1. 人脑对音乐的处理，是一个特征提取即“降维”的过程，还是以某种更复杂、更抽象的形式“储存”在了大脑里？还是二者的结合？

首先，在听音乐时，耳朵首先将声波转换成电信号，然后这些信号被传递到大脑的听觉皮层。在这里，大脑会进行初步的特征提取，包括音高、音量、节奏和音色等基本特征的分析。这一过程类似于降维，因为大脑将复杂的声波信号转化为更易处理的特征向量。

但是，音乐体验不仅仅停留在这些基本特征的层次。大脑还会进一步处理这些特征，将其组合成复杂的模式和结构。这个过程包括：1. 模式识别：识别旋律、和声、节奏模式等。这涉及到大脑对音乐中重复和变化部分的理解。2. 情感和联想：音乐往往会引发情感反应和记忆联想，这些都涉及到大脑的更高层次处理，包括情感中心（如边缘系统）和记忆中心（如海马体）的参与。3. 语境理解：理解音乐的语境，如文化背景、历史时期和个人经历等，这也需要大脑进行复杂的抽象思维。

因此人脑对音乐的处理很有可能既包括特征提取（降维），也包括将这些特征整合成更复杂、更抽象的表示形式。这些表示形式不仅包含音乐的基本特征，还包含情感、联想和语境等更高层次的信息。这样复杂的多层次处理使得音乐能够在大脑中以丰富且有意义的方式储存和回忆。

理解人脑对音乐的处理方式可以为计算机“理解音乐”提供重要的启示和指导。

2. 在声音，尤其是音乐相关的工作中，最大的困难在于数据集，更确切地说，是**版权问题**。很难获得大量正规且可用的音乐数据；因此，音乐相关的工作通常集中在拥有音乐版权的企业（如各大流媒体公司，包括 Apple Music、Spotify 和腾讯音乐）进行。
3. 人，尤其是个体，听歌的数量和范围是有限的，因此对音乐整体的认知必然是“有偏”的。而在一定程度上，正是这些偏差在一定程度上构成了不同的音乐“品味”。如果有一个能够像人一样感知和理解音乐的模型，它可以接触到的音乐数量将远

超任何个人或群体。这样一个模型是否会对音乐的整体理解把握得更好呢？

对我个人而言，当我听音乐的时候，我不仅仅是在听音乐本身，更是在聆听某个年代、一个时代、一种文化。这种体验是一个基于音乐内容的模型无法完全实现的。其次，作为人，听的音乐越多越好吗？对我来说，并不是这样。

德国哲学家、社会学家和音乐理论家西奥多·阿多诺在 1949 年出版的《新音乐的哲学》中预见到大批量机械化生产对文化工业的影响：这会导致艺术被简化为单纯的消遣，而消遣则会被误认为是艺术，二者混淆在一起，最终都沦为廉价的消费商品。随着流媒体创作平台的盛行，现在产生的音乐数量远超以往。越是在这样的时代，我们听音乐时，质量显得远比数量重要。

在此如此庞大的音乐数量面前，未来可能需要的不是“探索式推荐系统”，而是一个“智能化的过滤系统”。流媒体平台在做加法时，可以使用推荐算法和协同过滤等技术，但当真的需要做减法时，脱离音乐本身显然是不可能的。

#### 4. 关于可解释性

首先必须承认，对于 Echoes 模型，几乎没有较客观的可解释性——这当然也是深度神经网络的共同问题或特点。在前面的展示中可以看到，在很多任务上，Echoes 和 MFCC 特征模型有着很相似的表现。然而，对于同样的结果，我们只能说 Echoes 有自己的“方式”。

对于听音乐这件事，我相信人自己也没有一个“可解释”的说法——正如前面中提到的那样，大脑对于音乐的处理是以一种抽象的形式存在的，自身就具有不可解释性和复杂性。

因此这里我想提出一个想法——人所谓的创造性（尤其是在艺术领域）是否正来自于（大脑中）“不可解释”、“复杂”的那部分？反过来说，如果大脑的结构很简单，对于每一个“输出”的结果，在输出之前就知道明确的答案，我们还会认为这样的过程是“创造”吗？我认为这点对于神经网络同样适用。在音乐相关的任务下，我认为可解释性并不太算是神经网络模型的弊端，相反，这种不可解释性在某种程度上可以和人脑进行“匹配”。

当然，Echoes 模型中，这种不可解释性还是带来了一定的问题。例如，在刻画歌曲特征向量之间的距离刻画并没有明确的方式，这点在后续的工作中将继续探究。

#### 5. 关于推荐

当我们使用推荐系统时，通常是希望被推荐到自己“想要的”或“喜欢的”内容。但是，在推荐内容呈现之前，我们往往并不知道自己理想中的内容是什么。当推荐的内容呈现在我们眼前时，我们才会做出反应——喜欢或不喜欢。因此，人对推荐的反馈是后于推荐的。

在这个过程中有两种选择：以用户为主导或以模型为主导。由于推荐系统本身的商业重要性，通常以前者为主，即通过用户的历史行为、偏好等数据来推荐内容。而后者则提出一个新想法：如果模型本身有自己的“推荐的道理”，比如 Echoes 模型基于音乐内容相似性进行推荐，同样可以设计基于音乐内容的不相似性进行“探索”的模型。这引发了一个有趣的问题：是让音乐去找人，还是让人去找音乐？这种思路转变为推荐系统带来了新的挑战和机会。通过结合用户为主导和模型为主导的推荐方式，可以为用户提供更丰富、多样和个性化的音乐体验。这不仅有助于满足用户当前的需求，还能激发用户的潜在兴趣，提升整体用户体验和满意度。同时，这也对模型本身提出了更高的要求——模型必须结合音乐内容，而不能仅停留在音乐的“标签”等表层信息。模型需要深入分析和理解音乐的各个维度，包括旋律、节奏、和声、歌词和情感等，通过多层次的特征提取和复杂的抽象表示，来捕捉音乐的本质特征。这将使得推荐系统不仅能够提供基于历史偏好的推荐，还能进行探索性推荐，帮助用户发现新的音乐类型和风格，从而真正实现“让音乐去找人”和“让人去找音乐”的双向互动。

## 参考文献

- [1] Wikipedia. Music genre. [https://en.wikipedia.org/wiki/Music\\_genre](https://en.wikipedia.org/wiki/Music_genre). Accessed: 05-June-2024.
- [2] Wikipedia. Music information retrieval. <http://en.wikipedia.org/w/index.php?title=Music%20information%20retrieval&oldid=1226344782>. Accessed: 05-June-2024.
- [3] Meinard Müller, Daniel P. W. Ellis, Anssi Klapuri, and Gaël Richard. Signal processing for music analysis. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1088–1110, 2011.
- [4] Hendrik Purwins, Bo Li, Tuomas Virtanen, Jan Schlüter, Shuo-Yiin Chang, and Tara Sainath. Deep learning for audio signal processing. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):206–219, 2019.
- [5] Dongge Li, Ishwar K. Sethi, Nevenka Dimitrova, and Tom McGee. Classification of general audio data for content-based retrieval. *Pattern Recogn. Lett.*, 22(5):533–544, apr 2001.
- [6] D. Pye. Content-based methods for the management of digital music. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, volume 4, pages 2437–2440 vol.4, 2000.

- [7] Sound Field. The golden ratio and fibonacci in music (feat. be smart). <https://www.youtube.com/watch?v=9mozmHgg9Sk>, 2019. Accessed: 5-June-2024].
- [8] Lindasalwa Muda, Mumtaj Begam, and I. Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques, 2010.
- [9] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.
- [10] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML ’08, page 1096–1103, New York, NY, USA, 2008. Association for Computing Machinery.
- [11] Keunwoo Choi, George Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification, 2016.
- [12] Papia Nandi. Cnns for audio classification. <https://towardsdatascience.com/cnns-for-audio-classification-6244954665ab>, 2021. Accessed: 5-June-2024].
- [13] Juhan Nam, Keunwoo Choi, Jongpil Lee, Szu-Yu Chou, and Yi-Hsuan Yang. Deep learning for audio-based music classification and tagging: Teaching computers to distinguish rock from bach. *IEEE Signal Processing Magazine*, 36(1):41–51, 2019.
- [14] Fanny Roche, Thomas Hueber, Samuel Limier, and Laurent Girin. Autoencoders for music sound modeling: a comparison of linear, shallow, deep, recurrent and variational models, 2019.
- [15] Anastasia Natsiou, Luca Longo, and Sean O’Leary. An investigation of the reconstruction capacity of stacked convolutional autoencoders for log-mel-spectrograms, 2023.
- [16] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression, 2022.
- [17] Andy M. Sarroff and Michael A. Casey. Musical audio synthesis using autoencoding neural nets. In *International Conference on Mathematics and Computing*, 2014.
- [18] George Tzanetakis, Georg Essl, and Perry Cook. Automatic musical genre classification of audio signals, 2001.
- [19] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis, 2016.