

基于递归神经网络的自动作曲算法

李雄飞¹, 冯婷婷², 骆实¹, 张小利¹

(1. 吉林大学 计算机科学与技术学院,长春 130012;2. 吉林大学 软件学院,长春 130012)

摘要:提出了一种以音乐音频为处理对象的基于长短时记忆递归神经网络的音乐自动合成算法。该算法首先将音乐集以基于单位时长分割成由单位音乐序列组成的音乐序列集,并在对音乐进行预处理时提取了音乐音频的梅尔倒谱系数作为特征;其次,将进行数据处理过的特征向量构成的训练样本通过长短时记忆模型进行训练和预测;最后,将生成的音乐序列进行拼接融合从而得到新的音乐。为了验证算法的有效性,将模型生成的乐曲与人为作曲的乐曲进行了匿名打分评价,实验结果表明,该算法能够较好地实现自动作曲。

关键词:人工智能;递归神经网络;自动作曲算法;长短时记忆模型

中图分类号:TP181 **文献标志码:**A **文章编号:**1671-5497(2018)03-0866-08

DOI:10.13229/j.cnki.jdxbgxb20170509

Automatic music composition algorithm based on recurrent neural network

LI Xiong-fei¹, FENG Ting-ting², LUO Shi¹, ZHANG Xiao-li¹

(1. College of Computer Science and Technology, Jilin University, Changchun 130012, China; 2. College of Software, Jilin University, Changchun 130012, China)

Abstract: In this paper, we propose an automatic music composition algorithm based on Long Short Term Memory-Recurrent Neural Network (LSTM-RNN). In this algorithm, we first divide music set into set which consists sequences of unit by length, and in the preprocessing we get the Mel frequency cepstrum coefficient as the feature of audio music. Second, the proposed training samples are trained and predicted by LSTM-RNN. Finally, the generated music sequences are joined to get a new music. In order to verify the effectiveness of the algorithm, we carry out an anonymous evaluation of the original music and the music generated by the algorithm. The results show that the algorithm can work well on automatic music composition.

Key words: artificial intelligence; recurrent neural network; automatic music composition algorithm; long short term memory model

0 引言

近年来利用计算机技术分析音乐情感的特

性,以及以乐谱为研究对象的计算机作曲已成为计算机音乐的一大重要研究方向^[1-5]。计算机算法是通过某种策略控制生成音符序列,进而组成

收稿日期:2017-05-18.

基金项目:中国博士后科学基金项目(2017M611323).

作者简介:李雄飞(1963-),男,教授,博士生导师. 研究方向:机器学习,信息融合,图像处理. E-mail:lxf@jlu.edu.cn

通信作者:张小利(1987-),男,助理研究员,博士. 研究方向:机器学习,信息融合. E-mail:zhangxiaoli@jlu.edu.cn

音乐旋律,最终得到完整乐谱,此类方法需要大量音乐知识规则。而以音频为研究对象进行人工智能(Artificial Intelligence, AI)作曲,可使计算机自动生成音乐片段的排列组合生成新的音乐音频。基于音频的AI作曲不依赖大量的音乐知识规则,又能传递给听众直观感受。因此,该类算法比基于乐谱的传统作曲方法更具有实用性。本文以音乐音频作为研究对象,基于长短期记忆递归神经网络提出一种新的自动合成乐曲算法。

在传统计算机作曲方面,神经网络曾一度被认为不能学习到音乐的结构并且不适合用来做音乐作曲等研究,为解决该问题,Eck 等^[6]用了两个长短期记忆(Long short term memory, LSTM)模型来训练学习创作蓝调音乐,一个用于学习和弦,另一个用于学习旋律,和弦网络的输出连接到旋律网络作为旋律网络的输入。最终实验结果表明,系统能够学习标准的 12 小节蓝调和弦小节并且生成遵循和弦规律的音乐。此后,Franklin^[7]也使用 LSTM 网络来学习和训练爵士音乐。他们开发了一种在主歌和副歌三等分的音调表示方法。在此基础之上,Liu 等^[8]又使用了递归神经网络(Recurrent neural network, RNN)来学习了古典音乐,采用由 Allan&Williams 收集的巴赫的 midi 片段数据集,他们首先验证了神经网络在重组音乐的能力,将神经网络学习重组的音乐片段与原始的巴赫音乐片段进行比对,此后进一步对使用神经网络利用音乐碎片进行谱曲,在验证方面,采用多分类的测试指标对实验结果进行测试,最终测试结果表明,与人类感官有着比较大的差距。

在传统研究中,梅尔倒谱系数(Mel frequency cepstrum coefficient, MFCC)在音乐信号上能够高效地识别音调和频率,一直被用作分析音乐音频,Dhanalakshmi 等^[9]采用 MFCC 和线性预测编码(Linear predictive coding, LPC)分别作为音频分类的特征向量,使用支持向量机通过训练将音频进行场景分类,结果证明 MFCC 作为特征向量时分类精度更高。Mathieu 等^[10]在 GNU (General public licence)通用公共许可协议下开发了一个音频特征提取的系统 YAAFE 用于快速提取音频特征。而 AI 作曲又是以乐谱为载体进行研究,实质的研究为文本挖掘类研究,而本文首次提出以音频本身作为研究对象,从 MFCC 入手,将音频信号处理与 AI 作曲融合,提

出了基于 LSTM-RNN 的音乐音频自动合成算法,验证了 AI 作曲以音频为载体的可能性,使得成果更直观地展现于听众。

1 与音乐生成模型相关的形式化描述

单一的音符是没有意义的。从乐理上讲,一个曲子可划分为若干个小节,每个小节由一系列音符组成,因此,一个音乐小节是表达含义的最基本单位,将这些小节有机地组织起来才能体现出音乐情感和含义价值。著名的例子是莫扎特的圆舞曲《音乐骰子游戏》,他创作了 176 个音乐小节,然后将小节编号排列为两个特别的矩阵图,用掷骰子的方式来决定演奏的次序,每次掷骰子都是这些音乐片段的重组。本文将这样的可重复组合排序的一个或多个小节称为音乐模式,将大量音乐曲目分解为音乐模式,构成音乐模式库。这样,基于 AI 的音乐创作就可分为两个步骤:①在大量乐曲上训练音乐模型;②利用音乐模型从音乐模式数据库中抽取音乐模式组成乐曲。

在音乐中,时间被分成均等的单位,每个单位称为一拍,拍子的时值以音符的时值来表示,而拍子又有强弱拍之分,强拍之间的连接以小节来划分,以 3/4 拍的乐曲为例,以四分之一音符为一拍,每个小节有 3 拍,强弱拍顺序为强、弱、弱,当乐曲的规定速度为每分钟 60 拍时,每拍占用的时间为 1 s,3 s 为一个小节。因此,当音乐规定的速度为 n 拍/min 时,每 t 秒可以获取到 $\lceil \frac{n}{60}, \frac{t}{3} \rceil$ 个小节,称时长为 t 的这段音乐为单位音乐,显然这样的单位音乐是有意义、有旋律的音乐模式,则音乐的创作问题就变成了单位音乐的重组问题。

定义 1 单位音乐与音乐向量

若音乐音频 M 的时长为 T ,取时长 t 对 M 进行分割,则 $M = (m_1, m_2, m_3, \dots, m_i) i \in [1, \frac{T}{t} + 1]$,称 m_i 为单位音乐, t 为单位音乐 m_i 的单位时长。任意单位音乐 m_i 的特征用特征向量 v_i 刻画,则 v_i 称为该单位音乐的音乐向量。

定义 2 前序信息

对于一段乐曲中任意一个单位音频 m_i ,与其有时间顺序的前 n ($n < i$) 个单位音乐(子)序列 $(m_{i-n}, m_{i-n+1}, \dots, m_{i-1})$ 称之为单位音乐 m_i 的前序信息,记作 $pre(m_i)$ 。

可以把合成算法看成是已知前 $i-1$ 个单位音乐推测第 i 个单位音乐的问题, 其中 $n < i$ 。则第 i 个单位音乐的前序信息为 $\text{pre}(m_i)$, ($\text{pre}(m_i), m_i$) 为一个训练样本, 则预测问题可以表示对关于 $\text{pre}(m_i)$ 的函数 F 的构造问题:

定义 3 AI 生成音乐

针对目标函数 F , 选择一个 m_i 后, 就音乐序列 $M = (m_1, m_2, \dots, m_n)$ 而言, 对任意 $0 < i < n$, 若子序列 (m_1, m_2, \dots, m_i) 使 F 最优, 有序列 $(m_1, m_2, \dots, m_i, m_{i+1})$ 使 F 最优, 则称 M 为 AI 生成音乐。

2 音频预测和音乐合成

2.1 训练数据集组织

选择一批音频乐曲用于构建训练集。将每个乐曲分割为单位音乐序列, 具体步骤如下。

2.1.1 分割单位音乐

在获取单位音乐时, 旨在保留音乐节拍的强弱性以及较短的旋律性, 因此, 若单位时长 t 的取值太小, 会破坏小节的完整性, 则丧失了音乐的强弱节拍感, 若单位时长 t 取值太大, 容易保留过多的旋律信息, 经过试验, 本文取单位时长 $t=3$ s, 当音乐速度为 90~180 节拍/min 时, 单位音乐 m 包含的小节数约为 2~3 小节。音频编码中, 编码流 d_m 与时长有着依赖关系, 依据音乐时长, 将音频流切割成等单位时长的音频片段序列, 式(1)用于切割流数据 $d(t)$:

$$d(t) = dm[0 : fmrt * t] \quad (1)$$

式中: t 为单位时长; $fmrt$ 为该音频文件的采样频率; $d_m[0 : fmrt * t]$ 表示对数据流 d_m 的从下标 0 到下标 $fmrt * t$ 的数据切片。

2.1.2 特征处理

音乐通过影响人的听觉感受以传递情感信息, 实验表明, 人的听觉感受对音调的变化是呈线性变化的。MFCC 通过对频率和音调的对数关系转化反映了人耳的音高听觉特性。在以音频为载体的音乐情感和场景分类问题的研究结果表明, MFCC 在音乐信号上能高效地识别音调和频率, 可作为音频分类的特征^[9]。因此, 本文取 MFCC 作为单位音乐的特征。

常见的 MFCC 为 39 维, 由 13 维静态系数、13 维一阶差分系数以及 13 维二阶差分系数组成, 其中差分系数表示音乐的动态特征, 而 13 维静态系数又是由 1 维能量特征和 12 维系数构成。

MFCC 的计算过程为:

(1) 对每一帧信号做快速傅里叶变换(Fast fourier transform, FFT)计算幅度频谱。

(2) 将幅度频谱利用梅尔尺度变换到梅尔域, 经过等带宽的梅尔滤波器组滤波之后, 将滤波器组的输出能量进行叠加:

$$S_k = \log\left(\sum_{j=0}^{N-1} H_k(j) |X(j)|\right) \quad (2)$$

$$k = 1, 2, \dots, K$$

式中: S_k 为第 k 个滤波器的对数能量输出; $H_k(j)$ 为第 k 个三角滤波器的第 j 个点对应的权值; $|X(j)|$ 为变换到梅尔尺度上的 FFT 频谱幅值; K 为滤波器的个数, 一般为 24 个。

(3) 将滤波器的对数能量进行离散余弦变化, 可以得到 MFCC 系数:

$$c_n = \sqrt{\frac{2}{K}} \sum_{k=1}^K (\log S_k) \cos\left[n(k - 0.5) \frac{\pi}{K}\right] \quad (3)$$

式中: L 为 MFCC 静态系数的维数, 一般 $L \leq P$, 本文取 L 为 13 维。

至此, 令 $\mathbf{V}(m_i)$ 表示第 i 个单位音频 m_i 的音乐向量, 则 $\mathbf{V}(m_i) = (c_{1i}, c_{2i}, \dots, c_{ni})$ 称为单位音乐向量, 其中 $\mathbf{V}(m_i) \in \mathbf{R}$, n 为单位音乐向量的维数。

将提取完 MFCC 的单位音乐向量 \mathbf{V} 进行 Softmax 归一化, 对于 $\mathbf{V}(m_i)$ 中的第 k 个元素 c_k , Softmax 归一化的值为:

$$v_k = e^{c_k} / \sum_{j=1}^n e^{c_j} \quad (4)$$

则归一化后的单位音乐向量表示为 $\mathbf{V}(m_i) = (v_{1i}, v_{2i}, \dots, v_{ni})$ 。

2.2 模型训练与预测

训练样本表示为 $(\mathbf{V}(\text{pre}(m_i)), \mathbf{V}(m_i))$, 令包含了 n 首音乐 M 的数据集 $S = \{M_1, M_2, \dots, M_n\}$, i 为单位音频 m_i 在数据集 S 中的索引。则, 对于该模型来说, 输入是单位音频 m_i 的前序音乐序列 $\text{pre}(m_i)$, 形如 $[\mathbf{V}(m_1), \mathbf{V}(m_2), \dots, \mathbf{V}(m_{i-1})]$, 输出是单位音频 m_i 的相似特征向量 \mathbf{h} , 通过计算 \mathbf{h} 与数据集 S 中单位音频的距离确定 m_i 。

该模型目标函数设为 tanh 函数, LSTM-RNN 模型音乐预测问题 $F(\text{pre}(m_i); \theta)$ 问题可表示为参数集 $\theta = (W, U)$ 的函数构造问题:

$$F(\text{pre}(m_i); W, U) = h_i \quad (5)$$

$$h_i = o_i \tanh(c_i) \quad (6)$$

式中: o_i 表示LSTM模型中的输出门,令 V_i 表示第*i*时刻的前序信息 $pre(m_i)$ 的音乐向量 $V(pre(m_i))$, φ 表示sigmoid函数或tanh函数,则有:

$$o_i = \varphi(W_o V_i + U_o h_{i-1}) \quad (7)$$

令 I_i, f_i 分别表示LSTM模型的输入门和遗忘门, c_i 表示LSTM的记忆单元,则LSTM中单位时间*i*的记忆单元 c_i 经过输入门 I_i 和遗忘门 f_i 调整为新的内容 \tilde{c}_i 和早期的记忆内容 c_{i-1} 之和:

$$c_i = f_i c_{i-1} - I_i \tilde{c}_i \quad (8)$$

$$\tilde{c}_i = \tanh(W_c V_i + U_c h_{i-1}) \quad (9)$$

输入门 I_i 和遗忘门 f_i 分别控制新内容的输入和旧内容的遗忘:

$$I_i = \varphi(W_I V_i + U_I h_{i-1}) \quad (10)$$

$$f_i = \varphi(W_f V_i + U_f h_{i-1}) \quad (11)$$

当记忆单元进行更新后,隐藏层会根据当前输入门得到的计算结果计算当前隐藏层 h_i ,如式(6)所示。

至此,当W和U确定后,构造函数F也就唯一确定了。在LSTM中通常确定W和U的过程需引入优化函数RMSProp,令 $\theta = (W, U)$,RMSProp的迭代过程如下:

从训练集中随机抽取一批容量为N的样本 $\{V_1, V_2, \dots, V_N\}$,以及其相关的输出 $F(V_i; \theta)$ 及对应单位音乐 m_i 。计算梯度 $\nabla \theta$ 和误差 \hat{g} 并更新 r :

$$\hat{g} = \frac{1}{N} \nabla \theta \sum_i (F(V_i; \theta) - m_i)^2 \quad (12)$$

$$r = \rho r + (1 - \rho) \hat{g} \odot \hat{g} \quad (13)$$

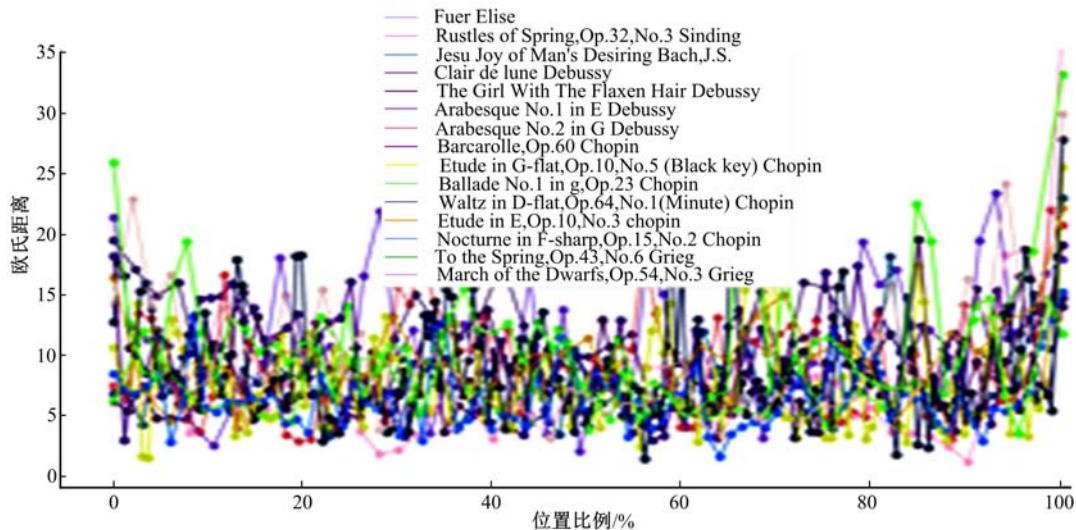


图1 相邻单位音乐向量间距离图

Fig. 1 Distance between adjacent units vector

再根据 r 和梯度 $\nabla \theta$ 计算参数更新量并更新 θ :

$$\Delta \theta = -\frac{\epsilon}{\delta + \sqrt{r}} \odot \hat{g} \quad (14)$$

$$\theta = \theta + \Delta \theta \quad (15)$$

式中: ϵ 为学习率; δ 为数值稳定量; ρ 为衰减速率。

由于音乐曲目通常在首尾两处单位音乐的MFCC与中间主体部分距离相差悬殊,图1展示了15首音乐的单位音乐特征的相邻两向量的距离,因此,分别将首尾部分的2个单位音频取出放入集合 Sh 和 St ,而其余部分作为乐曲主体放入集合 Sb ,即,对于一首时长为T的音乐 $M = (m_1, m_2, \dots, m_k), k = T/t+1$ 有 $m_1, m_2 \in Sh; m_3, \dots, m_{n-2} \in Sb; m_{n-1}, m_n \in St$,则数据集 $S = Sh \cup Sb \cup St, S$ 共有 $N = k_1 + k_2 + \dots + k_n$ 个单位音乐。在音乐合成中,首先从集合 Sh 中随机挑选出一条单位音频 m_1 作为输入, h 作为输出,然后不断将算法合成的输出 h 与 S 中的单位音乐向量进行相似度匹配,本文中采用的相似度匹配策略是进行欧氏距离计算,距离最近的但为音乐向量即为模型预测的下一条单位音乐 m_{i+1} ,如式(16)和式(17)所示。

$$m_{i+1} = S[x] \quad (16)$$

$$x = \text{index}(\min\{d(h, m_1), d(h, m_2), \dots, d(h, m_N)\}) \quad (17)$$

式中: x 为单位音乐在数据集 S 中的索引; index 为索引函数,取 h 与 S 中所有单位音频 m 的最短距离所对应的单位音频索引; N 为数据集 S 中的

单位音乐总数。

两个单位音频 m_a, m_b 之间的欧式距离 d_{ab} 计算过程如下:

$$d_{ab} = \sqrt{\sum_{j=1}^n (v_{a,j} - v_{b,j})^2} \quad (18)$$

式中: j 表示单位音乐 m 的 n 维向量 \mathbf{V} 的第 j 维向量值。

循环上述过程直到模型找到一首音乐 $m \in St$, 则生成终止, 音乐序列生成完毕。

上述算法过程描述如图 2 所示。

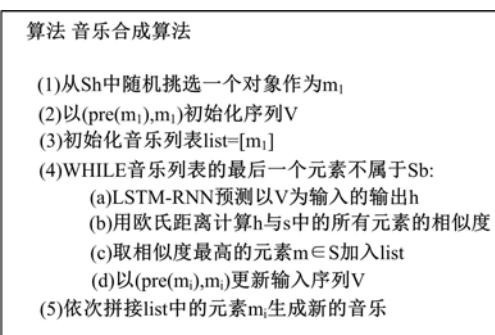


图 2 算法描述

Fig. 2 Algorithm description

2.3 平滑拼接

利用训练得到的 LSTM 模型可生成一个全新的音乐序列 (m_1, m_2, \dots, m_n) 。接下来是音频处理工作, 在对音乐进行拼接时, 相邻的单位音乐需要对音频进行平滑处理使完成后的新曲目显得自然而不突兀, 所以平滑处理的结果直接影响到生成模型最终得到的音乐质量。

在音乐两两连接时选取首尾相接处相同长度, 即相同时间长度的数据流部分, 对数据进行削弱处理, 为保持数据变化的流畅性, 采取对数据进

行线性削弱处理, 其中线性比例系数的计算根据式(19)得到。

$$rate(x; y) = \begin{cases} \frac{l(dm) - x}{l(dm)}, & y = fadeout \\ \frac{x}{l(dm)}, & y = fadein \end{cases} \quad (19)$$

式中: dm 为单位音乐 m 的流数据; x 为当前数据值在数据流 dm 中的索引; 函数 l 为 dm 数据流的格式化数组长度; y 为线性削弱方式, $y \in \{fadeout, fadein\}$, 当 $y = fadeout$ 时, 做比例系数从 1 至 0 的线性削弱计算, 相反, 当 $y = fadein$ 时, 做比例系数从 0 至 1 的线性增强计算。

根据式(20)对数据进行线性衰减计算。

$$fade(x) = \sum_{n=x}^{x+s-1} dm[n] \cdot rate(x) \quad (20)$$

式中: x 为当前数据索引值; s 为采样位数。

时间参数 t 成为了至关重要的参数, 其决定了播放时所能听到的时间长度, 即平滑处理的数据块的大小。

图 3 和图 4 分别展示了当时间值为 1 s 和 2 s 时经过放大后的响度值, 矩形框内为衔接点。从图 4 看出, 当时间值为 2 s 时, 变化范围略长, 依然能明显感觉到淡入淡出的处理感, 使两首曲子连接松散不够紧密, 从而从听觉上能很明确地分辨并不是一首音乐, 而做音轨响度分析时, 从处理后得到的数据部分的音轨响度图可以看到音轨衔接处有明显的长段削弱部分, 与原始音乐频率有很大差异。

而图 3 所展示的时间 $t=1$ 时音乐衔接部分的突兀感减弱而线性变化感也不明显, 在平滑部分得到了比较好的结果, 从听觉上辨别已经不明显, 在平滑部分得到了比较好的处理结果。

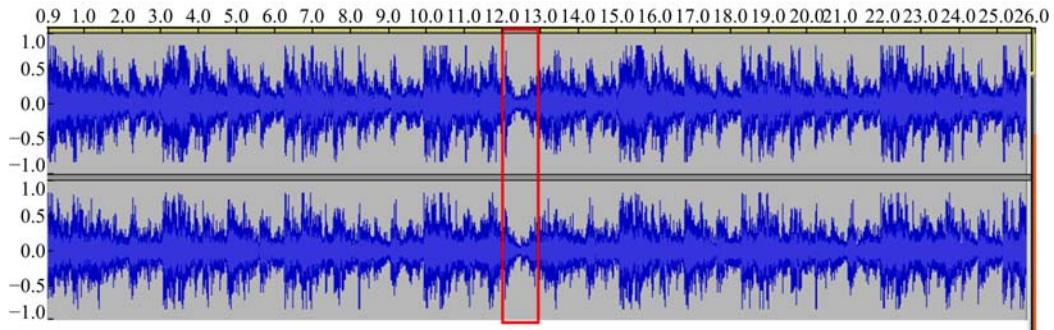
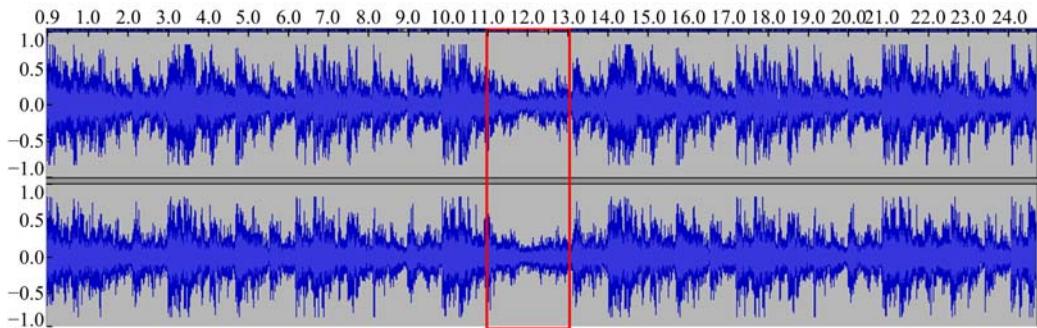


图 3 当 $t=1$ 时的音轨响度图

Fig. 3 Loudness of tracks when $t=1$

图 4 当 $t=2$ 时的音轨响度图Fig. 4 Loudness of tracks when $t=2$

3 实验结果与分析

3.1 实验一

3.1.1 测试数据与参数设置

根据音乐相关理论,古典音乐的情感通常不是固定的,总是激昂与平缓交替出现,这种现象在频谱上表现为能量的突变,本次实验根据这些突变将音乐预先且分为不同情感段,再根据不同的情感段产生的单位音乐集生成不同情感的音乐。

实验数据通过采集 215 首 3/4 拍,演奏速度为每分钟 60~180 节拍的古典音乐的乐谱,通过专业的软件将其输出为音频文件,然后将这些音频文件分割成以 3 s 为单位时长的单位音频共计 15 158 条。根据本文第 1 节的描述,每个分离后的单位音频由 1~3 个小节组成,这样的单位音频则包含了可被重复组合的音乐模式。在 LSTM-RNN 模型中,经过多次实验,训练过程中参数初始化设置如下:

(1) 设置优化函数 RMSProp 的全局学习率 $\epsilon = 0.001$, 初始参数值 $\theta = 0.9$, 数值稳定量 $\delta = 10^8$, 衰减速率 $\rho = 0.0$ 。

(2) 设置神经元连接的断开率为 0.3。

(3) 设置迭代次数为 20 次。

3.1.2 测试实验

在对计算机作曲领域,很难通过客观评价指标去评价效果,所以,一般采用主观法进行测试。例如,Salas 等^[1]进行了基于语言规则的作曲实验并在最终测试时采用类图灵测试,即用户调查的方式,他们从实验结果中选择了 5 首音乐与作曲家创作的 5 首音乐一同构成了测试问卷,并请 26 位测试者参与实验测试,请实验者对他们播放的音乐进行排序。本文将沿用 Salas 等^[1]的测试方法,将实验得出的乐曲与人为创作的乐曲交替

顺序给测试者播放,并且请测试者打分,以及评判喜好,再统计得出测试结果。

本次测试共有 10 首测试音频,其中 5 首来自训练曲库,5 首来自模型生成。共 24 人参与了本次测试的主观评价,其中 11 位学过乐器,10 位表示喜欢古典音乐。在测试中,他们只能看到音乐曲目的序号,其他信息不予显示,测试者试听音乐后,根据主观判断进行打分(0~5 分),0 分则表示不好听,5 分则表示非常好听。测试网站为 <http://47.94.96.142:8000/index/>。测试音频顺序列表如表 1 所示。各首曲子的主观评价直接得分如表 2 所示。

表 1 实验一测试音乐顺序列表

Table 1 Test music list I

序号	音乐名	作者
A	月光	德彪西
B	模型生成 1	算法合成
C	秋练习曲 35 之 2	查米纳德
D	升 f 小调作品前奏曲 28 之 8	肖邦
E	模型生成 2	算法合成
F	模型生成 3	算法合成
G	耶稣,世人仰望的喜悦	巴赫
H	模型生成 4	算法合成
I	致爱丽丝	贝多芬
J	模型生成 5	算法合成

表 2 实验一测试得分及排名结果

Table 2 Rank of test scores(Test I)

序号	排名	音乐名	作者	得分
I	1	致爱丽丝	贝多芬	76
G	2	耶稣,世人仰望的喜悦	巴赫	59
J	3	模型生成 5	算法合成	52
C	4	秋练习曲 35 之 2	查米纳德	50
H	5	模型生成 4	算法合成	43
F	6	模型生成 3	算法合成	43
D	7	升 f 小调作品前奏曲 28 之 8	肖邦	43
A	8	月光	德彪西	41
E	9	模型生成 2	算法合成	37
B	10	模型生成 1	算法合成	33

考虑到乐理知识以及主观喜好的倾向性,将测试人员的打分进行了加权统计,音乐评分通过式(21)进行计算。

$$S = \sum_{k=1}^n (\alpha_k + \sum_{i=1}^m \beta_{ki}) \cdot s_k \quad (21)$$

式中: α_k 为权重; β_{ki} 为加分权重; s_k 为测试人员对该曲目的评分; $k \in [1, n]$ 表示测试人员, $i \in [1, m]$ 表示 m 个加分权重项。权重 α_k 和 β_{ki} 的取值如表 3 所示。

表 3 权重分值表

Table 3 Weight score table

项目	符号表示	权重
学过乐器	β_1	0.5
喜欢古典音乐	β_2	0.5
基础权重	α	1

经计算,各首曲子的主观评价得分如表 4 所示。

通过表 2 与表 4 的统计结果表明,对于原始得分较高、排名靠前的曲目,通常是被大众所喜爱的,所以加权后对其没有造成影响,而群众认知度不高的曲目,在具有乐理基础以及喜好古典音乐的听众与普通测试者中的得分差异性较大,造成加权后的得分排名与原始得分排名有了一些差异。

表 4 实验一测试加权得分及排名结果

Table 4 Weighted rank and scores(Test I)

序号	排名	音乐名	作者	得分
I	1	致爱丽丝	贝多芬	119.5
G	2	耶稣,世人仰望的喜悦	巴赫	93
J	3	模型生成 5	算法合成	86
C	4	秋练习曲 35 之 2	查米纳德	77.5
H	5	模型生成 4	算法合成	68
D	6	升 f 小调作品前奏曲 28 之 8	肖邦	67
A	7	月光	德彪西	66.5
F	8	模型生成 3	算法合成	66
E	9	模型生成 2	算法合成	57.5
B	10	模型生成 1	算法合成	51

结果显示,本算法生成的音乐与人工作曲音乐的排名分布相对均匀,测试人员不能明确区分人工音乐和算法音乐,且在测试人员的打分排名中,模型生成的音乐有一首进入了排名的前三,而排在第一和第二的均是大家非常熟悉的音乐,但是得分末位也是来自本算法,证明算法生成的音

乐质量有差异;另外,该实验结果也说明了在音频处理方面,本实验所采取的拼接算法并不容易让人们发现音乐的拼接点,即在音乐拼接平滑处理方面效果较好。

3.2 实验二

3.2.1 测试实验

音乐是极富个人色彩的作品,为了让本算法更具有灵活性,本文在 3.1 节实验一的基础上增加了交互式计算的部分,在开始生成音乐时,可由使用者指定一个音乐片段作为开头,在音乐声称中间曲目时可由使用者决定是否介入人工选择,如果介入,系统将会在生成 m_i 时,根据 LSTM 的输出 h 与数据集 S 中的单位音乐进行匹配,将提供与 h 距离最短的 3 首单位音乐给使用者进行选择;如果不人工介入,算法默认自动匹配距离最短的单位音乐。加入人机交互部分后的算法流程如图 5。

算法 音乐合成算法

- (1) 从 Sh 中随机挑选一个对象作为 m_1
- (2) 以 $(pre(m_1), m_1)$ 初始化序列 V
- (3) 初始化音乐列表 $list = [m_1]$
- (4) WHILE 音乐列表的最后一个元素不属于 Sb:
 - (a) LSTM-RNN 预测以 V 为输入的输出 h
 - (b) 用欧式距离计算 h 与 S 中的所有元素的相似度
 - (c) IF 人为选取下一个元素 m:
 - (I) 取相似度最高的 3 首音乐 mlist 给使用者提供选择
 - (II) 使用者选择一首音乐 $m \in mlist$ 加入 list
 - (d) ELSE:
 - (I) 取相似度最高的元素 $m \notin S$ 加入 list
 - (e) 以 $(pre(m_i), m_i)$ 更新输入序列 V
 - (5) 依次拼接 list 中的元素 m_i 生成新的音乐

图 5 人机交互式算法描述

Fig. 5 Algorithm description

本次实验选取了 2 首加入交互式计算产生的音乐与加入 3.1.2 节中的城市音乐列表进行对比测试,测试音乐顺序列表如表 5 所示,得分结果如表 6 所示。

表 5 实验二测试音乐顺序列表

Table 5 Test music list II

序号	音乐名	作者
A	月光	德彪西
B	模型生成 1	算法合成
C	秋练习曲 35 之 2	查米纳德
D	人机合成 1	交互式合成
E	升 f 小调作品前奏曲 28 之 8	肖邦
F	模型生成 2	算法合成
G	模型生成 3	算法合成
H	耶稣,世人仰望的喜悦	巴赫
I	人机合成 2	交互式合成
J	模型生成 4	算法合成
K	致爱丽丝	贝多芬
L	模型生成 5	算法合成

表 6 实验二交互式测试加权得分及排名结果
Table 6 Weighted rank and scores(Test II)

序号	排名	音乐名	作者	得分
K	1	致爱丽丝	贝多芬	119.5
H	2	耶稣,世人仰望的喜悦	巴赫	93
L	3	模型生成 5	算法合成	86
I	4	人机合成 2	交互式合成	81.5
C	5	秋练习曲 35 之 2	查米纳德	77.5
J	6	模型生成 4	算法合成	68
E	7	升 f 小调作品前奏曲 28 之 8	肖邦	67
D	8	人机合成 1	交互式合成	66.5
A	9	月光	德彪西	66.5
F	10	模型生成 3	算法合成	66
E	11	模型生成 2	算法合成	57.5
B	12	模型生成 1	算法合成	51

测试结果显示,加入交互式计算的效果整体比不加入交互式计算得到的音乐要好,证明加入交互式计算可使算法合成音乐的质量更趋于稳定。

4 结束语

本文以音乐音频为操作对象在 AI 作曲以音频为载体的方面进行了尝试,借鉴语音信号处理手段,以 MFCC 作为特征向量,将音乐曲目看成具有时间序列特性的音乐片段序列,并以 LSTM-RNN 作为训练模型进行生成训练,该模型不仅能生成新的音乐序列,而且能平滑地将音乐片段拼接为一条完整的音频,在以音频为载体而进行 AI 作曲方面做了很好的尝试,但是模型作曲有长有短,结果也参差不齐,作曲质量依赖于音频素材的数量和质量,在加入交互式计算后得到一些生成质量上的提升,但是在如何得到普遍更高质量的音乐和算法的适应性方面还有待改进。

参考文献:

- [1] 刘润泉. 第三种作曲方式——论计算机音乐创作的新思维[J]. 中国音乐,2006(3):51-54.
 Liu Jian-quan. The third way of composing music on the new thinking of computer music creation[J]. Chinese Music,2006(3):51-54.
- [2] Turkalo D M. All music guide to electronica (book review)[J]. Library Journal, 2001,126(13):90.
- [3] Hiller L A, Isaacson L M. Experimental music/composition with an electronic computer[M]. New York: McGraw,1959.
- [4] Loubet E. The beginnings of electronic music in Japan, with a focus on the NHK studio: the 1970s [J]. Computer Music Journal,1998,22(1):49-55.
- [5] Sigtia S, Benetos E, Boulanger-Lewandowski N, et al. A hybrid recurrent neural network for music transcription[C]// 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, QLD, Australia, 2015: 2061-2065.
- [6] Eck D, Schmidhuber J. A first look at music composition using LSTM recurrent neural networks[M]. Lugano: IDSIA USI-SUPSI Instituto Dalle Molle, 2002.
- [7] Franklin J A. Recurrent neural networks for music computation [J]. Informs Journal on Computing, 2006,18(3):321-338.
- [8] Liu I, Ramakrishnan B. Bach in 2014: music composition with recurrent neural network[J]. Eprint Arxiv, 2014. <https://arxiv.org/pdf/1412.3191.pdf>.
- [9] Dhanalakshmi P, Palanivel S, Ramalingam V. Classification of audio signals using SVM and RBFNN [J]. Expert Systems with Applications, 2009, 36 (3):6069-6075.
- [10] Mathieu B, Essid S, Fillon T, et al. YAAFE, an easy to use and efficient audio feature extraction software[C]// International Society for Music Information Retrieval Conference, Ismir 2010, Utrecht, Netherlands,2010:441-446.
- [11] Salas H A G, Gelbukh A, Calvo H. Music composition based on linguistic approach[C]// Advances in Artificial Intelligence, Mexican International Conference on Artificial Intelligence, Pachuca, Mexico, 2010:117-128.