Music generation 总结

1、引言

随着人工智能和人工神经网络的快速发展，艺术与科学之间的界限正在逐渐消除。计算机音乐生成属于信息科学与艺术学的交叉学科，音频尤其是音乐生成的研究在多媒体研究领域是重要的一部分[1]。事实上，早在人工智能时代之前，就有许多尝试将音乐合成过程自动化。一个发展完善的音乐理论激发了许多对这项任务的启发方法，其中一些早在 19 世纪就开始了[2]。在 20 世纪中叶，一种 Markov-chain 的音乐作曲方法在[3]中得到发展。然而最近，在神经网络的帮助下，自动音乐产生了大量的进展[4] [5]。这些结果以及其他一些处理音乐或文字的作品展示了神经网络处理多维结构数据集的特殊能力。

针对计算机模拟音乐生成，国内外众多学者提出了许多切实可行的方法和理论。在传统的计算机音乐生成系统中，已经有了丰富的音乐表达和记录以及再现的方法。随着各类序列建模算法的出现[6]，促进了计算机音乐生成的深入发展。早期的音乐生成算法规定了一定的音名，音高范围，时值取值范围之后让计算机随机产生旋律。近年来，提出了若干浅层结构的音乐生成算法，例如隐形马尔科夫模型[7]，条件随机场[8]等。特别是，Frank 等人提出的 Tree-Based 音乐生成方法[9]，Walter 等人提出的基于 HMM 的音乐生成模型[10]，以及 Huang 等人提出基于深度信念网络[11]的音乐生成方法[12]。

随着近些年深度学习技术在计算机视觉[13]和语音识别[14]领域的发展，尤其是递归神经网络在自然语言处理，图像标注，自动问答，语音识别以及计算机艺术创作这些序列数据处理任务上的成功，对原始声音序列进行长时建模成为可能。基于字符级(Character-level) 的深度递归神经网络(Char-RNN) 在文本生成中取得了较好的效果[15]，这启发了本文将 RNN 应用于在计算机音乐生成。LSTM 在处理重要事件时的未知大小的时间滞后问题上显示出明显的更好的结果[16]。这种对间隙长度的类似的感知能力给 LSTM 提供了一种独特的优势，它可以通过隐藏的 Markov 模型、可选的递归神经网络和其他序列学习方法来学习，当算法与音乐一起工作时，音乐模式可以是暂时的复杂的，而 LSTM 似乎倾向于在很大程度上捕获这种复杂性[17]。

通过对音乐时间序列的分析，本文在已有理论成果基础上进行实验分析，设

计 LSTM 网络结构对生成音乐效果进行分析验证。

2、模型建立

2.1 LSTM 模型

该模型将基于单个 LSTM 单元，具有用于保持连续音符之间的时间依赖性的状态向量。在每个时间步骤中，我们输入一系列先前的音符，LSTM 最后一个单元的最终输出被送到一个全连接层以输出下一个音符的概率分布。通过这种方式，我们对概率分布进行建模。

LSTM 是利用门来控制信息流同时控制每一个 cell 单元阻止或者遗忘信息。当处理向量序列时，为了控制信息流，在每一个时刻应用一个输入门，遗忘门和输出门来决定信息的通过。假设段落由 T 个句子构成，$s_1,s_2,...,s_T$, s_t是第 t 个句子 s_t的特征表示，LSTM 的 cell 单元处理方式如下面的公式：

$$i_t = \sigma(W_i \cdot s_t + U_i \cdot h_{t-1} + b_i)$$
$$f_t = \sigma\left(W_f \cdot s_t + U_f \cdot h_{t-1} + b_f\right)$$
$$\widetilde{c_t} = tanh(W_c \cdot s_t + U_c \cdot h_{t-1} + b_c)$$
$$o_t = \sigma(W_o \cdot s_t + U_o \cdot h_{t-1} + b_o)$$
$$c_t = i_t^{\circ}\widetilde{c_t} + f_t^{\circ}\, c_{t-1}$$
$$h_t = o_t^{\circ} tanh(\, c_t)$$

此处s_t和h_t是第 t 时刻输入句子向量和输出句子向量，W_i，W_f，W_c，W_o，U_i，U_f，U_c和U_o是权重矩阵，b_i，b_f，b_c和b_o是偏置向量。符号°表示对应元素相乘，σ表示 sigmoid 函数。

在获得 LSTM 的最后一个隐藏层状态h_T后，我们将其输入到一个全连接层来获得概率分布，具体细节可以用下面的等式表达：

$$q = soft\max(W_y \cdot h_T + b_y)$$

此处W_y是权重向量，b_y是偏置向量，q是概率分布。

2.2 损失函数

我们使用了交叉熵函数作为损失函数，并使用 Adam 优化器来减少损失。

$$H(p,q) = -\sum_i p(i)\log q(i)$$

此处 $p(i)$ 表示第 i 个样本的标签，$q(i)$ 表示第 i 个样本的估计值。

3、实验设计

3.1 数据集

本实验的数据集是一组流行歌曲片段。如果您双击任何 MIDI 文件，您可以使用 GarageBand（Mac）或 MuseScore 等音乐播放应用程序打开它们。每首歌的格式应该是 (song_length, num_possible_notes)，其中 song_length> = min_song_length。歌曲中的音符的各个特征向量被处理成 one-hot 编码，这意味着它们是二进制向量，其中只有一个位置是 1。

3.2 参数设置

关于参数的设置，input_size 和 output_size 被定义为匹配每个 timestep 的编码输入和输出的尺寸。每首歌曲的编码表示形式 (song_length, num_possible_notes)，每个 timestep 播放的音符在所有可能的音符上编码为二进制向量。对于 LSTM，将 hidden unit 设置为 128 个，training_steps 为 200。此外设置训练期间 batch size 为 256，学习率为 0.001。为了训练模型，我们将从每首歌曲中选择长度时间段的片段，确保所有歌曲片段具有相同的长度并加速训练。实验环境如表 1 所示。

表 1 实验环境

Category	**Machine / Tools**
GPU	Tesla K80
Language	Python 2.7.14
Library	Google Tensorflow 1.5 CUDA v9.0.176 NVIDIA cuDNN v7

3.3 实验结果及分析

实验表明 LSTM 学习了训练数据中的音乐结构，并使用该结构在合成模式中约束其旋律输出生成音乐。

Step1: 安装 midi

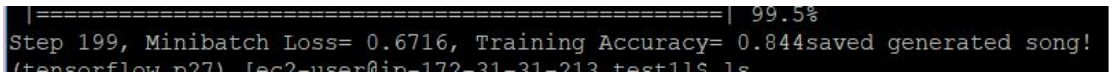
(tensorflow_p27) [ec2-user@ip-172-31-31-213 test1]$ cd python-midi-master

$python setup.py install

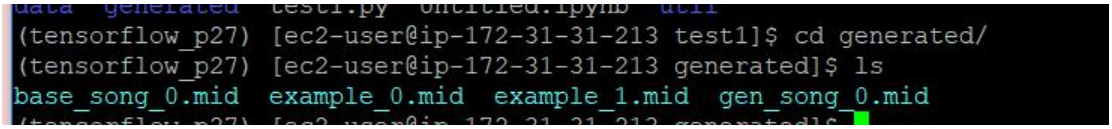
Step2:在 generated 文件夹中放入 1、2 个小结音乐

Step3: 运行

(tensorflow_p27) [ec2-user@ip-172-31-31-213 test1]$ python test1.py



Step4 在 generated 文件夹中生成 gen_song_0.mid



为了进一步提高生成音乐的效果，我们可以从以下几个方面进行改进：

- 限制模型输出：如果长时间训练模型，模型仍会不时输出真正的高/低音符。尽管这些音符出现的概率非常低，但它们不是 0，仍然会被模型偶然采样。为了解决这个问题，我们可以确定重新抽样概率，使它只来自前一个音符，或者继续抽样直到你得到一个与前一个音符接近的音符。
- 增加数据集：增加数据集的简单方法是将 batch_x 和 batch_y 中的值转置为随机向上或向下移动相同数量，从而产生不同的音调。这可以训练更强大的模型，从而更好地了解我们对音乐的感知如何依赖时间和空间不变。
- 支持和弦音乐：我们选择仅从每首歌曲中取出旋律（假定为每个 timestep 中播放的最高音符）。如果您不想一次播放一个音符，则可以更改歌曲编码为矩阵的方式，以及如何构建模型以进行训练和生成。

4、总结

本实验在训练网络时，在保证效率的情况下设置了固定的迭代次数和隐藏层神经单元数，针对 1 层 LSTM 网络结构进行实验，生成目标音乐序列文件。 另外， 目前音乐生成方法最终生成的音乐仅仅是拟合了样本语音， 输出的音乐类似于样本语音文件的子集，而本实验的最终目标是希望通过数据预处理和 LSTM 网络训练， 能够生成一种创新性且和原始样本音乐具有相似风格的的音乐文件。 因此，下一步将会改善音乐生成方法的初始化方案，通过不同文件的线性组合得出种子序列，而不是随机选取训练数据的子集作为种子序列，并且使用更深的网络结构，从而实现更好的音乐生成效果。

5、参考文献

[1] Wu Fei, Zhu Wen-wu, Yu Jun-qing. Multimedia technology research: 2014—deep learning and media computing [J]. Chinese Journal of Image and Graphics,2015,20(11) : 1423- 1433.

[2] Lovelace A A. 1842 Notes to the translation of the Sketch of the Analytical Engine[J]. Ada

User Journal, 2015.

[3] Hiller L A, Isaacson L M. Experimental Music; Composition with an Electronic Computer[M]. McGraw, 1959.

[4] Kluge E. INTERACTIVE GUIDANCE FOR MUSICAL IMPROVISATION AND AUTOMATIC ACCOMPANIMENT MUSIC:, WO/2016/195510[P]. 2016.

[5] Sigtia S, Benetos E, Dixon S. An end-to-end neural network for polyphonic piano music transcription[M]. IEEE Press, 2016.

[6] Ke Deng-feng, Yu Dong, Jia Jia. Guest editorial for special issue on deep learning for speech,text and image understanding[J]. Acta Automatica Sinica,2016,42(6): 805- 806.

[7] Baum L E, Petrie T, Soules G, et al. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains[J]. Annals of Mathematical Statistics, 1970, 41(1):164-171.

[8] Tseng H, Chang P, Andrew G, et al. A conditional random field word segmenter for sighan bakeoff 2005[C]. Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, 2005: 168- 171.

[9] Drewes F, Högberg J. An Algebra for Tree-Based Music Generation.[C]// International Conference on Algebraic Informatics. Springer-Verlag, 2007:172-188.

[10] Merwe A V D, Schulze W. Music Generation with Markov Models[J]. IEEE Multimedia, 2010, 18(3):78-85.

[11] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006,18(7):1527- 1554.

[12] Huang Q, Huang Z, Yuan Y, et al. A New Method Based on Deep Belief Networks for Learning Features from Symbolic Music[C]// International Conference on Semantics, Knowledge and Grids. IEEE, 2016:231-234.

[13] Yin Z, Chen Q, Zhang Y. Deep learning and its new progress in object and behavior recognition[J]. Journal of Image & Graphics, 2014.

[14] Zhou Y, Zhao H, Chen J, et al. Research on speech separation technology based on deep learning[J]. Cluster Computing, 2018:1-11.

[15] Graves A. Generating Sequences With Recurrent Neural Networks[J]. Computer Science, 2013.

[16] Colombo F, Muscinelli S P, Seeholzer A, et al. Algorithmic Composition of Melodies with Deep Recurrent Neural Networks[C]// Conference on Computer Simulation of Musical Creativity. 2016.

[17] Sutskever I, Vinyals O, Le Q V. Sequence to Sequence Learning with Neural Networks[J]. 2014, 4:3104-3112.