**目标：**满足最基本需求的dataset

参照Nottingham dataset： 时间规范，分类清晰，音色单一，区分melody和chord，有曲式结构和和弦的notation（abc格式）

在此基础上更进一步的信息量：more detailed features notation，明确的风格分类

目前待整理资源：MIDIset source (from reddit):

<https://www.reddit.com/r/WeAreTheMusicMakers/comments/3ajwe4/the_largest_midi_collection_on_the_internet/>

* + Genres include: Pop, Classical (Piano/Violin/Guitar), EDM, VideoGame, Movie/TV Theme (Rock etc)
  + **130,000 Midi File Collection 3.65Gb Uncompressed / 1.02 Gb Compressed ZIP**

|  |  |
| --- | --- |
| Stage1 | 【建库：正确性】  规范时间 -> 所有时间以鼠标录入为标准（quantification）  分离主旋律和非主旋律 -> 区分melody和chord -> supervised learning & rule-based (see“其他”)  歌曲结构的标注 -> 以及可能的方法    和弦正确性：rule-based 修改（需要提供可量化的rules） 旋律正确性：先天缺陷：无法判断， 要求高的需要人工打谱（museScore） |
| Stage2 | 【深化挖掘：更多特征】  （取决于人力， machine learning/stats 分析）  Necessary important feature需要列出来   * + music theoretical feature   + sound engineering feature（可以参照 opensmile / echonest）   有abc格式notation -> abc file conversion tool and source code |
| Stage3 | 【总结分类：风格学习】  明确的风格分类   * + rule-based（需要列出quantified规则）   + supervised learning |

**目前MIDIset的初步问题：**

* 时间：

|  |  |
| --- | --- |
| 问题 | 键盘输入导致时间节点不规范 |
| 目标 | 严格按照timeline，同鼠标输入 |
| 方法 | python midi tools 自动修复 |

* 乐器：

|  |  |
| --- | --- |
| 问题 | 负责主旋律的乐器不停变换，并且同时负责了部分和弦，无法直接得到主旋律 |
| 目标 | 简化feature？（未知目标） |
| 方法 | audacity支持individualizing每个乐器的部分   * + 1. 先用average hightest notes 乐器的代表主旋律     2. 如果主旋律同一时刻检测到有多个音 -> 需要进行chord and melody split -> 标记     3. 用代码得到乐器种类数量并且filter出1，2~n，n以上的歌曲 乐器种类n以上的歌曲 -> 大概率需要进行主旋律的修复 -> 标记 |

* + 命名：

|  |  |
| --- | --- |
| 问题 | 按title排序分类，命名混乱 |
| 目标 | 不同的文件夹基本代表爬虫的不同网站来源： 定义每个folder大概的风格并分类 e.g. 有纯钢琴，纯鼓点，流行曲（乐器很多很杂，有明显主旋律），儿歌（节奏型类似），爵士etc 再统一整理index和命名 |
| 方法 | 目前：手动抽样进行文件夹分类，重命名 |

* 细节：

|  |  |
| --- | --- |
| 问题 | 具体某些音错误 |
| 目标 | 样本量太大，只能尽量减少 |
| 方法 | 从最初的命名和文件夹可以大致判断文件是否同一来源，抽样检查，同一系列的midi明显错误多的话，直接删除整个系列（e.g. qfg2系列文件） 重要的歌曲需要手动录入 （by musescore） |

**stage1具体任务：**

coder

* 1. 熟悉python midi library, Music21，熟悉midi操作软件 e.g.Audacity
  2. 自动修复键盘录入造成的时间问题 -> 所有的midi notes转化成规范的timeframe <待解决问题：准确率和自动修复造成的误差>
  3. 整理并export出主旋律文件（先分成三大类：有清晰的最高声部并且单音，多个音需要split的，乐器太多需要手动split的）
  4. 利用现有的abc格式数据库，实现midi和abc的直接转化，尝试得到abc标记（歌曲结构，和弦，节奏型）

Musician

* 1. 手动抽样清理文件夹，按照大致种类重新分类，下标签，重命名（单一器乐，鼓点，流行曲，爵士etc）：e.g. W\下面按字母排序主要是流行歌，但是也有大面积的waltz，只看title就能发现，可以单独提出来
  2. 列出分离主旋律的规则/可能方法：保守策略：宁愿放过，不要做错（详见“其他”第二点）

引入bootstrapping的方法: 自己标注两千首：unsupervised分类

<待解决问题：有什么audio analysis的方法可以借鉴？>

* 1. 参考opensmile/echonest等audio analysis tool，以及abc格式的available features，列出重要（并且比较基础的）需要的features：e.g. 和弦，节奏型，调式，转调，渐强渐弱etc. （later filter out the existing supported features）
  2. 扩充/寻找 尽可能大的chord midi dataset （标准参照 \Chords and the Circle of Fifths\Chords），并且按照走向，调性，色彩，风格大致分类（以便以后systematic学习分类）

其他：

* 1. 如果是单一音色manual create midi file for specific dataset (比如民族传统音乐midi集)，建议用musescore再转换成midi更方便更准确

|  |  |
| --- | --- |
| How to separate the main meledy out of the midi file? --> unsupervised method of main melody detection  <方法细化：尽量减少人工> | * + - 1. The highest notes? (initial try       2. The simple chord has distance, and MIDI file already quantify the distance between the notes so we can find out the matching pattern and detect the chords ?? (need chord midi dataset and rules       3. The period repeating block in the bars (not rule-based       4. The first line of the MIDI soundtrack?? (unavailable) |

**Existing datasets：**

--> the Nottingham dataset : <https://github.com/jukedeck/nottingham-dataset>

the most standard midi file dataset, have mid and the **melody-only** and **chord-only** mid file/ small

--> The Lakh MIDI Dataset v0.1: <https://colinraffel.com/projects/lmd/>

big(180,000 songs) and 40,000 midi file match and align to the  Million Song Dataset (a big audio file). Advantage: big, 40,000 midi file have corresponding audio file, convenient for comparison and analysis

-->  SALAMI Data Set version 2.0  <https://github.com/DDMAL/salami-data-public>

Contains verse parsing; chord annotation

**Other non-dataset resources**

--> OVERALL SUM  <https://ldzhangyx.github.io/2019/02/25/music-toolkits/>

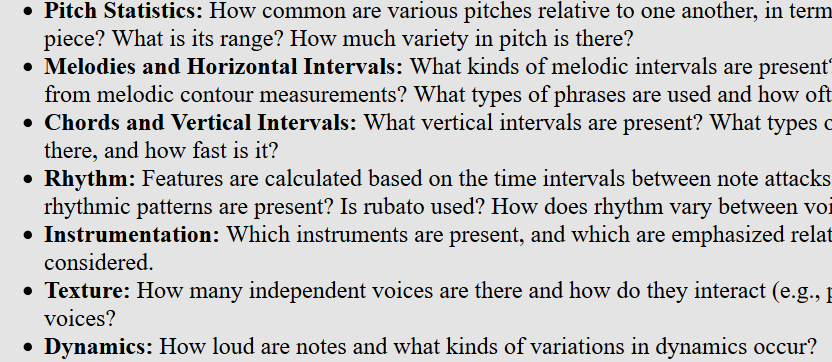
-> Python Code for potential data cleaning  <https://github.com/craffel/midi-dataset>

-->  some more information for datasets <https://pythonhosted.org/msaf/datasets.html>

--> Jsymbolic: <http://jmir.sourceforge.net/index_jSymbolic.html>

extract statistical information from musical data stored symbolically in file formats such as [MIDI](https://www.midi.org/)

--> .abc format



Analyse tool for Audio file:

-->  OPENSMILE

--> EchoNest

--> MuseScore: better and convenient input for music record

附：**数据集** Cited From <[*https://ldzhangyx.github.io/2019/02/25/music-toolkits/*](https://ldzhangyx.github.io/2019/02/25/music-toolkits/)>

**Nottingham Dataset**

* 地址：<https://github.com/jukedeck/nottingham-dataset>

特别有名的数据集，音乐界的MNIST。ABC格式可以被上面一些库直接解析，MIDI格式也可以转换。数据预处理的办法一般是shift到12个大调或者小调，速度调整至120bpm。

**Lakh Dataset**

* 地址：<https://colinraffel.com/projects/lmd/>
* GitHub：<https://github.com/craffel/midi-dataset>

Lakh MIDI数据集是176,581个独特MIDI文件的集合，其中45,129个已匹配并与Million Song Dataset中的条目对齐。

**Lakh Pianoroll**

* 地址：<https://salu133445.github.io/lakh-pianoroll-dataset/>

上面数据集的转化版本，但是表示使用了Pianoroll。

**Million Songs**

* 地址：<https://labrosa.ee.columbia.edu/millionsong/>

不是midi文件而是波形文件，特点在于数据量极大。

**SALAMI Dataset**

* 地址：<https://github.com/DDMAL/salami-data-public>

非常完整的标注数据集，值得一提的是可以从中提取和弦标注进行学习。

**MAESTRO Piano Dataset**

* 地址：<https://magenta.tensorflow.org/datasets/maestro>

Google Magenta一直在用的钢琴演奏数据集。有文字标注版和音乐原版，数据量极为庞大。

**IMSLP Music Library**

* 地址：<https://imslp.org/>

国际上最大的乐谱典藏数据库。其亮点在于录音和乐谱非常全，也许可以用于图像识别领域。有些乐谱似乎是收费的。

**the Classical piano MIDI database**

* 地址：<http://www.piano-midi.de/>

需要自己爬下来整理，钢琴MIDI数据集。

**The Largest MIDI Dataset**

* 地址：<https://www.reddit.com/r/WeAreTheMusicMakers/comments/3ajwe4/the_largest_midi_collection_on_the_internet/>

可以说是很神奇了，在Reddit上贡献了出来。是我看到的最大的MIDI数据集（3.65Gb），没有之一。

**MuseData**

* 地址：<http://musedata.stanford.edu/>

也是需要自己爬，但是这里的数据是根据流派做好划分的，非常一目了然。

**JSB Chorales**

* 地址：<https://github.com/czhuang/JSB-Chorales-dataset>

四声部众赞歌数据集。源地址挂掉了，这个GitHub提供了pickle包，很好用。

**FMA Dataset**

* 地址：<https://github.com/mdeff/fma>

这是一个音乐分析用的数据集，其亮点在于对流派、ID等数据的详尽标注。具体可以阅读readme。

**Midi Archive**

* 地址：<http://archive.cs.uu.nl/pub/MIDI/>

罗切斯特大学维护的一个数据集。

**The Session**

* 地址：<https://thesession.org/tunes>

非常非常完备的一个ABC格式的数据集，虽然需要自己爬，但是非常齐全。数据仍在不断增加中。

**MusicNet**

* 地址：<https://homes.cs.washington.edu/~thickstn/start.html>

这个数据集是raw的音频格式，大约10GB，提供了丰富的形式，标注详尽全面，甚至还做了PyTorch接口。

**the Symbolic Music dataset by Walder**

* 地址：<http://users.cecs.anu.edu.au/~u1018264/data.html>

清洗过的MIDI文件，作者目前还活跃在音乐研究一线。

**TheoryTab**

* 地址：<https://www.hooktheory.com/theorytab>

严格来说这不算是一个数据集，但是很多人在论文中用过它（比如MidiNet）。最出彩的地方在于其和弦标注。我觉得应该是网络上和弦标注最为全面的一个数据集了。这个网站现代化、可交互，非常值得看一看。

**J.S. Bach Series**

* 地址：<http://www.jsbach.net/midi/>
* 地址：<http://bachcentral.com/>

巴赫

**The Josquin Research Project**

* 地址：<http://josquin.stanford.edu/>

斯坦福的一个网站，好像也不能直接下载，但是数据整理得非常漂亮，提供了MIDI、XML等多种格式，收录了1420-1520年的复调音乐，并且支持在线搜索和试听。

**Drum Dataset**

* 地址：<https://www.reddit.com/r/WeAreTheMusicMakers/comments/3anwu8/the_drum_percussion_midi_archive_800k/>

虽然数据很小只有800k，但是这是我见过的第一个专用于鼓点的数据集。

**Video Game Datasets**

* 地址：<https://www.vgmusic.com/>

收录了游戏的背景音乐，而且真的非常非常齐全。

Cited From <[*https://ldzhangyx.github.io/2019/02/25/music-toolkits/*](https://ldzhangyx.github.io/2019/02/25/music-toolkits/)>