音楽データのDLへの適用

Team 16

音楽とDeep Learningの邂逅

楽曲に対して「もし◯◯ができたいいな」というニーズは昔から存在→Deep Learningがその手段として一助となり得るのではないか？

- 曲から音声(ボーカル)を外したい、反対にボーカルだけ抽出したい(ex.カラオケ練習)
- 特定の楽器を抽出したい
  - ex.ギターソロをコピーしたい等、楽器の練習
- ある楽曲を「○○風」にアレンジしたい
  - アンパンマンのマーチをB'zに…など
- 好きなアーティストAとBの曲を合体してみたい
- 曲の中から一番美味しい箇所(サビなど)を探り当てたい
  - ex.ストリーミング配信での試聴用音楽の生成
- 曲と曲との区切りを検出したい
  - →無音部分の検出はカセットテープの時代から機能として存在

2系統の音楽データとアプローチ

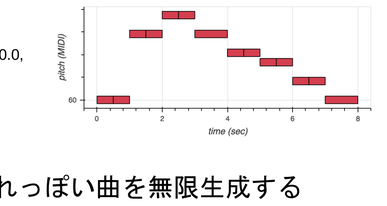
波形に着目するか？演奏情報に着目するか？

- (1)波形データ(WAVE)
  - GANSynth
    - GANを用いてAudioデータを合成する手法
    - 例えば、音楽CDで採用されているサンプリング周波数は44.1kHz。この場合は声波形を毎秒44,100回細切れにして、それぞれの時点の音声情報をデジタル情報にしたもの→データは手に入り易いが、学習・処理は重い
- (2)演奏情報(MIDI(※)ファイル)
  - MelodyRNN / MusicVAE
    - 「どの音がいつどのくらいの大きさ・長さで鳴るか」という情報の活用→楽曲情報が定量化されているため処理が容易である一方で、そもそも当該MIDIデータを何らかの形で作成する必要(カラオケ配信会社が高品質なデータを大量に持っていると思われる。)

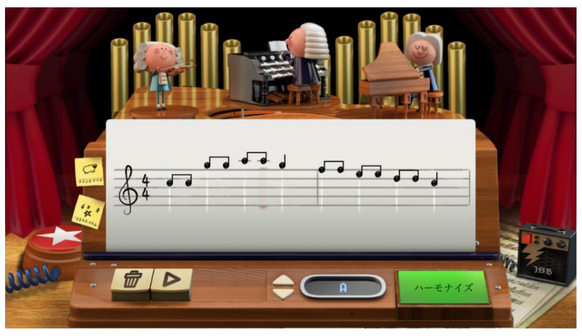
(※)「どの音がいつどのくらいの大きさ・長さで鳴るか」という情報を持っている。MIDI（ミディ、Musical Instrument Digital Interface）は、日本のMIDI規格協議会（JMSC、現在の社団法人音楽電子事業協会）と国際団体のMIDI Manufacturers Association (MMA) により策定された、電子楽器の演奏データを機器間でデジタル転送するための世界共通規格。物理的な送受信回路・インタフェース、通信プロトコル、ファイルフォーマットなど複数の規定からなる。

Ex.Google Dooldes(2019/3/21ヨハン・セバスティアン・バッハを称えて)

- 課題が追い込みを迎える中で、Googleの検索ロゴにバッハさんが登場。そして、「あなただけの「バッハ調」のメロディを、AIを使ったDoodleで作曲しませんか？」と我々の先を越される展開(?!)に
- https://www.google.com/doodles/celebrating-johann-sebastian-bach
  - 任意の2小節のメロディに対し、バッハの306曲を学習したハーモニー(単一の旋律要素のもとに、複数の声部が和声を構築する)を自動生成
  - 結果をMIDI形式で出力できる

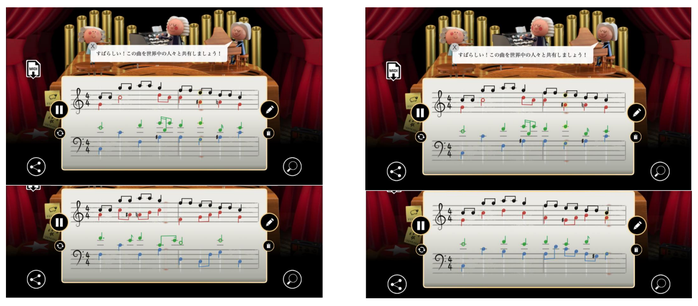


(おまけ)きらきら星にハーモニーをつける



(おまけ)処理中のバッハ君は煽り系キャラでしたが、示唆に富むコメントも言ってくれます

(おまけ)自画自賛の割には生成結果は当たり外れも大きかった



Melody RNN：モデル概要

- LSTMを用いた楽曲生成モデル
- 3つのオプション（Basic RNN, Lookback RNN, Attention RNN）からSpring Seminarでも扱ったAttentionを用いたRNNを選択
- AttentionはNeural Machine Translation by Jointly Learning to Align and Translate (D Bahdanau, K Cho, Y Bengio, 2014)の手法を採用している
- 元論文はエンコーダー・デコーダーモデルだが、本モデルでは、エンコーダー・デコーダーではなく、予測をする前のnステップ分の音に常に注目するようにしている

$$u_i^t = v^T \tanh(W_1' h_i + W_2' c_t)$$
$$a_i^t = \mathrm{softmax}(u_i^t)$$
$$h_t' = \sum_{i=t-n}^{t-1} a_i^t h_i$$

v, W'1, W'2: ハイパーパラメータ

hi: 予測をする前のnステップ分の結果

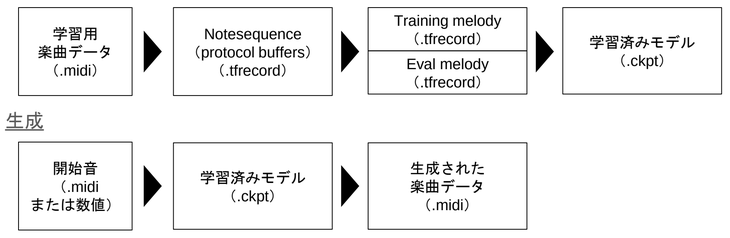
ct: 現在のRNNのセル状態

ht'はRNNの予測結果と次の入力の双方に適用される

Melody RNN：学習と生成

モデル学習と楽曲生成のフロー

学習



生成

Melody RNN：データセット

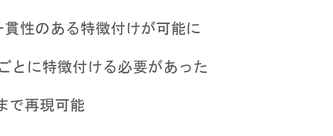
学習に用いたデータ

・米津玄師の楽曲9曲（Midi形式）（MuseSocreから取得）

・ビートルズの楽曲26曲（Midi形式）（Beatles MIDI filesから取得）

*MIDI/Notesequenceの楽曲はpitch（音程）、velocity（強弱）、start time and end time（音価）、tempos（テンポ）により表現される

```
# Add the notes to the sequence.
twinkle_twinkle.notes.add(pitch=60, start_time=0.0, end_time=0.5, velocity=80)
….
twinkle_twinkle.total_time = 8
twinkle_twinkle.tempos.add(qpm=60);
```



Melody RNN：学習と生成結果

学習時のパラメータ

- RNNのレイヤー数: [64, 64]
- Epoch数: 10,000回（米津玄師）／20,000回（ビートルズ）

生成時のパラメーター

- RNNのレイヤー数: [64, 64]
- ステップ数: 256
- 開始音: [60]（ド）

▶ 米津風、ビートルズ風のメロディの生成に成功！

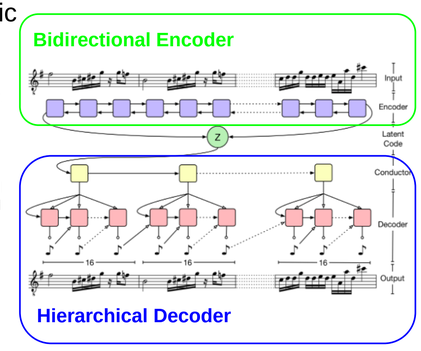
MusicVAE : A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music

概要
メロディシーケンス(16小節など)の潜在空間を学習したVAE
それっぽいメロディを一つの潜在変数から生成することができ、創作活動への応用が期待できる

このモデルでできること
Sample : 潜在空間内のrandomな点(Latent Vector)から新たに曲を生成できる
Interpolate : 複数のメロディをEncodeしてメロディ間の潜在変数を遷移させることでそれぞれのメロディ間をなめらかに補完するメロディを生成できる

入出力データ
Input/OutputともにMIDIファイル
Drumのみ、Melodyのみ、Trio(Melody, Bass, Dram)に対応
2小節、16小節に対応

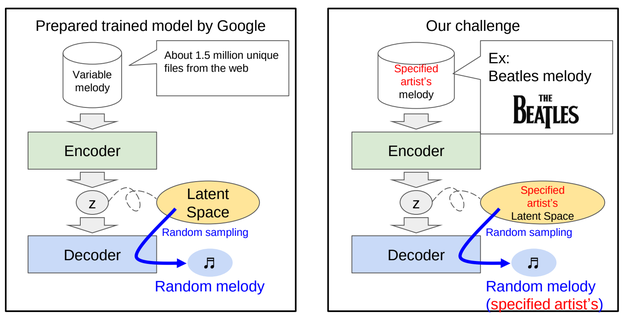
モデルの特徴
Bidirectional Encoder :
Hierarchical Decoder :
Trioでの曲生成



論文 : https://arxiv.org/pdf/1803.05428.pdf
Blog : https://magenta.tensorflow.org/music-vae

挑戦1：アーティストのそれっぽい曲を無限生成する

特定のアーティストについて独自データセットを用いて潜在空間を学習することで、ランダムサンプリングにてそのアーティストっぽいメロディの生成ができるはず！



挑戦1：アーティストのそれっぽい曲を無限生成する

結果と考察

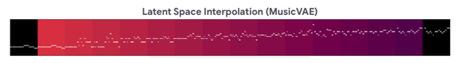
使用したデータセット(MIDIファイル)：米津玄師の楽曲(9曲)、Beatlesの楽曲(14曲)

- 学習
  うまく学習してくれていそう
- 生成 (デモにて実際の音を聞くことができます)
  - 米津玄師モデルで生成したメロディ(music_vae_random_sample_yonezu.mp3)
  - Beatlesモデルで生成したメロディ(music_vae_random_sample_beatles.mp3)

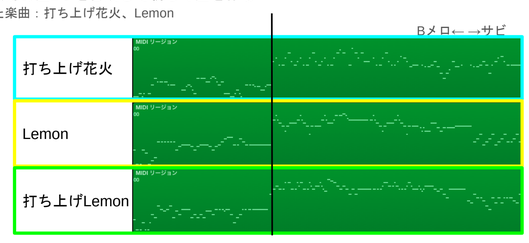
それぞれについて、それっぽいメロディを生成することができた！
一部、元データのMelodyそのままのような箇所(元の曲を切って貼った感じ)があるが、そのMelodyに入る、もしくは終わる部分について、自然な印象を受けるMelodyとなっている。

挑戦2：既存の曲から新しい曲の合成

提供されている学習済みの潜在空間上で同じアーティストの複数の曲をかけあわせることで、アーティストらしさやなにかしらの特徴をもった曲を生成できるはず



実験：米津玄師の曲2つを合成して新しく曲を作る
使用した楽曲：打ち上げ花火、Lemon



新しいメロディを生成することができた！Bメロからサビの盛り上がりも表現できていた。

GANSynthについて

・Neural Synthesizerの一つ (ICLR 2019 waiting review)

・ギターとピアノの中間音を生成する、といったような音声合成ができる

・WavenetとGANの組み合わせ

・高速なサンプリングにより高効率な学習、合成が可能に

Wavenetの50000倍ほどの速さ

・一つのglobalな潜在変数により一貫性のある特徴付けが可能に

Wavenetではtime stepごとに特徴付ける必要があった

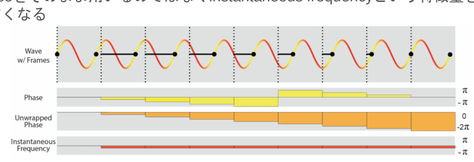
・音声波形の局所的な細かい特徴まで再現可能

GANだけでは細かい特徴付けは難しい

GANSynthで用いられている工夫点

・Wavenetのように直接波形を生成するのではなく、log-magnitudeとphaseを生成することにより滑らかな音声合成が可能に

・phaseをそのまま用いるのではなくinstantaneous frequencyという特徴量を用いることで、より学習がしやすくなる



・STFTのフレームサイズやメル周波数のスケールを大きくすることで、低音域で重なりやすい倍音を分離することでき、パフォーマンスが向上

学習データについて

- NSynthデータセットについて
  - 300,000曲の音符の注釈が付いた高品質の楽器の演奏音
  - 1つの楽器の音は、16kHzの周波数で4秒間に64000箇所サンプリングされたPCMのWaveフォーマットのオーディオファイル
- GANSynthでは
  - acoustic楽器かつ音高(24-84: 周波数32-1000)のデータを使用
  - Waveデータとラベルに音高(pitch)を使用
  - discriminatorの入力にWaveデータ、generatorの出力(z_noise)と、音高をラベルとして使用

音源の種類

Index	ID
0	acoustic
1	electronic
2	synthetic

楽器ごとのデータ数

Family	Acoustic	Electronic	Synthetic	Total
Bass	200	8,317	60,368	68,955
Brass	13,760	0	0	13,830
Flute	6,572	15	2,816	9,423
Guitar	13,343	16,815	5,275	35,423
Keyboard	8,508	42,465	3,838	54,991
Mallet	27,722	5,311	1,763	35,066
Organ	176	36,401	0	36,577
Reed	14,262	16	528	14,866
String	20,510	84	0	20,594
Synth Lead	0	0	5,501	5,501
Vocal	3,925	0	6,688	10,753
Total	108,978	110,214	86,777	305,979

再現実験

- 実装はgithubのmagentaのものを再利用
  - https://github.com/tensorflow/magenta/tree/master/magenta/models/gansynth
- 訓練データは次の２パターンを実施
  - 論文同様にNSynthのacoustic only,音高 24-84
  - 少し変えてNSynthのelectronic, synthetic,音高 24-84
- ハイパーパラメータ
  - ベストパフォーマンスの設定を流用
    - Mel-Spectrograms (メル周波数スペクトグラム)
    - Progressive Training (Progressive GAN)
      - 学習経過の進行とともにアップサンプリングの解像度を高くして行く手法
    - Hight Frequency Resolution

推論結果

- 入力はmidi音符またはランダム
  - 各音符の補完された潜在ベクトルを得る
  - 潜在ベクトルと音高をGeneratorに渡して合成す
- 学習済みの合成 (学習音源1,000,000のwavデータで学習)

自前で学習させたモデルを使った推論結果（4,000,000/11,000,000) 途中経過