1T-07

# Transformerを用いたファミコン風自動編曲手法の検討

小木曽　雄飛　　酒向　慎司
名古屋工業大学

## 1　はじめに

近年, 動画サイトや音楽投稿サイトではポピュラー音楽をファミコン風の音楽にした「ファミコン風音楽」が増えている．ファミコンとは, 1983 年に任天堂から発売された家庭用のゲーム機, ファミリーコンピュータの略記である．ここでは，ファミコンのゲーム内で使われていた BGM を「ファミコン音楽」と呼び，ポピュラー音楽をファミコン音楽風に編曲した音楽を「ファミコン風音楽」と呼ぶ.

ファミコン音楽には魅力があり, 様々な楽曲がファミコン風音楽に編曲されている．しかし，ファミコン風編曲は編曲の専門的知識や経験が必要のため難しい．本研究では，機械学習モデルを使用した，既存の楽曲を自動でファミコン風音楽に編曲する手法を検討する．そのため，既存の楽曲とこれらの楽曲をファミコン音楽風に編曲した楽曲の MIDI のペアデータセットを使用して，機械学習モデルにファミコン風編曲の変換規則を学習させる．MIDI データは多変量で自由度が高く, そのままの形では扱うのが難しいので，符号化させて扱う．この符号化の方法を変化させて，学習にどのような影響が出るか検証を行った.

## 2　ファミコン実機の制約

ファミコン音楽は電子音が使用されており，パルス波 2 音 (P1, P2)，三角波 1 音 (TR)，ノイズ 1 音 (NO) で構成されている．実機の制約があり，それぞれの音は同時に発音ができず，全体の同時発音数は 4 つに制限されている．ファミコン音楽は音楽の中では簡易的な構成となっているが, 4 音での表現の工夫がされており, 魅力のある音楽となっている.

本研究で扱うファミコン風音楽は，上記の構成音を使用し，実機の制約を満たすものとする．そのためには，既存の楽曲に対して，メロディやリズムといった原曲らしさを残しつつ，同時発音数が 4 音を超えないようにする必要がある．本研究は, 単一の音色で同時音が存在する既存の楽曲に対して，元の楽曲の構造を保ちつつ，音の削除，追加，音源の割り当てを行う．ただし，編曲する際は，時間的な構成は変えない.

## 3　Transformer を用いたファミコン風編曲

既存の楽曲とこれらの楽曲をファミコン音楽風に編曲した楽曲の MIDI のペアデータセットを使用して，機械学習モデルにファミコン風編曲の変換規則を学習させる．楽曲生成モデルで主流となっているエンコーダーデコーダーモデルである Music Transformer[1] を編曲生成モデルとして使用する．本研究で扱う楽曲は，同時音があるため，多声音楽を効率よく表現できる Performance Encoding[1] を使って MIDI データを表現する．その際, 同一の情報を保持している 2 種類の符号化方法により検証を行う．図 1 に 2 つの符号化方法の例を示す.

1 つ目は, 先行研究 [1] で使われていた Performance Encoding (符号化方法 1) を用いる．NOTE_ON, OFF (発音の始まりと終わり), TIME_SHIFT (時間の推移, 24Hz で離散化) の演奏情報を Token 化する．既存の楽曲とファミコン風楽曲は NOTE_ON, OFF の種類と数が異なる．ファミコン風楽曲は，先行研究 [2] を参考にして，各音色ごとに NOTE_ON，OFF を用意し，同時音がないため，NOTE_OFF の語彙数は 1 つとする．TIME_SHIFT はどちらも共通で，最大値を 4 秒とする．表 1 に符号化方法 1 の Token とその語彙数を示す.

符号化方法 1 のファミコン風楽曲の Token を見ると，同じ音高の音に対して，複数の符号が存在する．2 つ目は, 符号化方法 1 の冗長する部分を改良した Performance Encoding (符号化方法 2) を使用する．NOTE_OFF を用いる代わりに NOTE_ON で表現する時点で，音が持続される時間を指定する．そして, 1 つの Token で表現する代わりに 3 次元の Token で表現する．3 次元の Token は Token_Type (Token の種類), Pitch, Time (推移, 音の持続時間) を表す．表 2 に符号化方法 2 の Token とその語彙数を示す．表 1 と比較すると，使用する Token の種類が統一され，語彙数が削減されている.

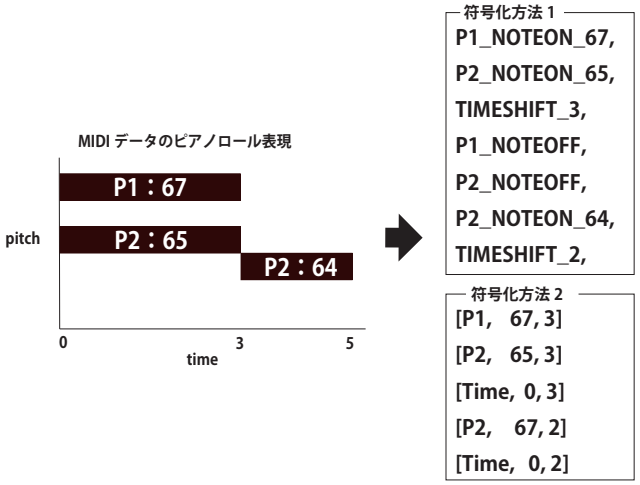


図 1: MIDI データの符号化の例

Transformer based automatic arrangement for NES music style
Yuto Ogiso, Shinji Sako
Nagoya Institute of Technology

| Token | 語彙数 |
|---|---|
| TIME_SHIFT | 96 |
| NOTE_ON | 88 |
| NOTE_OFF | 88 |
| 合計 | 272 |

(a) 既存の楽曲

| Token | 語彙数 |
|---|---|
| TIME_SHIFT | 96 |
| P1_NOTE_ON，OFF | 78 |
| P2_NOTE_ON，OFF | 78 |
| TR_NOTE_ON，OFF | 89 |
| NO_NOTE_ON，OFF | 17 |
| 合計 | 358 |

(b) ファミコン風楽曲

表 1: 符号化方法 1 の Token とその語彙数

| Token | 語彙数 |
|---|---|
| Token_Type | 2 |
| Pitch | 89 |
| Time | 96 |

(a) 既存の楽曲

| Token | 語彙数 |
|---|---|
| Token_Type | 5 |
| Pitch | 105 |
| Time | 96 |

(b) ファミコン風楽曲

表 2: 符号化方法 2 の Token とその語彙数

## 4 データセット

既存の楽曲とこれらの楽曲をファミコン音楽風に編曲した楽曲の MIDI のペアデータセットを作成する．ファミコン音楽風にされた楽曲は，ファミコン音楽の MIDI データセットである NES Music Database[3] から収集する．既存の楽曲はファミコン楽曲をピアノアレンジした MIDI データが公開されている Web サイトである NinSheetMusic からピアノ楽曲の MIDI データを収集する．ピアノ楽曲とこれらの楽曲をファミコン音楽風に編曲したペアデータセットを計 215 ペア作成した．本研究は時間的な構成を変えない編曲を前提とするため，時間の構造の対応関係をそろえた．

## 5 実験

作成したペアデータセットをもとに，入力がピアノ楽曲，出力がファミコン風音楽になるように Teacher Forcing で Music Transformer に学習させる．使用するデータセットは 4 秒ごとに分割した. 5 曲分だけテストデータとして使い，それ以外を学習データとして使用した．テストデータのピアノ楽曲を Music Transformer に入力して，生成されたものに対して評価実験を行った．

### 5.1 評価方法

生成された楽曲が元の楽曲の原曲らしさを残せているかどうかを確認する．文献 [4] に倣って，生成された楽曲と元の楽曲の MIDI データから抽出される特徴の統計的な類似度を計算する．生成された楽曲と元の楽曲の特徴量の分布を計算して比較する．特徴量は pitch class と IOI (Inter Onset Interval) の平均を使う．特徴量の分布比較は KL ダイバージェンス (KL) と Overlapped Area (OA) を使用する．

### 5.2 実験結果と考察

特徴の類似度の結果を表 3, 4 に示す．入力データ (原曲) と符号化方法 1 で生成されたデータ (生成 1)，入力データと符号化方法 2 で生成されたデータ (生成 2)，入力データと正解データ (正解) を比較した結果を示す．

| | 原曲 vs 生成 1 | 原曲 vs 生成 2 | 原曲 vs 正解 |
|---|---|---|---|
| KL | 0.4793 | 0.2986 | 0.2226 |
| OA | 0.7324 | 0.7771 | 0.8824 |

表 3: Pitch の類似度

| | 原曲 vs 生成 1 | 原曲 vs 生成 2 | 原曲 vs 正解 |
|---|---|---|---|
| KL | 0.0175 | 0.0114 | 0.0154 |
| OA | 0.7853 | 0.7597 | 0.8194 |

表 4: IOI の平均の類似度

原曲と生成 1 の類似度と，原曲と生成 2 の類似度を比較する．IOI の平均に関しては, KL はどちらも大きな差はなく, OA は原曲 vs 生成 1 のほうが値が大きくなっている．符号化方法 2 は時間の推移と音の持続時間を単一の Token で表現したため，時間推移，音の持続時間の学習精度が符号化方法 1 より低いと考えられる．Pitch に関しては符号化方法 2 で生成されたデータのほうが原曲との類似度が大きい．符号化方法 2 は Pitch を分けて符号化しているので，より原曲の Pitch を効率よく学習できたためだと考えられる．

しかし，実際に両方の符号化方法で生成された楽曲を聞くと，原曲らしさが残るような楽曲は生成されなかった．エンコーダの情報量だけでは，入力データの原曲らしさを十分に出力データに残すことは難しいためだと考えられる．

## 6 まとめ

本研究では既存の楽曲とこれらの楽曲をファミコン音楽風に編曲した楽曲のペアデータセットを用いたファミコン風編曲手法を検討した．その際, 2 種類の符号化方法を用いて，ファミコン風編曲の変換規則を機械学習モデルに学習させれるか検証を行った．しかし，どちらの方法も原曲らしさを残したファミコン風音楽は生成することができなかった．編曲全体を自動化させるのは難しいため，編曲全体ではなく, 音の削除といった編曲の一部分だけを自動化させた手法が有効だと考えられる．

## 参考文献

[1] Cheng Z. et al. :“An Improved Relative Self-Attention Mechanism for Transformer with Application to Music Generation,”2018.

[2] Chris D. et al. :“LakhNES: Improving multi-instrumental music generation with cross-domain pre-training,”2019.

[3] Chris D. et al. :“THE NES MUSIC DATABASE: A MULTI-INSTRUMENTAL DATASET WITH EXPRESSIVE PERFORMANCE ATTRIBUTES,”2018.

[4] Li-Chia Y. et al. :“On the evaluation of generative models in music,”2020.