

LSTMを用いたリカレントニューラルネットワークによる自動作曲

小山凌平 長名優子

東京工科大学 コンピュータサイエンス学部

1 はじめに

コンピュータを用いた自動作曲の歴史は古く、1957年のマルコフ過程を用いた研究以来、様々な研究が行われている。近年では、Deep Learning[1]を用いた手法なども数多く提案されている。そのような手法の1つとしてLSTM (Long Short-Term Memory)[2]を用いた自動作曲が提案されている[3]。この手法では、中間層にLSTMのブロックを用いたリカレントニューラルネットワークによって既存の楽曲の音形の特徴を学習し、それを利用して曲の生成を行っている。音形とは和音を構成する音どうしの高さの関係を表したもので、C, E, Gの和音であれば[1, 0, 0, 0, 1, 0, 0, 1]のように表現する。[1, 0, 0, 0, 1, 0, 0, 1]は和音を構成する一番下の音から半音ごとに音が存在するところを1として表現したものであり、1が存在する場所を表している。リカレントニューラルネットワークへは現在のステップまでの音形列を入力し、次のステップの音形が出力されるように学習が行われる。生成された音形の情報には音高の情報は含まれていない。この手法では、音高の情報は同時になっている音の中で最も低い音の音高を数値で表したもので表現している。数値は $A_0 \sim C_8$ の音に対し、0~87の値を割り当てている。音形と音高の組み合わせにより、実際の音の高さやリズムが表現されることになる。リカレントニューラルネットワークの出力によって得られた音形列に対し、学習データの中で最も類似した音形データに対応する音高のデータを割り当てることで音高を決定する。しかし、この手法では、音高が同じ音が連続する場合と1つの音が続いている場合とが区別できないという問題がある。例えば、 C_4 の4分音符が1つある場合と C_4 の8分音符が2つ連続する場合とは同じように扱われてしまうことになる。また、このシステムでは8分音符の長さを最小単位とし、それぞれに対して音形と音高の情報を割り当てることで曲を表現している。しかし、それぞれのデータが小節内のどの位置のものであるかが考慮されていないため、拍を考慮すること

ができていない。また、音高を決定する際に調などが考慮されていないといった問題もある。

本研究では、これらの問題点を考慮したLSTMを用いたリカレントニューラルネットワークによる自動作曲を提案する。

2 LSTMを用いたリカレントニューラルネットワークによる自動作曲

ここでは、提案するLSTMを用いたリカレントニューラルネットワークによる自動作曲について説明する。提案システムでは、既存の曲の特徴を学習させたリカレントニューラルネットワークにランダムな初期値を与えることで曲を生成をする。初期の音形と音高に相当するデータはランダムに生成し、それを入力として、次の時刻の音形・音高を出力する。出力された値に小節内での位置の情報を足したものを次の時刻の入力とし、入力と前の時刻の長期記憶とから次の音形と音高を出力する。図1に示すようにこれを繰り返すことにより曲が生成される。

2.1 学習データ

学習データとしてはクラシック曲のMIDIのデータベース[4]のデータを用いる。音の高さや長さの情報はMIDIデータからMusic21[5]というライブラリを用いて抽出する。Music21はMITで開発されたPythonのライブラリで、音楽解析に使用したり、出力された情報をMIDIデータやピアノロールの形に変換することができる。学習データとして用いる曲には色々な調のものが含まれているが、移調してすべてハ長調に揃えておく。こうすることで、同じ調で使われる音高のみが学習されることになり、不自然な高さの音が使用されにくくなるのが期待できる。抽出した情報か

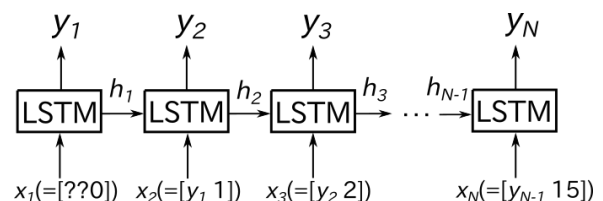


図1: 提案システムにおける曲の生成のイメージ



図 2: 16 分音符の長さ単位での情報の表現

ら 16 分音符の長さ単位で音形・音高・小節内の位置といった形でベクトル表現し、それをリカレントニューラルネットワークへの入出力として用いる。

2.2 曲のデータのベクトル表現

曲のデータはベクトルとして表現する。提案手法では、16 分音符の長さ単位 (以下、ブロックと呼ぶ) で考えるものとし、それぞれが音形、音高、小節内での位置の情報を持つ。つまり、図 2(a) のような楽譜は図 2(b) のようにブロック単位に分割した形に変換して考えることになる。

音形は、文献 [3] の手法と同様、和音を構成する音どうしの高さの関係を表したものである。ただし、音が存在する (1) かしないか (0) の 2 値ではなく、音が存在しない (0)、音の開始 (1)、前の音の継続 (2)、休符 (3) の 4 値で表すものとする。音形は、和音を構成する一番下の音から音が存在するところを 1 または 2 で表現する。図 2(b) の 5 つ目のブロックも 6 つ目のブロックも D₄, F₄, A₄ の 3 つ音から構成されているが、5 つ目のブロックは [1, 0, 0, 1, 0, 0, 0, 2]、6 つ目のブロックは [2, 0, 0, 2, 0, 0, 0, 2] と表されることになる。なお、ベクトルデータとしてブロックの情報を表す際には、0~3 の 4 値で表された音形そのものではなく、学習データに含まれるすべての音形に対して番号を割り当てた音形番号を用いる。

音高の情報は同時になっている音の中で最も高い音の音高を数値で表したもので表現する。数値は A₀~C₈ の音に対し、0~87 の値を割り当てたものを用いる。文献 [3] の手法では最も低い音の音高で表現しているが、提案システムでは最高音が主旋律を構成していることが多いと考え、最高音の音高の情報を用いるものとしている。

小節内での位置は 1 小節を 16 分音符の長さ単位で区切ったブロックを 0~15 の数値で表したもので表現する。



図 3: 生成された曲の一部

2.3 曲の生成

既存の曲の特徴を学習させたリカレントニューラルネットワークにランダムな初期値を与えることで曲を生成をする。初期の音形と音高に相当するデータはランダムに生成し、それを入力として、次の時刻の音形・音高を出力する。小節内の位置は 0~15 の値が繰り返されることが確定しているものであるため、出力させることはせず、入力としてのみ用いる。リカレントニューラルネットワークから出力された次の時刻の音形・音高に小節内の位置の情報を加えたものが次の時刻のリカレントニューラルネットワークへの入力となる。これを繰り返すことで、曲が生成される。生成されたデータを Music21 を用いて MIDI データや楽譜の形に変換したものを最終的なシステムの出力とする。

3 計算機実験

提案手法を用いて生成された曲の一部を図 3 に示す。

参考文献

- [1] 岡谷貴之：深層学習 (機械学習プロフェッショナルシリーズ), 講談社, 2015.
- [2] S. Hochreiter and Jrgen Schmidhuber : “Long short-term memory,” Neural Computation, Vol.9, No.8, pp.1735–1780, 1997.
- [3] 姫野雅大 : “LSTM による自動作曲システムの構築,” 東京大学大学院修士論文, 2016.
- [4] MIDI クラシック音楽データ集 by Windy, <https://windy-vis.com/art/> (2021 年 1 月 6 日参照)
- [5] Music21: a Toolkit for Computer-Aided Musicology - MIT <http://web.mit.edu/music21/> (2021 年 1 月 6 日参照)