Vol.2018-MUS-120 No.7
2018/8/22

注意機構を用いたエンコーダ・デコーダモデルに基づく歌声の音符推定

錦見 亮[1,a)]　中村 栄太[1,b)]　深山 覚[2,c)]　後藤 真孝[2,d)]　吉井 和佳[1,3,e)]

概要：本稿では，楽曲の主旋律を構成する音符系列を歌声から推定する手法について述べる．歌声は楽曲の主旋律を担うことが多く楽曲の印象に密接に関連しているため，歌声から音符推定することは楽曲解析において重要な技術である．従来は，音楽音響信号から歌声の連続的な音高軌跡（F0 軌跡）を予め推定し，F0 軌跡を時間・周波数方向に離散化することで音符の推定が行われる．しかし，歌声には歌唱表現（ビブラートやこぶし等）が含まれ，他の楽器に比べて音高のダイナミクスが非常に大きいため，音符推定には歌声変動を複雑かつ精密なモデル化する必要がある．また，事前に抽出される F0 軌跡を推定精度が，音符推定の精度にも影響する．そこで本研究では，DNN を用いて歌声から直接メロディ音符を推定する手法を提案する．具体的には，近年音声認識などで注目されている注意機構を用いたエンコーダ・デコーダモデルに基づき問題に取り組む．このモデルを使用するにあたり，音声認識（音声（信号）を文字や単語（記号）に変換）のアナロジーとして，歌声音符推定（歌声（信号）を音符（記号）に変換）を捉えた．人工的に合成した MIDI データと伴奏がないクリーンな歌声データを用いて，提案法の動作と性能を確認した．

1. はじめに

自動採譜は，音楽音響信号を楽譜に変換する技術であり，音楽情報処理分野における基礎技術である．音楽音響信号を離散的な記号で記述された楽譜に変換することで，音楽が人間や計算機が扱いが容易になる．特に，歌声採譜はポピュラー音楽中で主旋律を占めることが多い歌声から，主旋律の楽譜を推定する技術である．主旋律は楽曲において最も主要な構成要素であり，楽曲の印象に密接に関連しているため，主旋律を人間や計算機が扱いやすい形式に変換することは楽曲情報処理にとって重要である．歌声採譜が現実できれば，歌声と対応する楽譜の解析による歌声生成 [1]，得られた楽譜を対象にした音楽文法の解析，ハミング検索や能動的音楽鑑賞システム [2] など様々な場面で応用ができる．

従来，歌声解析に関する様々な研究が行われてきた．例えば，歌声分離 [3–5] では音楽音響信号から伴奏音と歌声とを分離する．また，基本周波数（F0）軌跡推定 [5–10] では，歌声から連続的な音高の軌跡を推定する．さらに歌声採譜を実現するためには，連続的な信号である歌声や F0 軌跡を時間・周波数方向に離散化して，半音単位の音高やビート単位の音価をもつ音符の系列を推定する必要がある．周波数方向の離散化については，F0 軌跡から半音単位の音高を持つ MIDI ノート（時間方向には離散化されていない）を推定する研究が多くなされている [11–13]．また，MIDI ノートに対して，各音符の音価を推定するリズム採譜の研究も存在する [14–16]．更に，入力に予め推定したビート時刻を用いて，F0 軌跡から音符の音高と音価を同時に推定する手法も存在する [17]．これらの手法を組み合わせて多段処理を行うことで，歌声採譜を一定の精度で達成することが可能である．しかし，こうした多段処理では前処理で発生した誤りが後段の処理に伝搬してしまう．

そこで本稿では，注意機構を用いたエンコーダ・デコーダモデルを用いて歌声を音符系列に直接変換することでこの問題の解決を図る．エンコーダ・デコーダモデルは長さの異なる 2 つの系列を直接変換するための DNN モデルである．エンコーダでは入力系列を中間表現に変換し，デコーダでは中間表現を出力系列に変換する．さらにデコーダ中で用いられる注意機構は，入力系列のどこに出力系列の各要素が対応するかを考慮することで，より長い系列の変換を可能にするモデルである．このモデルは，代表的な系列変換問題である機械翻訳 [18, 19] や音声認識 [20–22]

1 京都大学大学院情報学研究科
2 産業技術総合研究所 (AIST)
3 理化学研究所 革新知能統合研究センター (理研 AIP)
a) nishikimi@sap.ist.i.kyoto-u.ac.jp
b) enakamura@sap.ist.i.kyoto-u.ac.jp
c) s.fukayama@aist.go.jp
d) m.goto@aist.go.jp
e) yoshii@kuis.kyoto-u.ac.jp

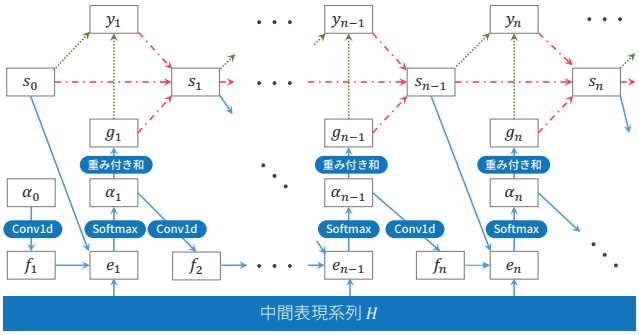

図 1 注意機構を用いたデコーダのモデル図．青の実線矢印が式 (1)，緑の点線矢印が式 (2)，赤の破線矢印が式 (3) を表す．

の分野で活発に研究されており，目覚ましい成果を上げている．本研究は，音声認識における音声（信号）から文字や単語（記号）系列への変換が歌声採譜における歌声（信号）から音符（記号）系列への変換と同型であることに着目し，上記のモデルを歌声採譜問題に適用したものである．

以降の本稿の構成は以下の通りである．2 章では，一般的な注意機構を用いたエンコーダ・デコーダモデル（以降，注意機構モデルと略す）について説明する．3 章では，歌声の音符推定問題に対して，そのモデルを適用する際に行った拡張について説明する．4 章では，提案法による評価実験の結果を示し，5 章で本稿についてのまとめを述べる．

2. 注意機構モデル

本章では注意機構モデル（特に，注意機構の部分）について説明する．注意機構モデルとして様々なものが提案されているが，本稿では特に [21] を参考にした．

2.1 エンコーダ

エンコーダは入力として受け取った特徴量系列 $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T] \in \mathbb{R}^{F \times T}$ を中間表現ベクトルの系列 $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_T] \in \mathbb{R}^{E \times T}$ に変換する．ここで，T, F, E はそれぞれ音響特徴量の長さ，音響特徴量の次元数，中間表現ベクトルの次元数を表す．通常エンコーダには，可変長系列データが扱える長短期記憶（long short-term memory, LSTM）やゲート付き回帰ユニット（gated recurrent unit, GRU）などの再帰型ニューラルネットワーク（recurrent neural network, RNN）が使用される．本研究では，時系列と同じ方向の再帰だけではなく逆方向の再帰も持つ双方向 LSTM（bidirectional LSTM）を使用した．中間表現ベクトルの次元数 E は予め設定するハイパーパラメータであり，エンコーダに RNN のみを使用する場合は，RNN の隠れ層の次元数が E となる．

2.2 注意機構付きデコーダ

デコーダでは中間表現ベクトルの系列 $\mathbf{H}$ から，出力記号系列 $\mathbf{Y} = [y_1, \ldots, y_N]$ を予測する．ここで，N は出力系列の長さ，$y_n \in \{1, \ldots, K\}$ は n 番目のタイムステップでの出力記号，K は出力として想定される記号の種類数（語彙数）を表す．後ほど詳しく説明するが，語彙数 K の中には出力記号系列の終端を表す特殊な記号 <EOS>（end of sequence の略）が含まれている．図 1 にデコーダのモデル概要を示す．エンコーダと同様にデコーダも RNN で構成される．n 番目のタイムステップにおける RNN 内部状態を $\mathbf{s}_n \in \mathbb{R}^D$ とすると，注意機構付きデコーダでは以下の 3 ステップを再帰的に計算する．

$$\boldsymbol{\alpha}_n = \mathrm{Attend}(\mathbf{s}_{n-1}, \boldsymbol{\alpha}_{n-1}, \mathbf{H}), \quad \mathbf{g_n} = \sum_{t=1}^{T} \alpha_{nt} \mathbf{h}_t \tag{1}$$

$$y_n = \mathrm{Generate}(\mathbf{s}_{n-1}, \mathbf{g}_n) \tag{2}$$

$$\mathbf{s}_n = \mathrm{Recurrency}(\mathbf{s}_{n-1}, \mathbf{g}_n, y_n) \tag{3}$$

ここで，Attend，Generate，Recurrency はそれぞれベクトルや行列に対して演算を行う関数である．以降，式 (1)，(2)，(3) のそれぞれついて詳細に説明する．

式 (1) は注意機構による演算を表す．$\boldsymbol{\alpha}_n \in \mathbb{R}^T$ は「注意重み」と呼ばれ，入力系列のどこに出力音符 y_n が対応するかを表す確率の集合である．注意重みによる中間表現 $\mathbf{H}$ の重み付き和 $\mathbf{g}_n \in \mathbb{R}^E$ が次のステップの計算 (2)，(3) に使用される．注意重み $\boldsymbol{\alpha}_n$ の各要素は以下のようにソフトマックス関数を用いて計算される．

$$\alpha_{nt} = \frac{\exp(e_{nt})}{\sum_{t'=1}^{T} \exp(e_{nt'})} \tag{4}$$

$$e_{nt} = \mathrm{Score}(\mathbf{s}_{n-1}, \mathbf{h}_t, \boldsymbol{\alpha}_{n-1}) \tag{5}$$

ここで，式 (5) 中の Score は行列演算を行う関数であり，これまで様々な計算方法が提案されていが，本稿では，次式で計算される畳み込み演算を用いた Score 関数 [21] を使用する．

$$\mathbf{f}_n = \mathbf{F} * \boldsymbol{\alpha}_{n-1} \tag{6}$$

$$e_{nt} = \mathbf{w}^\top \tanh(\mathbf{W}\mathbf{s}_{n-1} + \mathbf{V}\mathbf{h}_t + \mathbf{U}\mathbf{f}_{nt} + \mathbf{b}^{\mathrm{Att}}) \tag{7}$$

式 (6) において，$*$ は 1 次元の畳み込み演算，$\mathbf{F} \in \mathbb{R}^{C \times FW}$ は畳み込みに用いるフィルター，$\mathbf{f}_n \in \mathbb{R}^{T \times C}$ は 1 次元畳み込み演算の結果を表す．ここで，C はフィルターのチャンネル数，FW はフィルターサイズを表す．式 (7) において，$\mathbf{w} \in \mathbb{R}^A$ は重みベクトル，$\mathbf{W} \in \mathbb{R}^{A \times D}$, $\mathbf{V} \in \mathbb{R}^{A \times E}$, $\mathbf{U} \in \mathbb{R}^{A \times C}$ は重み行列，$\mathbf{b}^{\mathrm{Att}} \in \mathbb{R}^A$ はバイアスを表す．ここで，A は行列 $\mathbf{W}$，$\mathbf{V}$，$\mathbf{U}$ の行数および $\mathbf{b}^{\mathrm{Att}}$ の要素数で，事前に設定するハイパーパラメータである．

式 (2) は，前のタイムステップの内部状態 $\mathbf{s}_{n-1}$ と式 (1) で計算された重み付き和 $\mathbf{g}_n$ からデコーダの出力記号 y_n を計算する過程を表す．より具体的には，式 (2) では以下のように出力記号 y_n が決定される．

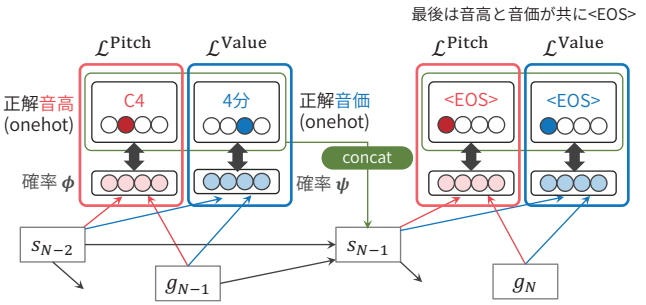

図 2　提案法の学習アルゴリズム．

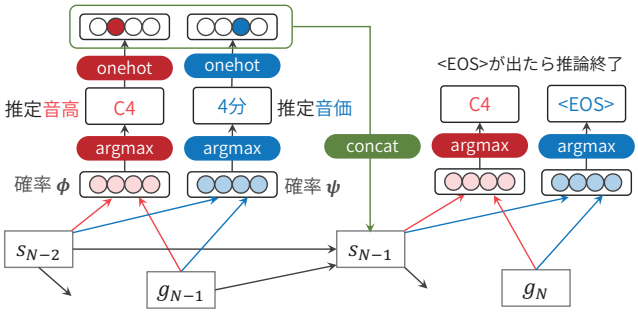

図 3　提案法の推論アルゴリズム．

$$\boldsymbol{\pi} = \text{Softmax}\left(\mathbf{P}\mathbf{s}_{n-1} + \mathbf{Q}\mathbf{g}_n + \mathbf{b}^{\text{Gen}}\right) \tag{8}$$

$$y_n = \underset{y_n \in \{1,\ldots,K\}}{\text{argmax}} \left(\pi_{y_n}\right) \tag{9}$$

ここで，$\mathbf{P} \in \mathbb{R}^{K \times D}$，$\mathbf{Q} \in \mathbb{R}^{K \times E}$ は重み行列，$\mathbf{b}^{\text{Gen}} \in \mathbb{R}^K$ はバイアスを表す．式 (8) 中の Softmax 関数の引数ベクトルを計算する方法は様々であるが，音高では最も単純な全結合層による計算方法を用いた．

式 (3) は，次のタイムステップの RNN 内部状態 $\mathbf{s}_n$ を計算する過程を表す．式 (3) を計算する際，y_n は onehot ベクトルに変換された後で Recurrency 関数に渡される．さらに，学習時と推論時で式 (3) の計算に用いる y_n は異なる．学習時は，正解データとして与えられた y_n を用いるのに対し，推論時は，式 (2) で求められた y_n を用いる．また，デコーダで行う再帰計算の回数も学習時と推論時で異なる．学習時は学習データに含まれる記号数と同じ回数だけ再帰計算を行う．推論時は y_n が<EOS>になるか，出力系列の長さが予め設定した最大長になるまで再帰計算を行う．

3. 提案法

本章では，注意機構モデルに基づいて，歌声の音符系列を推定する手法について説明する．

3.1 問題設定

我々が取り組む問題を以下のように定める．

入力:スペクトログラム $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T] \in \mathbb{R}^{T \times F}$
出力:音符系列 $\mathbf{Y} = [y_1, \ldots, y_N] = [(p_1, v_1), \ldots, (p_N, v_N)]$

ここで，T, F, N はそれぞれスペクトログラムの長さと周波数ビン数，音符の個数を表す．各 $\mathbf{x}_t \in \mathbb{R}^F$ は時刻 t におけるスペクトルを表す．各音符 y_n は半音単位の音高ラベル $p_n \in \{1, \ldots, K\}$ と音価ラベル $v_n \in \{1, \ldots, L\}$ で表現される．ここで，K と L はそれぞれ出力として想定されている音高や音価の種類数（語彙数）を表す．音高の語彙数 K の中には休符を表すラベルも含まれる．

3.2 注意機構モデルの拡張

歌声採譜ではモデルが出力すべき記号が音高と音価の 2 通り存在するため，2 章で説明したモデルをそのまま適用することはできない．この問題に対する最も単純な解決策は，音高と音価のペアを 1 つの記号とみなす方法である．この場合，モデルが出力すべき語彙数は<EOS>を含めて，$K \times L + 1$ となる．この語彙数は音声認識に比べれば少ないが，より語彙数を少なくした方が安定してモデルが学習できると期待される．そこで本研究では，音高と音価を独立した記号とみなし，提案法が両方の記号を同時に出力するよう拡張する．具体的には，式 (8)，(9) を音高用と音価用の 2 種類用意する．

$$\boldsymbol{\phi} = \text{Softmax}\left(\hat{\mathbf{P}}\mathbf{s}_{n-1} + \hat{\mathbf{Q}}\mathbf{g}_n + \hat{\mathbf{b}}\right), \tag{10}$$

$$p_n = \underset{p_n \in \{1,\ldots,K\}}{\text{argmax}} \left(\phi_{p_n}\right) \tag{11}$$

$$\boldsymbol{\psi} = \text{Softmax}\left(\bar{\mathbf{P}}\mathbf{s}_{n-1} + \bar{\mathbf{Q}}\mathbf{g}_n + \bar{\mathbf{b}}\right), \tag{12}$$

$$v_n = \underset{v_n \in \{1,\ldots,L\}}{\text{argmax}} \left(\psi_{v_n}\right) \tag{13}$$

ここで，$\hat{\mathbf{P}} \in \mathbb{R}^{K \times D}$，$\hat{\mathbf{Q}} \in \mathbb{R}^{K \times E}$，$\bar{\mathbf{P}} \in \mathbb{R}^{L \times D}$，$\bar{\mathbf{Q}} \in \mathbb{R}^{L \times E}$ は重み行列，$\hat{\mathbf{b}} \in \mathbb{R}^K$，$\bar{\mathbf{b}} \in \mathbb{R}^L$ はバイアスを表す．また，音高と音価を独立させるにあたり，<EOS>も音高用と音価用の 2 種類用意する．これにより，出力すべき語彙数は音高と音価の<EOS>を含めて $K + L + 2$ となる．

3.3 学習アルゴリズムと推論アルゴリズム

出力の語彙数を 3.2 章の方法を用いて拡張した場合における，デコーダの学習アルゴリズム 2 と推論アルゴリズム 3 について説明する．学習時は音高と音価それぞれを個別に onehot ベルトルに変換し，2 つの onehot ベルトルを concat したものをデコーダの入力とする．損失の計算は音高の交差エントロピー損失 $\mathcal{L}^{\text{Pitch}}$ と音価の交差エントロピー損失 $\mathcal{L}^{\text{Value}}$ を個別に計算し，それらの合計値 $\mathcal{L}^{\text{Pitch}} + \mathcal{L}^{\text{Value}}$ を各タイムステップにおける損失とする．推論時は式 (11) と (13) で求めた音高と音価をそれぞれ onehot ベルトルに変換し，それらを concat したものを次のデコーダの入力とする．デコーダが出力系列の長さが予め設定した最大長に達するか，出力音高と音価のいずれか一方にでも<EOS>が現れた場合に推論は完了する．

4. 評価実験

注意機構を用いたエンコーダ・デコーダモデルに基づく音符推定実験について報告する．本稿では，提案モデルの動作と性能を確認した．

4.1 実験条件

実験に用いた入出力のデータ形式，データセット，モデルの設定，及び評価尺度について説明する．

4.1.1 入力スペクトログラムの形式

すべての楽曲は 44100 [Hz] であり，窓幅 2048 点のハン窓を用いて，窓シフト幅 441 点（10 [msec]）の STFT を行った．その後，各スペクトログラムを最大値が 1 になるように正規化した．今回の実験では提案法に入力される楽曲はすべて同じテンポであると仮定した．音響的には同じ長さの音符でも楽譜上での音符の音価はテンポによって異なり，提案法では楽曲のテンポの違いを考慮した上で音符の音価を推定するのは難しいと考えたからである．そこで，得られたスペクトログラムに対してフェーズボコーダを用いて BPM150 に変換した．上記設定の STFT の場合，BPM150 の楽曲における 16 音符 1 つ分が 0.1 秒（10 フレーム分）に相当する．さらに，229 次元の Mel-scale 周波数に変換し，窓幅 5 フレーム，窓シフト幅 2 フレームのフレームスタッキング [23] を行い，提案法に入力した．

4.1.2 出力音符系列の形式

提案法が想定する音高の語彙数は $K = 42$，音価の語彙数は $L = 17$ とした．音高の語彙の内訳は，E2 から G5 まで 40 半音，および休符と<EOS>である．音価の語彙の内訳は，16 分音符の整数倍の音価を持つ音符のうち 16 分音符から全音符までの 16 通り，および<EOS>である．提案法が想定する語彙から逸脱する音符（休符）を含むものは使用しない．また，出力音符列は単旋律であると仮定した．

4.1.3 人工データ

提案法の動作を確認するために，人工的に合成したデータセット（音響データと楽譜データ）を用意した．各データにはランダムに決定された 1 ∼ 10 個までの音符が含まれる．データ中の各音符の音高と音価もランダムに決定される．音高は E2 から G5 まで（半音単位で 40 通り）のうちのいずれか，音価は 16 分音符から全音符まで（16 分音符単位で 16 通り）のうちのいずれかである．人工データには休符は含めなかった．また，同じ音高を持つ音符が連続するのを禁止したデータセット（D1）と，許容したデータセット（D2）の 2 種類用意した．窓シフト幅が 10 ミリ秒の STFT で 16 分音符が 10 フレーム分（0.1 秒）になるように，人工データの BPM を 150 とした．すべての楽譜データに対して，クラリネットの MIDI 音源を用いてサンプリング周波数が 44100 [Hz] の音響データを生成した．データ数は学習データが 5000，検証データが 100，テストデータが 100 である．

4.1.4 歌声データ

実際の歌声に対する提案法の動作を確認するために，RWC 研究法音楽データベース [24] のポピュラー楽曲の歌声データを用いたデータセット（D3）を作成した．各楽曲の MIDI アノテーションデータを一小節ごとに切り出して楽譜データを作成した．小節線を跨ぐ音符がある場合は，小節線ではなくその音符の終了位置でデータを切り出した．ただし，主旋律に 2 小節よりの長い音符を含む楽曲は使用しなかった．また，実験に使用した楽曲でも，切り出した区間に全音符よりも長い音符が含まれる場合，その区間は使用しなかった．休符の音価はできるだけ長いものとして扱った．ただし，休符が小節線を跨ぐ場合はその休符を小節線で 2 つに切り分けた．また，全休符のみのデータは除外した．各楽譜データについて，最初の音符（休符）のオンセット時刻と最後の音符（休符）のオフセット時刻を用いて歌声の音響データを切り出した．各音響データは STFT でスペクトログラムに変換後，フェーズボコーダを用いて BPM150 に変換したものが提案法への入力として用いられる．100 曲あるデータベース内のポピュラー楽曲のうち，アノテーションデータ [25] が正確であり，上記のデータ生成方法で 4.1.1 章や 4.1.2 章で述べた仮定を満たす 54 曲を選んで使用した．使用した 54 曲のうち，学習データが 52 曲，検証データが 1 曲，評価データが 1 曲だった．

4.1.5 モデル設定

提案モデルのエンコーダは 300×2 次元の隠れ層を持つ 3 層の双方向 LSTM で構成した．学習時には過学習を抑制するためにエンコーダに対して Dropout を行った．Dropout 率はいずれの層も 0.2 に設定した．デコーダは注意機構付きの単方向 LSTM であり，1 層 100 次元の隠れ層を持つ．注意機構内の 1 次元畳み込みにおける各パラメータ値は，チャンネル数 $C = 10$，フィルタサイズ $FW = 100$，パディングサイズは 50，ストライドは 1 とした．また，ハイパーパラメータ A の値は 200 とした．最適化アルゴリズムには Adam [26] を用いた．Adam の各パラメータ値は $\alpha = 0.001$（学習率），$\beta_1 = 0.9$，$\beta_2 = 0.999$，$\epsilon = 10^{-8}$ である．損失関数には交差エントロピー損失を用いるが，過学習を抑制するために，L2 ノルムよる荷重減衰（重みパラメータの正則化）を行った．正則化の強さを制御するハイパーパラメータの値は 10^{-5} とした．すべての全結合層の重みパラメータは $(-0.1, 0.1)$ の一様分布を用いてランダムに初期化した．エンコーダの双方向 LSTM とデコーダーの単方向 LSTM の重みパラメータ，および注意機構内の 1 次元 CNN のフィルターの初期化には He ら [27] が提案した手法を用いた．またすべてのバイアスは 0 で初期化した．学習初期にフレーム数の多いデータを用いると収束が遅くなるため，提案法の学習時には，入力データをフレー

ム数が少ない順にソートした．ミニバッチサイズを 50，エポック数を 50 として学習を行う．以上のネットワークの実装には PyTorch v0.4.0 [28] を用いた．

4.1.6 評価尺度

評価時のモデルパラメータには，学習時にエポックあたりの平均検証損失（validation loss）が最小だったものを用いた．評価尺度には，音声認識でしばしば用いられ次式で定義される単語誤り率（word error rate，WER）を使用した．

$$\mathrm{WER} = \frac{S + D + I}{N} \times 100\ [\%] \qquad (14)$$

ここで，式 (14) の分子は正解音符系列に対する推定音符系列の編集距離を表し，S，D，I はそれぞれ音符の置換，削除，挿入の数を表す．また，N は正解音符系列に含まれる音符数を表す．正解音符系列と推定音符系列に対して，音高と音価の両方を比較して計算される WER（音符一致 WER），音高のみを比較して計算される WER（音高一致 WER），音価のみを比較して計算される WER（音価一致 WER）の 3 通りを計算した．

4.2 実験結果

データセット D1，D2，D3 に対する実験結果を表 1 に，損失関数の経過を示す．D1 に対しては 50 エポックでモデルが十分学習できているのに対して，D2 と D3 に対してはモデルを十分に学習できなかった．D2 に対しては音価一致 WER が悪いのに対して，D3 に対しては音高一致 WER が悪かった．D2 は MIDI 合成音であり各音符の音高は常に一定であるため，同じ音高の音符が連続した場合に音符の境目が認識しづらくなり，音価一致 WER が悪くなったと考えられる．逆に，音符の音高が一定であるため，音高は認識しやすかったと考えられる．一方 D3 は実際の歌声から生成したデータセットであるため，各音符内の音高はビブラート等で激しく変動する．そのため，音高一致 WER が悪くなったと考えられる．逆に，その変動の大きさや声質の変化により音符境界が認識しやすくなり，音高一致 WER よりも音価一致 WER のほうが精度が良かった．

図 4 に提案法をデータセット D1，D2，D3 を用いて学習したときの訓練損失と検証損失の値の遷移を示す．D1 を用いた場合，定期的に損失の値が急上昇するものの，50 エポック内で学習損失と検証損失の両方が 0 付近に収束する様子が確認された．D2 を用いた場合，1 における結果は芳しくなかったが，図 4 から 50 エポックでは十分にモデルが学習できなかっただけだと分かった．図 4 から，過学習は起こしていないことが確認されるので，より多くの時間をかけて提案法を学習し，音符系列の推定精度を測定する必要がある．D3 を用いた場合，青線の学習損失は順調に下がっているが，橙線の検証損失は徐々に増加しており，モデルが過学習を起こしていることが確認された．しかし，実際の歌声に対してでも過学習できるだけの能力を現在のモデルが有しており，今後モデルのパラメータの値を調節や学習データ数を増やすことにより性能の改善が期待される．

表 1 実験結果

人工データセット	D1	D2	D3
音符一致 WER	0.56 %	66.79 %	80.73 %
音高一致 WER	0.37 %	13.99 %	69.27 %
音価一致 WER	0.19 %	62.34 %	37.27 %

5. おわりに

本稿では，注意機構を用いたエンコーダ・デコーダモデルに基づき，歌声から音符系列を推定する手法について議論した．提案法では，注意機構モデルを歌声採譜に合わせて拡張する方法について述べるとともに，簡単なデータを用いて提案法が入力歌声に対して動作することを確かめた．

今後の方針として最も興味深いのは，楽曲のテンポやビートを推定する DNN と提案法とを統合することである．提案法の入力はテンポが一定であることを想定しているが，この統合により様々なテンポの曲だけではなく，曲内でテンポが変動する曲も扱えるようになると考えられる．更に，発展的な内容として本手法を伴奏付き音楽音響信号に対応させることも重要な課題である．これが実現されれば，前処理としての歌声・伴奏分離を必要とせず，市販楽曲をそのまま入力とするだけで，歌声採譜が可能になる．

謝辞 本研究の一部は，JST ACCEL No. JPMJAC1602，JSPS 科研費 No. 26700020，No. 16H01744 および No. 16J05486 の支援を受けた．

参考文献

[1] Blaauw, M. and Bonada, J.: A Neural Parametric Singing Synthesizer Modeling Timbre and Expression from Natural Songs, *Applied Sciences*, Vol. 7, No. 12 (2017).

[2] Goto, M., Yoshii, K., Fujihara, H., Mauch, M. and Nakano, T.: Songle: A Web Service for Active Music Listening Improved by User Contributions., *Proc. of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011)*, pp. 311–316 (2011).

[3] Li, Y. and Wang, D.: Separation of singing voice from music accompaniment for monaural recordings, *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 15, No. 4, pp. 1475–1487 (2007).

[4] Huang, P.-S., Chen, S. D., Smaragdis, P. and Hasegawa-Johnson, M.: Singing-voice Separation from Monaural Recordings Using Robust Principal Component Analysis, *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2012)*, pp. 57–60 (2012).

[5] Ikemiya, Y., Yoshii, K. and Itoyama, K.: Singing voice analysis and editing based on mutually dependent F0 estimation and source separation, *2015 IEEE International Conference on Acoustics, Speech and Signal Pro-*

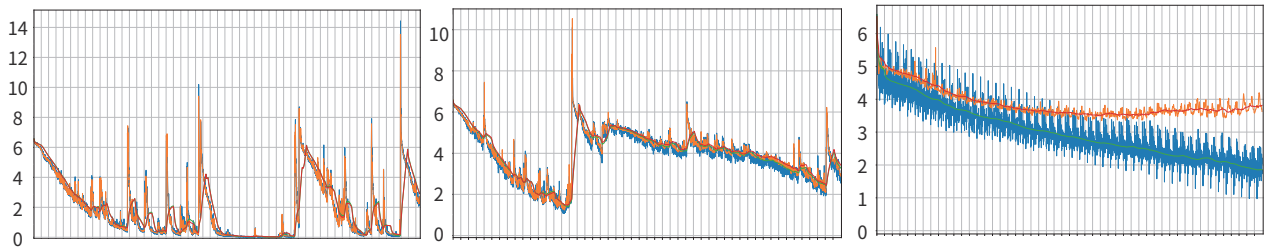

図 4 学習時における学習損失と検証損失．青線が学習損失，緑線が 1 エポック分の平均学習損失，橙線が検証損失，赤線が 1 エポック分の平均検証損失を表す．縦軸が損失の値，横軸はエポック数を表す．左から順にデータセット D1，D2，D3 を用いて提案法を学習したときの損失の値の遷移である．

cessing (ICASSP 2015), pp. 574–578 (2015).

[6] Hermes, D. J.: Measurement of Pitch by Subharmonic Summation, *The journal of the acoustical society of America*, Vol. 83, No. 1, pp. 257–264 (1988).

[7] Goto, M.: PreFEst: A Predominant-F0 Estimation Method for Polyphonic Musical Audio Signals, *Proceedings of the 2nd Music Information Retrieval Evaluation eXchange* (2005).

[8] Salamon, J. and Gómez, E.: Melody extraction from polyphonic music signals using pitch contour characteristics, *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 20, No. 6, pp. 1759–1770 (2012).

[9] Durrieu, J.-L., Richard, G., David, B. and Févotte, C.: Source/filter model for unsupervised main melody extraction from polyphonic audio signals, *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 18, No. 3, pp. 564–575 (2010).

[10] Mauch, M. and Dixon, S.: pYIN: A fundamental frequency estimator using probabilistic threshold distributions, *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014)*, pp. 659–663 (2014).

[11] Molina, E., Tardón, L. J., Barbancho, A. M. and Barbancho, I.: SiPTH: Singing Transcription Based on Hysteresis Defined on the Pitch-Time Curve, *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, Vol. 23, No. 2, pp. 252–263 (2015).

[12] Yang, L., Maezawa, A., Smith, J. B. L. and Chew, E.: Probabilistic Transcription of Sung Melody Using a Pitch Dynamic Model, *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, pp. 301–305 (2017).

[13] Kroher, N. and Gómez, E.: Automatic transcription of flamenco singing from polyphonic music recordings, *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, Vol. 24, No. 5, pp. 901–913 (2016).

[14] Takeda, H., Saito, N., Otsuki, T., Nakai, M., Shimodaira, H. and Sagayama, S.: Hidden Markov Model for Automatic Transcription of MIDI Signals, *IEEE Workshop on Multimedia Signal Processing (MMSP 2002)*, pp. 428–431 (2002).

[15] Raphael, C.: A Hybrid Graphical Model for Rhythmic Parsing, *Artificial Intelligence*, Vol. 137, No. 1-2, pp. 217–238 (2002).

[16] Hamanaka, M., Goto, M., Asoh, H. and Otsu, N.: A Learning-Based Quantization: Unsupervised Estimation of the Model Parameters, *Proc. of the 5th International Conference on Multimodal Interfaces (ICMI 2003)*, pp. 369–372 (2003).

[17] Nishikimi, R., Nakamura, E., Goto, M., Itoyama, K. and Yoshii, K.: Scale- and Rhythm-Aware Musical Note Estimation for Vocal F0 Trajectories Based on a Semi-Tatum-Synchronous Hierarchical Hidden Semi-Markov Model, *Proc. of the 18th International Society for Music Information Retrieval Conference, (ISMIR 2017)*, pp. 376–382 (2017).

[18] Luong, M.-T., Pham, H. and Manning, C. D.: Effective Approaches to Attention-based Neural Machine Translation, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pp. 1412–1421 (2015).

[19] *Neural Machine Translation by Jointly Learning to Align and Translate* (2015).

[20] Chan, W., Jaitly, N., Le, Q. and Vinyals, O.: Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition, *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, pp. 4960–4964 (2016).

[21] Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K. and Bengio, Y.: Attention-Based Models for Speech Recognition, *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pp. 577–585 (2015).

[22] Prabhavalkar, R., Sainath, T. N., Li, B., Rao, K. and Jaitly, N.: An Analysis of "Attention" in Sequence-to-Sequence Models, *Proc. of Interspeech* (2017).

[23] Sak, H., Senior, A., Rao, K. and Beaufays, F.: Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition, *INTERSPEECH 2017*, pp. 1468–1472 (2015).

[24] Goto, M., Hashiguchi, H., Nishimura, T. and Oka, R.: RWC Music Database: Popular, Classical and Jazz Music Databases, *The 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pp. 287–288 (2002).

[25] Goto, M.: AIST Annotation for the RWC Music Database., *The 7th International Conference on Music Information Retrieval (ISMIR 2006)*, pp. 359–360 (2006).

[26] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, *arXiv preprint arXiv:1412.6980*, pp. 1–15 (2014).

[27] He, K., Zhang, X., Ren, S. and Sun, J.: Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, *2015 IEEE International Conference on Computer Vision (ICCV 2015)*, pp. 1026–1034 (2015).

[28] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A.: Automatic differentiation in pytorch (2017).