Vol.2022-MUS-134 No.44
Vol.2022-SLP-142 No.44
2022/6/18

コード進行を考慮して多様なメロディを生成するTransformerとVAEの複合モデル

寺島 凌[1,2,a)]　阿部 由吾[1,2,b)]

概要：深層学習技術の発展により，機械によって自動生成される音楽の品質は日々向上している．特にシンボリック音楽生成の領域では，Transformer を活用することで長期の依存関係を考慮したメロディの生成が可能になっている．一方で，これらの音楽生成モデルを現実の用途や創作に活用するためには，生成される音楽の操作可能性と多様性が両立される必要がある．本研究では，指定されたコード進行に沿ったメロディを生成できる深層学習モデルを提案する．具体的には，Transformer を利用したメロディの自己回帰タスクにおいて，コード進行の情報を条件づけるような定式化を行う．加えて，生成結果の多様性を向上させるために VAE を導入し，潜在空間での操作を行う．実験では，指定されたコード進行と生成されたメロディの親和度を評価する．

A combined model of Transformer and VAE for melody generation conditioned on chord progressions

1. はじめに

MIDI などのシンボリックな音楽表現を使った自動音楽生成の分野では，深層学習の発展によって人がルールを設計しなくても比較的自然な曲の生成が可能になりつつある．音楽は時間的な構造を持つデータとして扱われるので，以前は再帰的ニューラルネットワーク (RNN) に基づくアーキテクチャが主流だった [1] が，近年では Transformer [2] を基にした手法が増えている [3], [4], [5]．RNN に比べて Transformer の方が時間的に離れている要素同士の関係をうまく扱うことが可能であり，音楽の中に様々なスケールで存在する繰り返しの構造を捉えられることが Transformer の利点として挙げられる．

深層学習を用いた自動音楽生成における一つの課題は「生成される音楽をどのように操作するか」という点である．例えば，「眠りにつきやすい音楽」「元気が出る曲」などの特定の機能を持つ音楽を生成するためには，モデルの生成プロセスがユーザ指定の曲調などによって条件づけられる必要がある．作曲家の楽曲制作を支援するためのツールとして自動音楽生成が使われる場合でも，作曲家の用途に合わせて生成結果のジャンル・リズム・複雑度・コード進行などを自在に制御できることが望ましい．前者の例においては曲全体のマクロな特徴が操作の対象だが，後者では小節単位のミクロな操作を施したい場合もある．

作曲家の制作支援への応用を念頭に置くと，コード進行に基づくメロディの生成を可能にすることは大きな意義がある．複数の音が同時に鳴ったものをコード，複数のコードが時間方向に連なったものをコード進行と呼ぶが，多くの音楽ジャンル (e.g. ジャズ，ポップ) においては作品の構造や与える印象がコード進行に強く依存する．例えば，西洋音楽におけるカデンツ (cadence) の概念は，コード進行によって「曲やフレーズの終わり方」の印象が様々に異なることを表している．したがって，作曲においてコード進行を活用することの重要性は，とりわけコード間の前後関係を考慮することにある．2.1 節で紹介するように，コードに合わせてメロディを生成できる深層学習モデルは既に存在するが，コードの前後関係まで考慮している先行研究は極めて少ない．

本研究では，ユーザが指定したコード進行 (=コードの前後関係) に基づくメロディ生成のタスクに Transformer を

[1] equal contribution
[2] 東京大学大学院 情報理工学系研究科
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
[a)] terajima@isi.imi.i.u-tokyo.ac.jp
[b)] y-abe@isi.imi.i.u-tokyo.ac.jp

適用する．さらに，このモデルに variational autoencoder (VAE) [6] を導入し，潜在空間に摂動を加えることで生成結果を変化させ，提案モデルが作り出せるメロディの多様性を向上させることを目指す．

本論文の貢献は以下の通りである．

- コードの前後関係に基づくメロディ生成を行うための深層学習モデルを提案
- コードとメロディの対応を定量的に評価する指標としてコード親和度を提案
- サンプリング手法の工夫や潜在空間の操作によって多様なメロディの生成を実現

2. 先行研究

2.1 ユーザによる操作が可能な自動音楽生成モデル

自動音楽生成において，ユーザによる何らかの操作を可能にする深層学習モデルは既に多く提案されている．近年の研究において最も代表的だと言えるのは，VAE に基づくアプローチである．Roberts らの MusicVAE [7] では，双方向 RNN からなる encoder と decoder を用いて潜在空間を構成することで，2 つの音楽シーケンスを混ぜ合わせたような新たなシーケンスの生成を可能にした．Kawai らは，VAE の潜在空間から操作対象となる音楽要素の情報を排除するような敵対的学習を行うことで，その音楽要素が decoder への条件づけによって操作しやすくなるような工夫を行った [8]．これにより，メロディを構成する音の数やピッチの分散など，生成されるメロディの様々な大域的性質を効果的に操作することができた．

Transformer に基づくモデルでは，シーケンスの一部あるいは特殊なトークン・タグ情報をモデルへの入力に追加することで生成結果を操作するのが一般的である．例えば Music Transformer [3] では，指定したメロディに対してその続きや，それに合う伴奏をモデルに出力させるというタスクが行われている．OpenAI の大規模モデル MuseNet [4] では，全ての入力シーケンスの始めに作者と使用楽器の情報を表すトークンが付加されているので，生成時はこれらのトークンを変更することで生成結果のスタイルを操作することができる．一方で，Transformer と VAE を組み合わせることでユーザによる操作が可能な自動音楽生成を達成した先行研究も存在する [5], [9]．これらのモデルについては 2.2 節で詳しく扱う．

また，コードに合わせたメロディ生成を扱っている先行研究も存在する．音楽シーケンスを 2 次元データとして扱い，畳み込みニューラルネットワーク (CNN) を用いて自動音楽生成を実現した MidiNet [10] では，一小節ごとに指定されるコードに合わせてメロディ生成が行われた．RNN ベースの確率的生成モデルである DeepBach [11] では，その時々のコード情報をメロディ生成におけるソフトな制約として用いている．しかし，これらのモデルではコード同士の前後関係は扱われていない．

コード同士の前後関係も含めてコード進行を扱っている例としては，いずれも双方向 RNN を用いた手法である [12] と [13] が挙げられる．前者では，曲全体のコード進行がメロディを生成するときの制約として与えられており，この制約がメロディ生成の精度を向上させることが報告されている．後者では，VAE によってコード進行と音楽スタイルに関する disengangled な潜在空間が獲得されており，これが音楽シーケンスのスタイル変換に活用されている．対して，本研究は Transformer を活用することでコード進行に基づく音楽生成のスケーラビリティ向上を試みている．

2.2 Transformer と VAE の複合モデル

データの抽象表現獲得・操作可能性に長けた VAE と，時系列データにおける長期的な依存関係を扱うのに長けた Transformer を組み合わせた手法は，音楽生成に限らず，様々な分野で活用されている．たとえば，Yan ら [14] は，VAE を利用して，動画データの各時刻のフレームの抽象表現を獲得し，その時系列データを用いた自己回帰問題で Transformer を訓練することによって，動画生成を行う手法を提案した．この手法では，画像の抽象表現獲得のために VAE を，時系列方向の依存関係を扱うために Transformer を，別々に利用している．また，Liu ら [15] は，VAE の各層の構造に，Transformer に用いられる multi-head attention 機構を導入したモデルを提案し，文章生成タスクに応用した．このモデルは，単語系列全体を一つの VAE に入力して，文全体の情報を潜在空間にエンコードした後，そこからサンプリングされた潜在表現をデコードすることによって，文の再構成を行うことができる．そして，潜在表現に操作を加えることで，生成文に変化を加えられることを示した．このモデルでは，系列データとしての文全体を一度に扱うために Transformer を利用し，そこから得られた文全体の特徴量を一度に潜在空間に落とし込むことで生成文の操作を可能にするために VAE を利用している．そして，Wang ら [16] は，欠損文補完タスクに対して，VAE と encoder-decoder 構造の Transformer モデルを組み合わせたモデルを提案した．Transformer encoder に欠損ありの文章系列を入力し，得られた抽象表現を VAE に通す．そこからサンプリングされた潜在表現と，Transformer decoder における予測中の文の特徴量を統合し，後続する単語系列を予測する．このモデルも，単語系列データを扱うために Transformer を利用し，生成文を潜在空間での操作可能にするために VAE を利用している．

Transformer と VAE を組み合わせた手法は，音楽生成の分野でも取り入れられている．Jiang ら [9] は，音楽生成タスクに対して，VAE と encoder-decoder 構造の Transformer モデルを組み合わせたモデルを提案した．このモデルでは Transformer encoder と Transformer decoder の間

Vol.2022-MUS-134 No.44
Vol.2022-SLP-142 No.44
2022/6/18

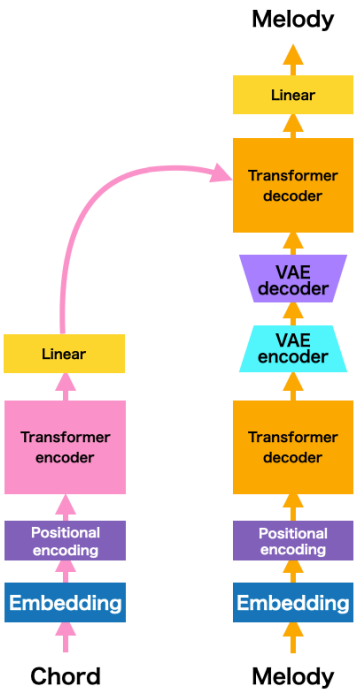


図 1 提案モデルの構造
Fig. 1 Architecture of proposed model.

表 1 トークンの種類一覧
Table 1 Types of tokens used in input representation

タイプ	値	説明
Note	40–109 (全 70 種類)	音符を表す． 値は音の高さを表す．
Rest	—	休符を表す． 使い方は Note タイプと同様．
Chord	CM, Dm など (全 25 種類)	コードの種類を表す．
Duration	1–132096 (全 44 種類)	音の長さを MIDI 上の tick 数で表す．
Padding	—	Attention 計算から除外される．
BOS	—	Beginning-of-Sentence． メロディ系列の先頭につく．

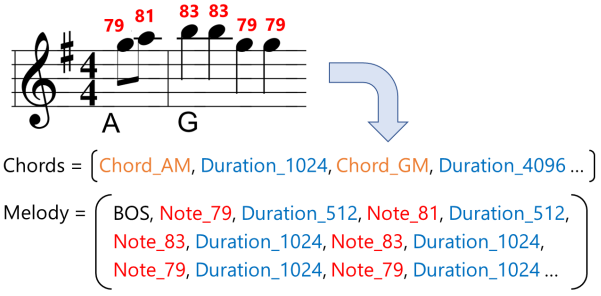


図 2 メロディ・コード系列のデータ表現
Fig. 2 Input representation of melody and chord data.

に VAE を挟み込む形となっている．Transformer encoder と Transformer decoder の両方ともに，メロディの系列データが入力される．Transformer encoder に入力されたメロディ系列が VAE の潜在空間にエンコードされ，そこからサンプリングされた潜在表現と，Transformer decoder に入力された予測中のメロディ系列の特徴量を統合し，後続するメロディトークンを予測する．このような構造によって，メロディ系列全体の抽象表現を VAE の潜在空間内に獲得させ，それを操作することによって，生成されるメロディ系列に変化を生じさせることができる．そして，Wu ら [5] は，Transformer decoder の各 self-attention 層に生成を条件づける embedding を加えることで，音楽特徴の操作をより強力に行う手法を提案した．特に，VAE を用いて獲得した小節ごとの潜在表現を操作し，提案された手法で Transformer decoder を条件づけることで，音楽のリズムやハーモニーの複雑度を小節ごとに変化させるスタイル変換を実現した．この手法を単純に応用することで小節ごとにコードを変化させることも可能だが，コードの前後関係を考慮することはできない．

以上のように，VAE と Transformer を組み合わせ，それぞれの長所を活かしたモデルは複数存在するが，本研究が目的とする，コードの前後関係を考慮した条件付けの下でのメロディ生成と，潜在空間における操作可能性を両立させた事例は存在していない．

3. 方法

3.1 モデル構造

本研究の目的は，コード進行に基づくメロディ生成とメロディの操作可能性を両立するモデルを提案することである．したがって，コードの系列データとメロディの系列データを入力とする，encoder-decoder 構造の Transformer の中に VAE を組み込んだモデル（**図 1**）を提案する．先行事例に見られる，Transformer と VAE を組み合わせた研究 [5], [9] とは異なり，本研究の提案モデルは，Transformer decoder の中間部分に VAE をはさみこむ形になっている．以下，モデル構造と入出力関係について，設計の意図を含めて詳細に説明する．

Transformer encoder の入力は，長さ L_{src} のコード情報の系列データ c_i $(i = 0, 1, ..., L_{\mathrm{src}} - 1)$ であり，Transformer decoder の入力は，長さ L_{tgt} のメロディ情報の系列データ m_i $(i = 0, 1, ..., L_{\mathrm{tgt}} - 1)$ である．encoder-decoder 構造の Transformer はこれらの入力を受け取り，Transformer decoder に入力された系列データの自己回帰問題を解くように訓練される．すなわち，Transformer decoder に入力されるメロディ系列 m_i $(i = 0, 1, ..., L_{\mathrm{tgt}} - 1)$ から m_i $(i = 1, 2, ..., L_{\mathrm{tgt}})$ を予測する．Transformer はこの予測誤差 $\mathcal{L}_{\mathrm{pred}}$ を最小化するように訓練される．

表 2 コード親和度の算出に使われる点数表
Table 2 Scoring table for calculation of chord affinity

	ルートからの距離 (半音の数)											
コードの種類	0	1	2	3	4	5	6	7	8	9	10	11
メジャーコード	10	-2	-1	-2	10	-5	-2	10	-2	-1	-2	0
マイナーコード	10	-2	-1	10	-2	-5	-2	10	-1	-2	0	-2

VAE は Transformer decoder の中間部分に加えられている．入力 m_i $(i = 0, 1, ..., L_{\text{tgt}} - 1)$ に対する Transformer decoder の中間表現 h_i $(i = 0, 1, ..., L_{\text{tgt}} - 1)$ を VAE の潜在空間にエンコードし，サンプリングを行った後，再構成する．このとき，VAE は中間表現の各要素 h_i に対して独立に適用する．この VAE による再構成された中間表現 $\hat{h}_i$ $(i = 0, 1, ..., L_{\text{tgt}} - 1)$ を再度 Transformer decoder の後半部分に入力する．ここで，Transformer decoder の前半部分は，メロディ系列の self-attention 計算のみを行い，VAE を過ぎた後の Transformer decoder の後半部分は，VAE の出力に対する self-attention 計算に加えて，Transformer encoder の出力を用いた source-target attention 計算も行うことに注意されたい．VAE は，中間表現の再構成問題を解くべく，h_i と $\hat{h}_i$ の間の再構成損失 $\mathcal{L}_{\text{recon}}$ を用いて訓練される．

このようにコードの系列データを Transformer encoder の入力，メロディの系列データを Transformer decoder の入力とすることで，コード進行の条件付けのもとでのメロディ生成が可能になる．Transformer encoder を通してコード系列に対して self-attention が適用されるので，ここで提案したモデルはコードの前後関係を考慮することができる．また，Transformer encoder からの入力を受け取る前の Transformer decoder の中間部分に VAE を挟み込むことで，VAE の学習がコード情報に依存せず，メロディ情報のみに依存した中間表現 h_i の再構成を行うことができる．それによって，VAE の潜在表現にコード進行の条件付けの情報が混入することを防ぐことができる．

本研究の提案手法では，式 1 に示す損失関数を利用して，Transformer と VAE を同時に訓練する．ここで，$\mathcal{L}_{\text{kld}}$ は潜在表現の分布と標準正規分布との間の KL ダイバージェンスである．

$$\mathcal{L}_{\text{all}} = \mathcal{L}_{\text{pred}} + c_{\text{kld}} * \mathcal{L}_{\text{kld}} + c_{\text{recon}} * \mathcal{L}_{\text{recon}} \quad (1)$$

最後に，本研究で用いるハイパーパラメータ及び訓練条件を記す．Transformer encoder は 2 層の self-attention 層を持ち，その次元数は 64 である．Transformer decoder は合計 4 層の self-attention 層 (VAE の前後に 2 層ずつ) と 2 層の source-target attention 層（VAE の後のみ）を持ち，その次元数は 256 である．attention 層内のヘッド数は 2 で共通とする．VAE encoder/decoder はそれぞれ 2 層の線形層，潜在変数の次元数は 16 とする．コード系列・メロディ系列の長さはそれぞれ $L_{\text{src}} = 256$，$L_{\text{tgt}} = 768$ である．モデルの訓練には Adam[17] を使用し，学習率は初期値を 10^{-4} として cosine annealing によるスケジューリングを行う．$\epsilon = 0.1$ の label smoothing を適用し，損失関数の比例定数は $c_{\text{kld}} = 0.25, c_{\text{recon}} = 0.25$ とする．バッチサイズは 20 であり，損失関数が 10 エポック連続で上昇するまで訓練を行う (early stopping)．これらのハイパーパラメータや訓練条件は，予測精度や生成結果を観察しつつ，簡単なグリッドサーチを行うことで適切なものを選択した．

3.2 データセットとデータ表現

データセットには，イギリスとアメリカの民謡を集めた Nottingham dataset [18] を用いる．このデータセットは monophonic なメロディの情報と，人手でラベル付けされたコードの情報を含んでおり，コード進行に条件づけられたメロディ生成を限りなくシンプルな形で試すのに適している．元のデータセットに含まれる 1034 曲の中から後述のデータ表現に変換できなかった 91 曲を除外した上で，残りのうち 843 曲を訓練用，50 曲をバリデーション用，50 曲をテスト用のデータセットとする．また，訓練用データセットに含まれる曲に対して-5 から 6 のピッチ変更（音の高さの平行移動）を施すことで，データ数を 12 倍にするデータ拡張を行った．

メロディ情報・コード情報のデータ表現形式に関しても，ここでは比較的単純なものを採用する．メロディ情報については，基本的に 1 つの音符に対して 2 つ 1 組のトークンが対応する．各トークンはそれぞれ音符の高さ (Note トークン) と長さ (Duration トークン) を表しており，**図 2** のように Note トークンと Duration トークンが交互に配置されることで一連の monophonic なメロディが表現される．同様に，コード情報はコードの種類を表す Chord トークンと Duration トークンの組によって表される．加えて，メロディ系列の最初には曲の始まりを表す BOS (Beginning-of-sentence) トークンが付加されているほか，入力データの長さをモデルの次元に合わせるために適宜 padding が用いられている．トークンは合計で 142 種類存在するが，その一覧を**表 1** に示す．

元のデータセットは多様な種類のコードを含んでいるが，本研究では {dim, m7, m7(-5), dim7} などのコードを

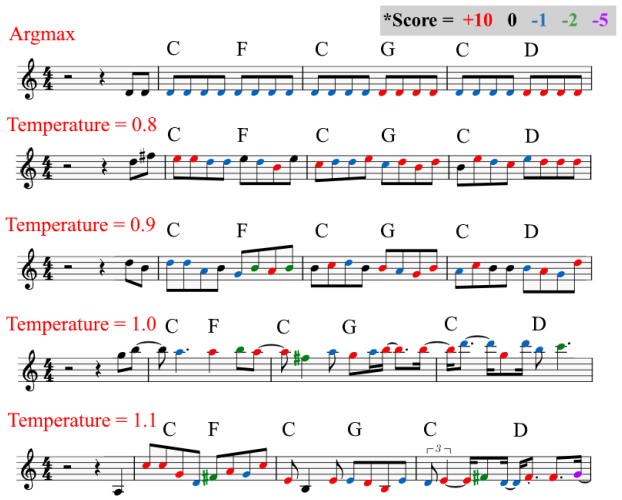


図 3 異なるサンプリング設定におけるメロディの生成例．各音符はそのコード親和度のスコアに応じて色分けされている．温度パラメータが大きいほど，不規則なリズムやコード外の音が出力される．

Fig. 3 Generated melodies for various sampling settings.

マイナーコード，{aug, 7, M7, sus2, sus4} などのコードをメジャーコードとみなすことで，ほとんどのコードを 24 種類 (ルート音は 12 種類ある) に帰着させた．その他の特殊なコードは分類不可として 25 種類目の Chord トークンを割り当てた．また，Duration トークンについては，元のデータセットに出現する音符やコードの長さのみを列挙し，トークンとして今回のデータ表現に加えている．

なお，以上のデータセットとデータ表現を採用する場合，$(L_{\mathrm{src}}, L_{\mathrm{tgt}}) = (256, 768)$ の設定では基本的に各曲のコード系列・メロディ系列を一度に提案モデルに入力できる．

3.3 評価

訓練済みモデルの生成結果は，本研究で新たに導入するコード親和度 (chord affinity) によって主に評価する．コード親和度は，与えたコード進行と生成されたメロディの相性を簡易的に定量評価するための指標であり，コードに含まれる音がメロディに多く使われるほど親和度の値は高くなる．具体的には，メロディの各音符に対して，その時指定されているコードのルート音との相対距離 (半音の数) およびコードの長短 (メジャー/マイナー) に応じて**表 2** の通りに点数が割り当てられる．この点数の平均が，あるコード系列とメロディ系列のコード親和度として定義される．なお，表 2 の点数は，コードの自動判定アルゴリズムを提供している `chorder` ライブラリ[*1]の実装を参考にして設定している．

ある訓練済みモデルがコード同士の前後関係を考慮してメロディを生成しているかをコード親和度のみで判定することはできないが，コード進行の条件付けによって出力結果が妥当な形で操作されているかどうかを見る上では有効な指標となっている．

[*1] https://github.com/joshuachang2311/chorder

表 3 異なるサンプリング設定におけるモデル評価

Table 3 Evaluation of model for various sampling settings

評価対象	コード親和度	生成失敗率
完全ランダム	1.083	—
提案モデル ($\tau = 1.0$)	4.016	0.42
提案モデル ($\tau = 0.9$)	4.479	0.18
提案モデル ($\tau = 0.8$)	4.722	0.14
提案モデル (Argmax)	6.382	0.02
真のデータ	6.428	—

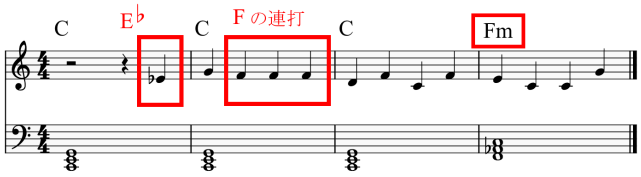


図 4 後半のコードが前半のメロディに影響を与えている生成例

Fig. 4 Example of a chord's global effect.

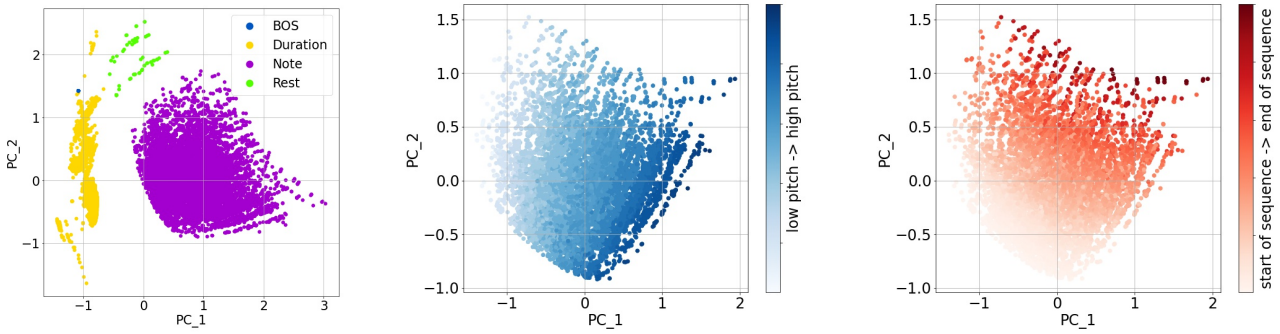
4. 実験結果

4.1 コード進行による条件づけの下での生成実験

BOS トークンのみを含む長さ 1 のメロディ系列を Transformer decoder の初期入力とし，コード進行の条件付けのもと自己回帰を行うことによってメロディを生成する．モデルが出力する分布からサンプリングをすることでメロディを決定するが，このときのサンプリング方法や温度パラメータ τ の値によって最終的に得られるメロディの質は大きく変わる．テスト用データセットに含まれる曲のコード進行を Transformer encoder の入力としてメロディ生成を実行した場合の生成例を**図 3** に，評価結果を**表 3** に示す．条件によっては，得られた出力系列が文法を満たさないために楽譜に変換できない場合がある．これらの結果は「生成失敗」とみなし，その発生率も表 3 に記録する．

表 3 より，訓練済みモデルが生成するメロディのコード親和度はランダムに音符を並べた場合と比べて安定的に高く，コード情報が生成結果に妥当な形で反映されている．図 3 の生成例において，赤色で示されるコード内の音符が多く出力されていることからも，このことが確認できる．一方で，テスト用データセットに含まれる真のデータ (メロディ系列) から算出されるコード親和度の値には到達しておらず，データセットの音楽的特徴を完全には学習することができていないと言える．

図 3 の生成例は，生成時のサンプリングの条件によって得られるメロディ出力が質的に異なることを表している．Argmax サンプリングではモデルが同一の音符を出力し続ける振る舞いに収束してしまうため，コード親和度の値は高いが実用性は無い．温度パラメータを低め ($\tau = 0.8, 0.9$)



(a) 4 種類のトークン　(b) Note のみ（音の高さの違いで色分け）　(c) Note のみ（位置の違いで色分け）

図 5　テスト用データセット内のメロディトークンの VAE 潜在表現に対する PCA の結果
Fig. 5　PCA plots of VAE's latent representations for melody tokens in the test dataset.

に設定して確率的サンプリングを行うと，コードに沿ったメロディが安定して出力されるが，リズムは単調になりがちである．温度パラメータを高くする ($\tau = 1.0, 1.1$) とより多様なリズムが得られるが，コードとの間で不協和音 (i.e. 図 3 における緑色・紫色の音符) が発生する頻度が増えるほか，文法ミスなどによる生成失敗率も上昇する．モデルの学習状況が改善すればこのトレードオフは緩和されると考えられるが，依然として温度パラメータは生成結果を左右する重要なハイパーパラメータとなる．

4.2　コードの前後関係を考慮したメロディ生成

4.1 節では，提案モデルの生成結果にコード情報が反映されていることを確認した．本節では，コードの前後関係を考慮したメロディ生成が可能かを定性的に考察する．

図 4 は，C → Fm というやや変わったコード系列を入力としてモデルに与えたときの生成例である．C コードでは稀である E♭や F の連打という要素が 1，2 小節目で現れていることから，4 小節目の Fm コードが前半のメロディに大きく影響していると推察される．

コードを小節ごとに対応させるようなモデルでは，メロディを介して前半のコード情報が後半に伝わることはあっても，後半のコード情報が前半に影響を与えることはないので，図 4 のような挙動が現れることはない (注釈として，双方向 RNN などを使う場合は必ずしもこの限りではない)．対して，本研究のモデルでは，その Transformer encoder の構造から推測される通り，コード系列全体の情報に基づいてメロディを生成できることが確かめられた．したがって，コードの前後関係も含めたコード系列とメロディ系列の複雑な関係を，原理的にはデータセットから学習することができる．一方で，系列全体のコード情報が条件付けに混ざってくるため，コードの指定による局所的な操作性が損なわれている可能性も考えられる (図 4 の例は，4 小節目の Fm コードが 1，2 小節目の C コードをオーバーライドしていると解釈することもできる)．局所的な (小節単位の) 操作性を担保しつつ，コード進行の大域的情報も同時に考慮しながらメロディ生成を行うのが提案モデルの理想的な振る舞いだが，そのためには追加の工夫が必要であるかもしれない．

4.3　VAE の潜在空間の可視化

本節では，提案モデルの VAE における潜在空間がどのような情報を保持しているかを検証する．ある曲のメロディトークンを m_i $(i = 0, 1, ..., L_{\mathrm{tgt}} - 1)$ としたとき，これを Transformer decoder に入力し，VAE の中間層が出力する平均値を VAE の潜在表現 z_i $(i = 0, 1, ..., L_{\mathrm{tgt}} - 1)$ とする．テスト用データセットは全 50 曲あり，メロディ系列に用いられるトークンの総数（PAD トークンを除いた，BOS, Duration, Note, Rest のいずれかのトークンの総数）は 15130 個である．これらを Transformer decoder に入力し，VAE の潜在空間における潜在表現の集合に対して，主成分分析 (Principal component analysis, PCA) を行った結果が**図 5**(a) である．図の軸は第 1 主成分（PC_1），第 2 主成分（PC_2）である．この図から，Transformer decoder 内の VAE の潜在空間において，トークンの種類を区別できていることがわかる．

次に，Transformer decoder に対する入力トークン m_i として，Note トークンを入力した場合の，VAE の潜在空間における潜在表現 z_i の集合に対して，主成分分析を行った．テスト用データセット全 50 曲のメロディ系列に用いられる Note トークンの総数は 7444 個であり，これらの潜在表現の点集合に対して主成分分析を行い，第 1 主成分，第 2 主成分を軸として散布図を描画した結果が図 5(b),(c) である．図 5(b) は，入力された Note トークン m_i の値，すなわち音の高さを示す値に関して，小さい方から大きい方へ順に白色から青色へと色の勾配を付けている．図 5(c) は，入力された Note トークン m_i の位置 i，すなわち入力メロディ系列の中での時間方向における始めの位置から終わりの位置の順に，白色から赤色へと色の勾配を付けてい

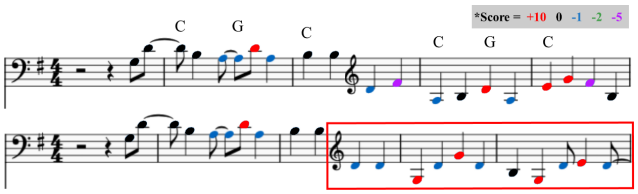


図 6 潜在空間に与えた摂動によってメロディが変化している例
Fig. 6 Example of adding perturbation to the latent space.

る．これら二つの図を見ると，色の勾配の方向がおよそ直交している様子がわかる．これは，第 1 主成分，第 2 主成分からなる潜在空間において，入力された Note トークンの音の高さと位置の情報が連続的に配置され，構造化されていることを示している．

4.4 VAE の潜在空間における生成メロディの操作

本節では，提案モデルの VAE において得られた潜在表現に摂動を加えることで，生成されるメロディ系列に操作を加えられるかを検証する．

図 6 は，VAE の潜在表現に摂動を加えることで生成されるメロディが変化した例を示している．図下段の赤枠の範囲を生成する過程において，図 5(a) で計算された第 1 主成分と同じ方向ベクトルを持ち，大きさが 0.1 であるような摂動が加えられている．これにより，赤枠以前のメロディが同一であるにも関わらず，全く異なるメロディが続きとして得られることが分かる．また，同様の実験を 30 回繰り返し，摂動の追加によってコード親和度が減少するかを検証したが，変化は見られなかった（コード親和度の平均値および標準偏差は摂動無しの条件で 4.081 ± 0.464，摂動ありの条件で 4.191 ± 0.377）．したがって，潜在空間における摂動の追加によって，コード進行とメロディの対応関係を維持したまま，生成されるメロディの多様性を増幅させることが可能である．

5. 議論

はじめに，コード進行を考慮したメロディ生成をより実用的な性能で実現するために必要な改善について議論する．第一に挙げられるのは，データ表現の改善である．本研究で採用したデータ表現では，ある音符やコードの楽譜上の正確な位置を求めるためには Duration トークンの値の累積和を取るような操作が必要になるため，4.2 節で述べたような局所的な操作性が損なわれている可能性がある．音楽の時間的構造をより明確に取り扱うことができる REMI [19] などのデータ表現を採用することで，この問題の改善が可能である．

第二に，適切な大規模データセットを用意することも重要な課題である．コード進行の前後関係を考慮する場合，その組み合わせの数はコード自体の種類数に対して爆発的に増加するため，その学習を可能にするためには多数の楽曲データが必要である．楽譜や MIDI データからコードを自動的に判別するアルゴリズムは存在するが，これらの手法は局所的に最適なコードを当てはめるものであるため，コードの前後関係に関する情報が一部抜け落ちてしまう可能性がある．したがって，人手によってラベル付けされたコード情報 (e.g. Web 上に公開されているコード譜) とメロディを対応付けたデータセットを用意することが望ましい．コード進行の使われ方は音楽ジャンルによっても異なるので，データセットを構成する曲のジャンルも考慮する必要がある．

第三に，本研究ではメロディの自動生成においてコードの前後関係が考慮されているかを定量的には評価できていない．音楽理論的な正解が必ずしもあるわけでは無いため厳密な指標化は難しいが，代表的なコード進行に関して現実の楽曲と統計的性質を比較することが有望な方針である．

最後に，VAE の潜在空間における操作に関して，本研究では，VAE の潜在表現に対して摂動を加えることで，コード親和度を下げることなく，生成メロディ系列に変化を生じさせられることを示したが，それがどのような性質の変化であるのかを検証することはできていない．今後は，VAE の潜在表現に対する操作を行いつつ，曲全体の長さや音の高さの幅，繰り返し構造などの性質に対する定量的な評価を行うことで，加えた摂動と生成メロディ系列への影響の関係性を検証したい．

6. 結論

本研究では，ユーザが指定するコード進行によって条件づけられる自動音楽生成を実現するために Transformer と VAE を組み合わせた深層学習モデルを提案し，その性能を評価した．コード親和度の評価により，提案モデルがコード系列に基づくメロディ生成を行っていることが確かめられた．また，コード系列全体の情報がメロディ生成に関与している例を提示し，コードの前後関係を考慮したメロディ生成が原理的に可能であることを論じた．また，入力メロディ系列に対する VAE の潜在表現を可視化し，VAE の潜在空間において，入力メロディトークンの情報が構造化された状態で表現されていることを示した．そして，VAE の潜在表現に対して摂動を加えることにより，生成メロディ系列に対する操作を行えることを確認した．今後はデータ表現の方法を改善した上で，コード進行の影響がより色濃く反映されるポップ音楽等の大規模データセットに本手法を適用し，その実用性をさらに詳しく検証する必要がある．また，生成されるメロディ系列を解釈可能な形で操作できるように提案手法を改良することも今後の課題である．

謝辞 本研究は，先端人工知能学教育寄付講座，および東京大学の次世代知能科学研究センター (AI センター) の補助を受けて行われました．

参考文献

[1] Meade, N., Barreyre, N., Lowe, S. C. and Oore, S.: Exploring conditioning for generative music systems with human-interpretable controls, *arXiv preprint arXiv:1907.04352* (2019).
[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I.: Attention is all you need, *Advances in Neural Information Processing Systems*, Vol. 30 (2017).
[3] Huang, C.-Z. A., Vaswani, A., Uszkoreit, J., Shazeer, N., Simon, I., Hawthorne, C., Dai, A. M., Hoffman, M. D., Dinculescu, M. and Eck, D.: Music transformer, *arXiv preprint arXiv:1809.04281* (2018).
[4] Christine, P.: MuseNet, *OpenAI*, Last updated 25 Apr. 2019, https://openai.com/blog/musenet/.
[5] Wu, S.-L. and Yang, Y.-H.: MuseMorphose: Full-Song and Fine-Grained Music Style Transfer with One Transformer VAE, *arXiv preprint arXiv:2105.04090* (2021).
[6] Kingma, D. P. and Welling, M.: Auto-encoding variational bayes, *arXiv preprint arXiv:1312.6114* (2013).
[7] Roberts, A., Engel, J., Raffel, C., Hawthorne, C. and Eck, D.: A hierarchical latent vector model for learning long-term structure in music, *International Conference on Machine Learning*, PMLR, pp. 4364–4373 (2018).
[8] Kawai, L., Esling, P. and Harada, T.: Attributes-aware deep music transformation, *Proceedings of the 21st International Society for Music Information Retrieval Conference* (2020).
[9] Jiang, J., Xia, G. G., Carlton, D. B., Anderson, C. N. and Miyakawa, R. H.: Transformer vae: A hierarchical model for structure-aware and interpretable music representation learning, *IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, pp. 516–520 (2020).
[10] Yang, L.-C., Chou, S.-Y. and Yang, Y.-H.: MidiNet: A convolutional generative adversarial network for symbolic-domain music generation, *arXiv preprint arXiv:1703.10847* (2017).
[11] Hadjeres, G., Pachet, F. and Nielsen, F.: Deepbach: a steerable model for bach chorales generation, *International Conference on Machine Learning*, PMLR, pp. 1362–1371 (2017).
[12] Chen, K., Zhang, W., Dubnov, S., Xia, G. and Li, W.: The effect of explicit structure encoding of deep neural networks for symbolic music generation, *International Workshop on Multilayer Music Representation and Processing*, IEEE, pp. 77–84 (2019).
[13] Wang, Z., Wang, D., Zhang, Y. and Xia, G.: Learning interpretable representation for controllable polyphonic music generation, *arXiv preprint arXiv:2008.07122* (2020).
[14] Yan, W., Zhang, Y., Abbeel, P. and Srinivas, A.: VideoGPT: Video Generation using VQ-VAE and Transformers, *arXiv preprint arXiv:2104.10157* (2021).
[15] Liu, D. and Liu, G.: A Transformer-Based Variational Autoencoder for Sentence Generation, *International Joint Conference on Neural Networks*, pp. 1–7 (2019).
[16] Wang, T. and Wan, X.: T-CVAE: Transformer-Based Conditioned Variational Autoencoder for Story Completion., *International Joint Conferences on Artificial Intelligence*, pp. 5233–5239 (2019).
[17] Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
[18] Foxley, E.: Nottingham Database, Last updated 12 Jan. 2021, http://www.chezfred.org.uk/freds/music/index.htm.
[19] Huang, Y.-S. and Yang, Y.-H.: Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions, *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 1180–1188 (2020).