

deepGTTM-III: グルーピング構造と拍節構造の自動獲得

浜中雅俊^{†1} 平田圭二^{†2} 東条敏^{†3}

概要: 本稿では、ディープラーニングに基づき音楽理論 GTTM のグルーピング構造と拍節構造の自動分析器について述べる。GTTM は 4 つの分析が直列に並んでいて、前段の分析結果が後段の分析に用いられる。また、後段の結果が前段へフィードバックされるループも存在している。しかし、我々がこれまでに構築してきた分析器では 4 つの分析が独立しておりフィードバックループも構成できていなかった。たとえば、deepGTTM-I と deepGTTM-II は深層学習を用いてグルーピング構造と拍節構造を独立に学習していた。本研究では、deepGTTM-I と deepGTTM-II のネットワークを統合したネットワークを持つ deepGTTM-III を用いてグルーピング構造と拍節構造を同時に学習することで、フィードバックループをネットワークの形で学習することを目指す。実験の結果、deepGTTM-III は、グルーピング構造では deepGTTM-I より高い性能であること、拍節構造では deepGTTM-II と同等の性能であることを確認した。

キーワード: A generative theory of tonal music (GTTM), グルーピング構造, 拍節構造, 深層学習

deepGTTM-III: Automatic Acquisition of Grouping and Metrical Structures

MASATOSHI HAMANAKA^{†1} KEIJI HIRATA^{†2} SATOSHI TOJO^{†3}

Abstract: This paper describes an analyzer that simultaneously learns the grouping and metrical structures on the basis of the generative theory of tonal music (GTTM) by using a deep learning technique. The GTTM is composed of four modules that are in series. The GTTM has feedback loop in which the former module uses the result of the latter module. However, each module is independent in previous GTTM analyzers, thus they do not form a feedback loop. For example, deepGTTM-I and deepGTTM-II independently learn the grouping and metrical structures by using a deep learning technique. In light of this, we present deepGTTM-III, a new analyzer that includes the concept of feedback that enables simultaneous learning of grouping and metrical structures by integrating both networks of deepGTTM-I and deepGTTM-II. Experimental results show that deepGTTM-III outperforms deepGTTM-I and deepGTTM-II.

Keywords: A generative theory of tonal music (GTTM), grouping structure, metrical structure, deep learning.

1. はじめに

本研究の最終的な目標は、音楽理論 GTTM [1]に基づいてメロディのタイムスパン木を自動的に獲得するシステムを構築することである。GTTM は、聴取者が音楽を理解する直観を形式的に記述するための理論で、グルーピング構造分析、拍節構造分析、タイムスパン簡約、プロロンゲーション簡約という 4 つのサブ理論から構成されており、各サブ理論はそれぞれ、グルーピング構造、拍節構造、タイムスパン木、プロロンゲーション木を獲得する (図 1)。

グルーピング構造分析は、連続したメロディをフレーズやモチーフなどに階層的に分割するもので、長いメロディを歌うときにどこで息継ぎすべきかを見つけるような分析である。拍節構造分析は、4 分音符/2分音符/1 小節/2 小節/4 小節など各拍節レベルにおける強拍と弱拍を同定するもので、聴取者が曲に合わせて手拍子を打つタイミングや指揮者がタクトを振るタイミングを求めるような分析であ

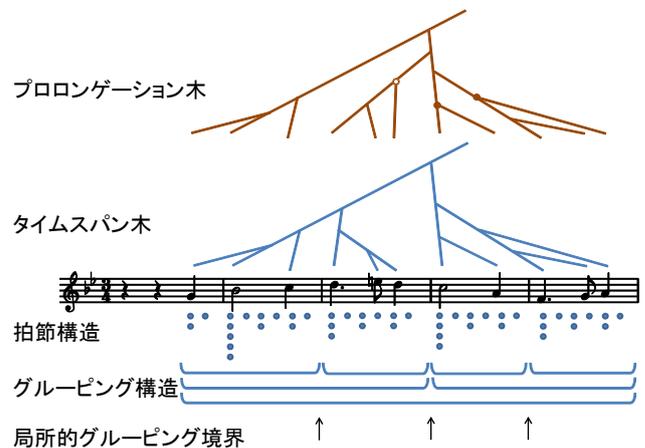


図 1: 局所的グルーピング境界, グルーピング構造, 拍節構造, タイムスパン木, プロロンゲーション木

る。タイムスパン簡約は、メロディの重要な部分と装飾的な部分を分離するもので、構造的に重要な音が幹になるような 2 分木 (タイムスパン木) を求める分析である。

タイムスパン木は、演奏の表情づけや[2], メロディの生成[3], メロディの要約[4]などに用いられる。また、2 つのメロディの中間のメロディを求めるメロディモーフィング[5]やメロディの予測[6]に用いることができる。しかし、

^{†1} 理化学研究所
RIKEN

^{†2} 公立はこだて未来大学
Future University Hakodate

^{†3} 北陸先端大学院大学
JAIST

我々が開発してきたタイムスパン木分析器は精度が低かったため[7,8], これらのシステム[2-6]を使うためには音楽家が手作業で分析したタイムスパン木が必要であった。

GTTM を計算機上に実装するためには, 3つの問題が存在する。

- ルールの競合

GTTM の各サブ理論は次の 2 種類のルールで表現されている: 構成ルールはグルーピング構造や拍節構造が満たすべき条件や制約を記述したルールであり, 選好ルールは構成ルールを満たす複数の構造のいずれかが好ましいかを示すルールである。選好ルールを適用する際には, 複数のルール間での優先度が決まっていなため, 競合が起きることがある。

図 2 は, 拍節選好ルール 5c (MPR5c) と 5a (MPR5a) という 2 つのルールが競合している例である。MPR5c は, 比較的長いスラーが強拍になる拍節構造を優先し, MPR5a は, 比較的長い音価の音が強拍になる拍節構造を優先する。拍節構成ルール 3 (MWR3) は, 強拍が 2 拍ごとあるいは 3 拍ごとに来ることを定義しており, 図 2 の第 1 音と第 2 音の両方を強拍にすることはできない。

そこで我々は, ルールの優先順位を操作するための調節可能なパラメータを 46 個備えたタイムスパン木分析器 ATTA を構築した。そして, ATTA を用いて調節可能なパラメータを適切に調節することで音楽家が作成した正解データに近いグルーピング構造と拍節構造が獲得できることを確認した[7]。しかし, パラメータの調整は音楽知識を有する音楽家が適切に行う必要があった。一方, 調節可能なパラメータを持たない FATTA は, ATTA から出力される構造に対して安定度を算出し最も安定する構造を出力する[8]。FATTA は正解データに近い拍節構造を獲得できたが, グルーピング構造とタイムスパン木については性能が低かった。

これに対し, 本稿で提案する deepGTTM は, 深層ネットワークがルールの優先順位を学習する。

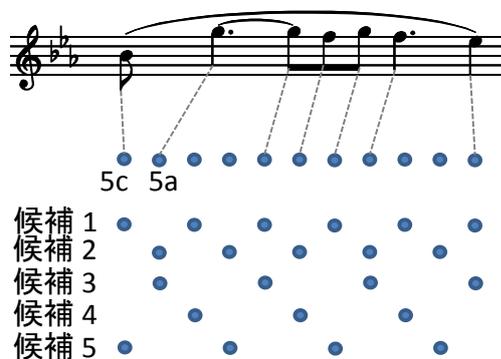


図 2: ルールの競合

- ボトムアップなルールとトップダウンなルール
 GTTM には, ボトムアップなルールとトップダウンなルールがある。たとえば, 前の音のオフセットと次の

音のオンセットの間隔が広いほどグルーピング境界になりやすいことを定めたグルーピング選好ルール 2 (GPR2) はボトムアップなルールである。一方, 大きなグループが 2 つのサブグループに分割される場合, 同じ長さのサブグループになりやすいことを定めた GPR5 はトップダウンなルールである。

ボトムアップな処理で構成された ATTA や FATTA が出力する構造ではしばしば高次の階層構造に誤りがあった。一方, σ GTTM [9], σ GTTMII [10] および σ GTTMIII [11] はトップダウンな処理のみであった。 σ GTTM と GTTMII は GTTM 分析に機会学習を導入しグルーピング境界の検出を可能としていたが, タイムスパン木の獲得はできなかった。 σ GTTMIII は, GTTM データベース[12]にある 300 曲のタイムスパン木を PCFG で学習することで, タイムスパン木の自動獲得を可能としていた。 σ GTTMIII はこれまで構築したタイムスパン木分析システムの中で最も高性能であった。しかし, 分析結果を応用システム[2-6]で用いることを考えると, これらの分析システム[7-11]の性能は不十分であった。

これに対し, 本稿で提案する deepGTTM では学習データからトップダウンなルールとボトムアップなルールの両方を学習する。

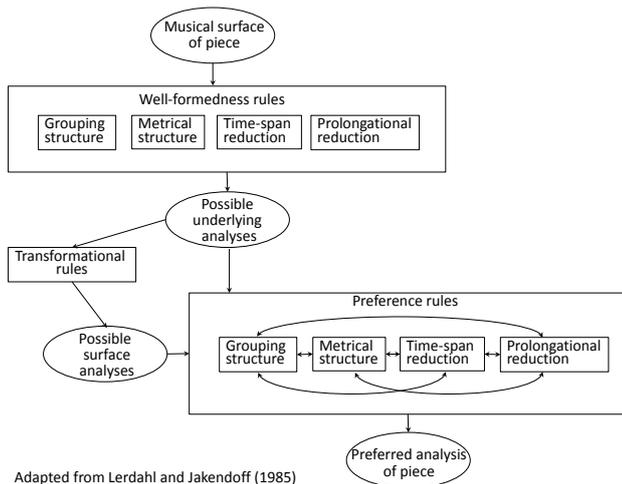
- フィードバックループ

GTTM の 4 つの分析は直列に繋がっている。すなわち, 前段の分析結果が後段の分析に用いられる。一方, GTTM には後段の分析結果が前段の分析に用いられるフィードバックループが存在する。たとえば, GPR7 は, タイムスパン木やプロロンゲーション木が安定するようなグルーピング構造を優先する。別の例では, MPR9 は, タイムスパン木と最も矛盾しなくなるような拍節構造を優先する。しかし, 従来の GTTM 分析器[7-14]では, 各分析が完全に独立しておりフィードバックループが構成できていなかった。たとえば, deepGTTM-I [13] および deepGTTM-II [14] では, 深層学習を用いてそれぞれ独立にグルーピング構造と拍節構造を学習していた。

図 3 は, 著者らによって GTTM の概要を示したもので, 左上が構成ルール, 右下が選好ルールである。選考ルールにあるフィードバックループをナイーブに実装した場合, 分析結果が発散して最適解が求まらない可能性がある。

2. 関連研究

近年, 深層学習は音楽情報検索の様々なタスクに利用されている[15-19]。FCN (Fully Convolutional Networks) に基づく自動タグ付けシステムでは, 感情やジャンル, 楽器編成など高次の情報の予測を可能としていた[15]。Bottleneck



Adapted from Lerdaahl and Jakendoff (1985)

図 3：音楽理論 GTTM の概要

アーキテクチャーの深層ネットワークを用いたコード予測システムでは、SVM (Support Vector Machines) や HMM (Hidden Markov Models) を上回る性能を示していた[16]. 連続分布型 HMM と深層学習を組み合わせた手法に基づくコード推定システムでは、長大なコード進行の推定を可能としていた[17]. 深層の CNN (Convolutional Neural Network) に基づく音楽推薦システムでは、音響信号から潜在変数の予測性能で従来の手法に比べて明確に性能向上していた[18]. 教師付き学習済みのニューラルネットワークを用いたポリフォニーの自動採譜システムでは、音響に基づく教師なしモデルよりも性能が向上していた[19]. これらのシステムは[15-19], 他の機械学習手法を深層学習で置き換えることで従来に比べて性能が向上したことを示している.

一方、階層的な音楽構造を獲得する我々のタスクにおいては、従来の機械学習手法は有効ではなかった。なぜなら、楽譜とその音楽構造の関係を直接求めようとすると、その間の飛躍が大きすぎて求めることができないためである。そこで我々は、2段階で学習を進めることにした。まず、各選好ルールの適用について深層ネットワークに学習させる。次に、選好ルールを学習済みのネットワークを用いて楽譜とその音楽構造の関係を学習する。選好ルールの学習でネットワークに音楽知識が蓄積されることで、楽譜とその音楽構造との関係が学習可能となる。

3. deepGTTM-I,II, および III

ディープラーニングに基づく GTTM 分析器 deepGTTM-I,II, および III を構築した。deepGTTM-I はグルーピング構造分析における局所的グルーピング境界[13], deepGTTM-II は拍節構造[14]の獲得を可能としていた。本稿で新たに提案する deepGTTM-III は、deepGTTM-I と II を統合したものである。

GTTM の分析に深層学習を用いることのメリットは以下の3点である。

- ボトムアップなルールとトップダウンなルールの両方が学習可能
 従来の GTTM 分析ツールは、研究者やプログラマーにより構築されてきた。GTTM の選好ルールには曖昧なものが多く、その実装は人によって異なる可能性がある。一方、deepGTTM は学習ベースのシステムであるため分析器の性能は、学習データおよび学習ネットワークに依存する。ボトムアップなルールとトップダウンなルールの両方を学習するため、ネットワークには分析エリア全体の楽譜情報を入力する。
- ルールの優先順序の学習
 ルールの優先度は曲のコンテキストに依存するため、適用されているルールのみからルールの優先度を求めようとしていた σ GTTM や σ GTTMII は分析誤りが多かった。一方、deepGTTM での深層ネットワークは入力された楽譜のコンテキストからルールの優先度をネットワークの重みとバイアスの形で学習する。
- 深層ネットワークに内在するフィードバックループ
 deepGTTM-III の深層ネットワークは、グルーピングと拍節の構造をマルチタスク学習するため、ネットワーク内に GTTM のフィードバックプロセスが内在される。deepGTTM-III はグルーピング構造と拍節構造で隠れ層を共有するため、グルーピング構造を学習する時でも拍節の構造を獲得するために重要な情報が伝播する。同様に、拍節構造を学習する時でもグルーピング構造を獲得するために重要な情報が伝播する。

3.1 ネットワーク構成

deepGTTM-I, II, III を DBN (deep belief network) を用いて構成する。図 4 は、グルーピング構造を獲得する deepGTTM-I の概要を示したものである。DBN の入力は、MusicXML にある各音符のオンセット・オフセット時刻、ピッチ、ベロシティを 0 から 1 の範囲となるよう正規化したものである。出力は、9 種類のグルーピング選好ルール (GPR2a, 2b, 3a, 3b, 3c, 4, 5, 6, 7) と局所的グルーピング境界

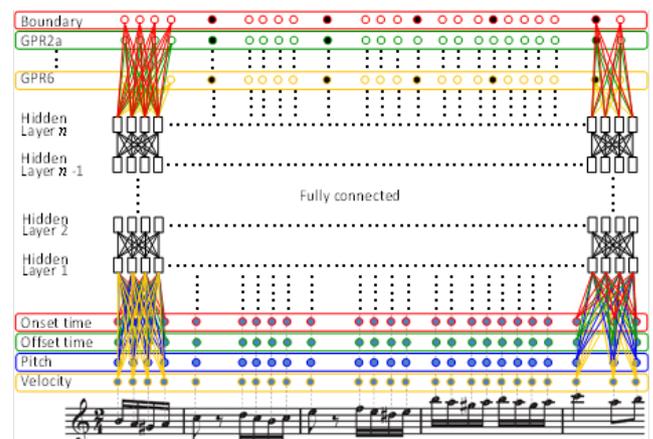


図 4：deepGTTM-I

の合計 10 個でマルチタスク学習を行う。各出力は、ルールが適用される (=1) と適用されない (=0), あるいは、グルーピング境界に隣接する (=1) と隣接しない (=0) の 2 つのユニットを持つ。

図 5 は、拍節構造を獲得する deepGTTM-II の概要で、その入力は、各音符のオンセット・オフセット時刻、ピッチ、ベロシティに加えて音楽家が分析したグルーピング構造である。グルーピング構造は隣接音がグルーピング境界の場合 1, そうでない場合 0 が入力され、各階層グルーピング構造はそれぞれ独立に入力される。出力は、7 種類の拍節選好ルール (MPR2, 3, 4, 5a, 5b, 5c, 5d) と 1 つのレベルの拍節構造の合計 8 個で、マルチタスク学習を行う。各出力は、ルールが適用される (=1) と適用されない (=0), あるいは、強拍である (=1) と弱拍である (=0) の 2 つのユニットを持つ。拍節構造は階層的なため、ある階層の拍節構造が求まると 1 層の隠れ層と次の階層の出力層を追加する。最終段の隠れ層と (n,n+1,..., n+h) と出力層とはロジスティック回帰で構成されている。すべての出力層は、1 層目から最終段までの隠れ層を共有している。

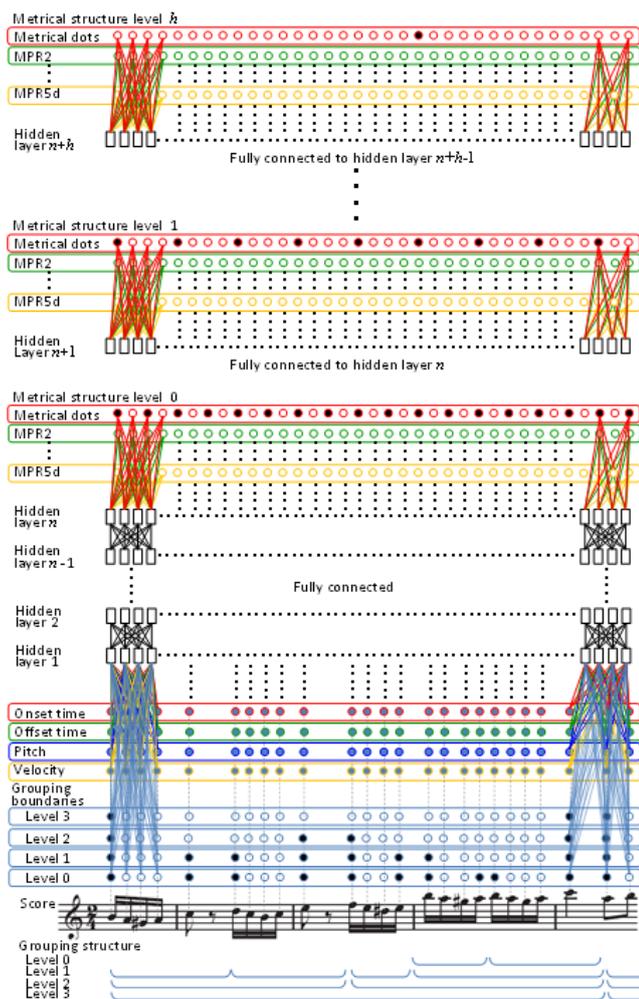


図 5 : deepGTTM-II

図 6 は、グルーピング構造と拍節構造の両方を獲得する deepGTTM-III の概要である。deepGTTM-III の入力 is deepGTTM-I と同じであり、出力は、deepGTTM-I と deepGTTM-II を合わせたものとなっている。

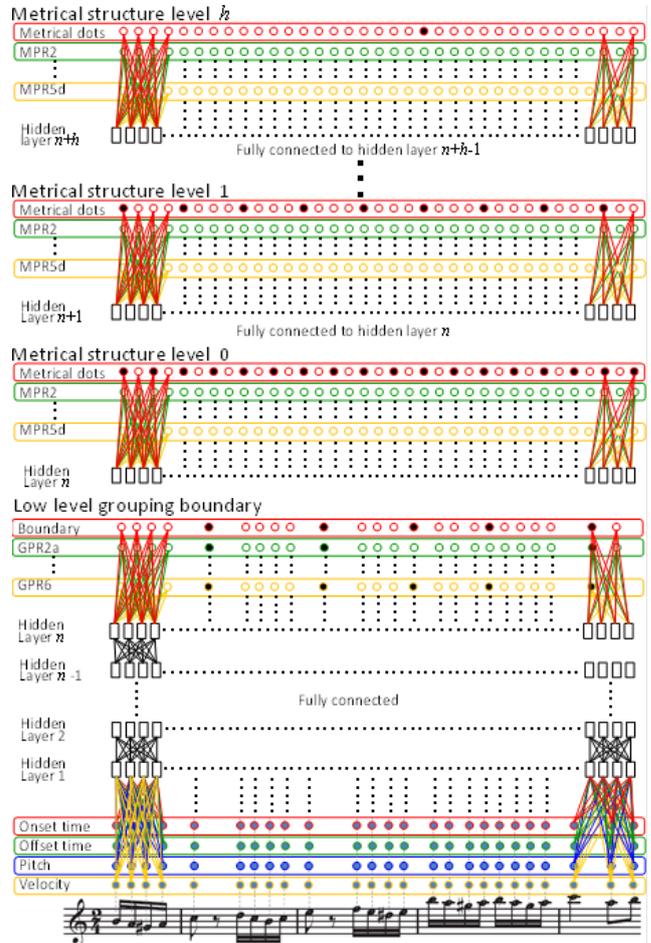


図 5 : deepGTTM-III

3.2 ネットワークの学習

本項では、局所的グルーピング境界と拍節構造を深層ネットワークで学習する方法について述べる。

(1) プレトレーニング

ネットワークに入力データの性質を覚えさせるプレトレーニングでは大量の教師なしデータを必要とする。そこで、MakeMusic 社の MusicXML のページからリンクが張られているページから 15,000 曲の MusicXML を収集する(図 5a) [20]。具体的には、まず Web 巡回スクリプトで 5 リンク以内を巡回し、XML のリストを収集する。つぎに、ファイル名から明らかに MusicXML でないものを除き、XML のリストをダウンロードする。最後に、ダウンロードしたファイルのうち、MusicXML でないものを削除した。

(2) ルールの適用およびグルーピング・拍節構造の学習

GTTM データベースには 300 曲の MusicXML と、音楽家によるグルーピング・拍節構造の分析結果およびグルーピング・拍節選好ルールの適用結果がある。しかし、深層のニューラルネットワークのファインチューニングするため

には 300 曲では少なすぎるため、自動教師付けデータを作成する。自動教師付けデータは、ATTA を用いて GPR2a, 2b, 3a, 3b, 3c, 3d, および, MPR2, 3, 4, 5a, 5b, 5c, 5d を適用したものである。ルールが適用されるかどうかを決める調節可能なパラメータの値を決めれば、これらのルールの適用結果は楽譜からは一意に決まり、曖昧性がない。

教師付きデータは、GTTM を良く理解している 3 人の音楽家がクラシック曲から切り出した 8 小節の長さの 300 個のメロディの楽譜データと、それを GTTM に基づき手作業で分析したグルーピング・拍節構造分析データである。学習データを増加させるため、まず、楽曲全体を半音ずつ上下にシフトし、12 の key すべてにトランスポーズした曲を用意した。さらに、楽曲の全ての音符の音価を 2 倍, 4 倍, 8 倍, 0.5 倍, 0.25 倍, 0.125 倍にした曲を用意した。データの総数は 25,200 (=300x12x7) である。

グルーピング選好ルールおよび拍節選好ルールの優先順序は、自動教師付けデータおよび教師付きデータを用いてバックプロパゲーションにより学習される。deepGTTM-I および II は、複雑なネットワークである。すなわち、1 つのレベルの選好ルール、局所的グルーピング境界および拍節構造の学習はマルチタスク学習である。さらに、選好ルールは複数あるため多次元のマルチタスク学習となる。局所的グルーピング境界の学習は以下の 4 段階である。

- Step 1: 曲順をランダムシャッフルした後、1 曲ずつ上から下まで順に選択する。
- Step 2: 音の遷移の順をランダムシャッフルした後、1 遷移ずつ上から下まで順に選択する。
- Step 3: 選択された遷移が境界となっている (=1) か境界となっていない (=0) かを出力から入力へバックプロパゲーションして学習する。
- Step 4: Step2 において次の遷移あるいは Step1 において次の曲を選択する。

拍節構造の学習は以下の 4 段階である。

- Step 1: 曲順をランダムシャッフルした後、1 曲ずつ上から下まで順に選択する。
- Step 2: 拍の位置をランダムシャッフルした後、1 拍ごと上から下まで選択する。
- Step 3: 選択された拍が強拍となっている (=1) か弱拍となっている (=0) かを出力から入力へバックプロパゲーションして学習する。
- Step 4: Step2 において次の拍あるいは Step1 において次の曲を選択する。

選好ルールのマルチタスク学習は以下の 3 段階である。

- Step 1: 選好ルールの順をランダムシャッフルした後、上から下まで順に選択する。
- Step 2: 選択された選好ルールに対しマルチタスク学習を行う。
- Step 3: Step1 において次のルールを選択する。

(3) deepGTTM-III におけるグルーピング構造と拍節構造の学習

deepGTTM-III の深層ネットワークは、グルーピング構造と拍節構造をマルチタスク学習する。deepGTTM-II で拍節構造を獲得するためには入力にグルーピング構造が必要だったのに対し、deepGTTM-III では不要である点が異なる。すなわち、deepGTTM-III では局所的グルーピング境界を深層ネットワーク内で予測するため、グルーピング構造を外部から入力しなくても、予測したグルーピング境界を用いて拍節構造を獲得する。

deepGTTM-I と II のネットワークは、まず各選好ルールの適用を一定 epoch 回学習した後に構造の学習を行う。そして、ルールの学習と構造の学習を繰り返す。一方、deepGTTM-III では、グルーピング選好ルール、局所的グルーピング境界、拍節選好ルール、拍節構造の学習をそれぞれ一定 epoch 回ずつ繰り返す。グルーピングと拍節構造の学習が相互に行われるため 3 節で述べたようなフィードバックループがネットワーク内で非明示的に学習されていくことが期待される。

グルーピングおよび拍節選好ルールの一部は非常に似通っているために、相補的に学習されることが期待できる。たとえば、GPR6(parallelism) は、2 つ以上のメロディの断片が並列的である場合、それらが並列的なグループを構成することを優先する。一方、MPR1 (parallelism) は、2 つ以上のグループが並列的である場合、並列的な拍節構造となることを優先する。

他の例としては GPR2b と MPR5a がある。4 音からなる音列が n_1, n_2, n_3, n_4 であったとき、GPR2b(Attack-Point) は、 n_2 と n_3 のオンセット間隔がその前後の間隔より長い場合に、グルーピング境界となることを優先する。一方、MPR5a は、比較的長い音符が強拍になることを優先する。

4. 実験結果

GTTM データベースに収められた 300 曲のうち、200 曲を学習データ、100 曲を評価データとして deepGTTM-II の性能を適合率 P と再現率 R を組み合わせた F 値で評価した。

$$F \square = 2 \times \frac{P \times R}{P + R} \quad (2)$$

ただし、

適合率 P : 正解データのグループ/拍点と同じグループ/拍点、システムの出力に含まれている割合。

再現率 R : システムが出力したグループ/拍点、正解データに含まれている割合。

表 1 は、各層 3000 ユニット、11 層からなるネットワークを用いた場合の deepGTTM-III の F 値と、同じ層数・ユニット数での deepGTTM-I と II の F 値と比べたものである。局所的グルーピング境界の獲得では、deepGTTM-III が

deepGTTM-I の性能を上回った。一方、拍節構造の獲得では、deepGTTM-II の性能が僅かに deepGTTM-III を上回った。deepGTTM-II では入力に音楽家が作成した正しいグルーピング構造を必要としていたのに対し、deepGTTM-III は入力にグルーピング構造を必要としないメリットがある。すなわち、グルーピング構造の正解データが存在しない場合に deepGTTM-III は効果的である。

表 1 : deepGTTM-I,II と III の性能比較

| | グルーピング境界 | | 拍節境界 | |
|---------------------------|----------|--------|----------|---------|
| | deep-III | deep-I | deep-III | Deep-II |
| メロディ | | | | |
| 1. Grande Valse Brillante | 0.80 | 0.79 | 0.93 | 0.94 |
| 2. Moments Musicaux | 0.80 | 0.81 | 0.99 | 1.00 |
| 3. Turkish March | 0.77 | 0.76 | 0.96 | 0.98 |
| 4. Anitras Tanz | 0.78 | 0.76 | 0.90 | 0.90 |
| 5. Valse du Petit Chien | 0.80 | 0.78 | 0.99 | 0.99 |
| : | : | : | : | : |
| 平均 (100 曲) | 0.81 | 0.78 | 0.94 | 0.96 |

5. おわりに

本稿では、グルーピング構造分析器 deepGTTM-I と拍節構造分析器 deepGTTM-II を統合した deepGTTM-III について述べた。deepGTTM-I と II は、それぞれグルーピング構造と拍節構造を独立に学習する必要があったが、deepGTTM-III ではグルーピング構造と拍節構造を同時に学習する。実験の結果、局所的グルーピング境界の検出において、deepGTTM-III は deepGTTM-I より高い性能を示した。一方、拍節構造の獲得において、deepGTTM-II と III は F 値で 0.02 の性能差でありほぼ等しい性能となった。本研究は深層学習を用いて GTTM を計算機実装する試みの第一歩である。今度、深層学習を用いたタイムスパン木分析器の構築を行っていく。

謝辞 本研究の一部は JSPS 科研費 17H01847, 25700036, 16H01744, 23500145 の支援を受けたものです。

参考文献

[1] Lerdahl, F. and Jackendoff, R.: A Generative Theory of Tonal Music. MIT Press (1985)
 [2] Hirata, K. and Hiraga R.: Ha-Hi-Hun plays Chopin's Etude, In Working Notes of IJCAI-03 Workshop on methods for automatic music performance and their applications in a public rendering contest (2003)
 [3] Hirata, K., Matsuda, S., Kaji K., and Nagao K.: Annotated Music for Retrieval, Reproduction, and Sharing, In: Proceedings of the 2004 International Computer Music Conference (ICMC2004), pp. 584–587 (2004)

[4] Hirata K. and Matsuda S.: Interactive Music Summarization based on GTTM, In: Proceeding of the 2002 International Society for Music Information Retrieval Conference (ISMIR2002), pp. 86–93 (2002)
 [5] Hamanaka, M., Hirata, K., and Tojo, S.: Melody Morphing Method based on GTTM. In: Proceedings of the 2008 International Computer Music Conference (ICMC2008), pp. 155–158 (2008)
 [6] Hamanaka, M., Hirata, K., and Tojo, S.: Melody Extrapolation in GTTM Approach. In: Proceedings of the 2009 International Computer Music Conference (ICMC2009), pp. 89–92 (2009)
 [7] Hamanaka, M., Hirata, K., and Tojo, S.: Implementing ‘a generative theory of tonal music’, Journal of New Music Research, 35(4), 249–277 (2006)
 [8] Hamanaka, M., Hirata, K., and Tojo, S.: FATTA: Full automatic time-span tree analyzer, In: Proceedings of the 2007 International Computer Music Conference (ICMC2007), pp. 153–156 (2007)
 [9] Miura, Y., Hamanaka, M., Hirata, K., and Tojo, S.: Decision tree to detect GTTM group boundaries, In: Proceedings of the 2009 International Computer Music Conference (ICMC2009), pp. 125–128 (2009)
 [10] Kanamori, K. and Hamanaka, M.: Method to Detect GTTM Local Grouping Boundaries based on Clustering and Statistical Learning, In: Proceedings of the 2014 International Computer Music Conference (ICMC2014), pp. 125–128 (2014) 千葉則茂, 村岡一信. レイトレーシング CG 入門. サイエンス社, 1990, 282p.
 [11] Hamanaka, M., Hirata, K., and Tojo, S.: σ GTTM III: Learning-based Time-span Tree Generator Based on PCFG, In: Proceedings of the 11th International Symposium on Computer Music Multidisciplinary Research (CMMR 2015), pp. 303–317 (2015)
 [12] Hamanaka, M., Hirata, K., and Tojo, S.: Musical Structural Analysis Database Based on GTTM, In: Proceeding of the 2014 International Society for Music Information Retrieval Conference (ISMIR2014), pp. 325–330 (2014)
 [13] Hamanaka, M., Hirata, K., and Tojo, S.: deepGTTM-I: Local Boundary Analyzer based on a Deep Learning Technique, In: Proceedings of the 12th International Symposium on Computer Music Multidisciplinary Research (CMMR 2016), pp. 8–20 (2016)
 [14] Hamanaka, M., Hirata, K., and Tojo, S.: deepGTTM-II: Automatic Generation of Metrical Structure based on Deep Learning Technique, In: Proceedings of 13th Sound and Music Computing Conference (SMC2016), pp. 203–210 (2014)
 [15] Choi, K., Fazekas, G. and Sandler, M.: Automatic Tagging Using Deep Convolutional Neural Networks, In: Proceeding of the 2016 International Society for Music Information Retrieval Conference (ISMIR2016), pp. 805–811 (2016)
 [16] Zhou, X., and Lerch, A.: Chord Detection Using Deep Learning, In: Proceeding of the 2015 International Society for Music Information Retrieval Conference (ISMIR2015), pp. 52–58 (2015)
 [17] Deng, J., and Kwok, Y.: Hybrid Gaussian-HMM-Deep Learning Approach for Automatic Chord Estimation with Very Large Vocabulary, In: Proceeding of the 2016 International Society for Music Information Retrieval Conference (ISMIR2016), pp. 812–818 (2016)
 [18] Oord, A., Sander, D., and Benjamin, S.: Deep content-based music recommendation, In: Proceeding of the Advances in Neural Information Processing Systems 26 (NIPS 2013), pp. 2643–2651 (2013)
 [19] Sigtia, S., Benetos, E., and Dixon, S.: An End-to-End Neural Network for Polyphonic Piano Music Transcription, IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 24 (5), 927–939 (2016)
 [20] MakeMusic Inc., “Finale,” 2017, <http://www.finalemusic.com/>.