

# deepGTTM-II: ディープラーニングに基づく拍節構造分析器

浜中 雅俊<sup>†1</sup> 平田 圭二<sup>†2</sup> 東条 敏<sup>†3</sup>

**概要:** 本稿では、音楽理論 Generative Theory of Tonal Music (GTTM) の拍節構造分析を自動で獲得する分析器について述べる。これまで自動で拍節構造を分析するシステムも構築されているが、分析誤りが多く、音楽家による修正が必須であったためディープラーニングに基づく新たな分析器を構築する。音楽家が GTTM に基づき拍節構造を分析した 300 曲の分析データベースを我々は保有しているが、それだけでは学習データが不足なため楽譜と拍節構造の関係を直接学習することは困難である。そこで我々は、楽譜と拍節構造との関係を以下の 3 つのステップでマルチタスク学習するシステム deepGTTM-II を提案する。まず、1 万 5 千件の教師なし楽曲データを用いてネットワークのプレトレーニングを行う。次に、GTTM ルールの適用箇所を 1 万 5 千件の教師なし楽曲に我々がこれまでに構築した自動分析器を用いて自動でラベル付けした自動教師つき楽曲データを作成し、バックプロパゲーションにより教師付き学習を行う。最後に、300 曲の教師付きデータを用いてファインチューニングを行う。実験の結果、deepGTTM-II は、従来の GTTM 分析器と比べて高い性能であることを確認した。

## deepGTTM-II: Metrical Structure Analyzer based on Deep Learning Technique

Masatoshi Hamanaka<sup>†1</sup> Keiji Hirata<sup>†2</sup> Satoshi Tojo<sup>†3</sup>

**Abstract:** This paper describes an analyzer that automatically generates the metrical structure of a generative theory of tonal music (GTTM). Although a fully automatic time-span tree analyzer has been developed, musicologists have to correct the errors in the metrical structure. In light of this, we use a deep learning technique for generating the metrical structure of a GTTM. Because we only have 300 pieces of music with the metrical structure analyzed by musicologist, directly learning the relationship between the score and metrical structure is difficult due to the lack of training data. To solve this problem, we propose a multidimensional multitask learning analyzer called deepGTTM-II that can learn the relationship between score and metrical structures in the following three steps. First, we conduct unsupervised pre-training of a network using 15,000 pieces in a non-labeled dataset. After pre-training, the network involves supervised fine-tuning by back propagation from output to input layers using an automatic labeled dataset, which consists of 15,000 pieces labeled with an automatic analyzer that we previously constructed. Finally, the network involves supervised fine-tuning using a labeled dataset. The experimental results demonstrated that the deepGTTM-II outperformed the previous analyzers for a GTTM in F-measure for generating the metrical structure.

### 1. はじめに

本稿では、音楽理論 Generative Theory of Tonal Music (GTTM)[1]に基づき拍節構造を自動獲得する分析器を提案する。GTTM は、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の順で分析が進められる。二番目の分析である拍節構造の分析結果は、その後の分析に影響を与えるため、精度高く分析することが極めて重要である。GTTM の拍節構造の獲得を可能としていた従来の分析器には、ATTA[2]および FATTA[3]があったが、いずれも分析精度が低く、分析結果を音楽家が手動で修正する必要があった。

そこで本研究では、GTTM における局所的グルーピング境界の検出に、ディープラーニングを用いた deepGTTM-I を構築し、検出精度の向上を目指す。ディープラーニングは多層化したニューラルネットワークで、大量のデータを

用いてプレトレーニングと呼ばれる教師なし学習を繰り返し行うことで、ファインチューニングと呼ばれるラベル付きデータを用いた教師つき学習の効率を上げることを可能とする。本研究の最終的なゴールは、音楽家による分析結果をディープラーニングで学習させることで、人間と同じ分析結果を出力するネットワークを構築することである。

ディープラーニングを用いて音楽理論 GTTM に基づく分析器を構築するためには、以下の 3 つを考慮する必要がある。

- 多タスクの識別問題であること  
特徴ベクトルからその時のラベルを推定するのが、一般的な識別問題である。一方、たとえば拍が強拍であるか弱拍であるかは、すべての拍に発生しうる。そこで本研究では、ある拍が強拍（あるいは弱拍）であるかどうかを判定する問題を 1 つのタスクとらえる。すると、楽曲から拍節構造を求める問題は、マルチタスクの識別問題と捉えることができる。4.2 では、マルチタスクのディープラーニングを用いたモデル化について述べる。

- 大量の学習データが必要となること  
多層からなるニューラルネットワークを学習するためには

<sup>†1</sup> 京都大学

Kyoto University

<sup>†2</sup> 公立はこだて未来大学

Future University Hakodate

<sup>†3</sup> 北陸先端科学技術大学院大学

JAIST

大量のデータが必要である。ファインチューニングに用いる教師付データは、GTTM データベースに収められている300曲のラベル付きの曲をデータとして用いる[4]。一方、プレトレーニングに用いる教師なしデータは300曲では明らかに不足で、もっと多くの曲が必要である。教師なしデータはラベルが不要であるため、makemusic社のMusicXMLのページ[5]で紹介されているサイトからダウンロードした15,000曲のデータを用いることにした。4.1では、学習セットの作成について述べる。

#### ● GTTM のルールが適用されていること

GTTMは複数のルールによって構成されており、拍節構造の分析では、ルールが多く適用されている箇所が強拍になりやすい。GTTMデータベースに収められている楽曲では、音楽家による分析結果として、階層的な強拍の位置だけでなく、ルールの適用位置も収められている。これらルールの適用位置は、拍節構造を求めるための大きな手がかりとなる。4.3節では、マルチタスクディープラーニングの学習に、適用されたルールを利用することについて述べる。

マルチタスクディープラーニングを用いて、拍節構造の学習を行った結果、deepGTTM-IIが従来の分析器に比べて高い性能であることを確認した。以下本稿の構成は、2節で従来の分析システム・応用システムについて検討を行い、3節でGTTMの実装上の問題点について述べる。そして、4節でdeepGTTM-IIを提案し、5節でその評価実験を行う。最後に6節で、まとめと今後の課題について述べる。

## 2. 関連研究

これまで様々な音楽理論[6, 7, 8]が提案されてきたが、GTTMは他の音楽理論と比べて比較的厳密なルールで記述されており、音楽知識を形式化する上で最も有望であると我々は考え、これまで10年以上にわたり分析システム、応用システムの両面から研究を進めてきた(図1)[9]。

### (1) 分析システム

図1のタイムラインより上は、我々がこれまで構築してきた分析システム・手法である。我々は、2004年にグルーピング構造分析器および拍節構造分析器を構築し、それらを統合したタイムスパン木分析器ATTA (Automatic Time-span Tree Analyzer) [2]を2005年に構築した。ATTAは、46個の調節可能なパラメータを持ち、それらを適切に調整すればグルーピング構造では8割弱、拍節構造では9割強の正解率、タイムスパン木では、6割の正解率であった。ATTAはGTTMのルールの優先順序を適切に適用すれば、正解率の高いグルーピング構造や拍節構造を獲得できることを示した点で意義があるが、その一方で、パラメータを調整するためには、音楽的な知識が必要で、音楽家でなければ扱うことが困難であった。

2007年に構築したFATTA[3]は、タイムスパン木の安定性に関するルールに基づきタイムスパン木の安定性を定義し、

その安定性が高くなるようにATTAのパラメータを自動調整するシステムであった。拍節構造分析の正解率は9割程度であったが、グルーピング構造分析およびタイムスパン簡約の正解率は5割弱で、音楽家の手作業による修正を行わなければ分析結果を利用することは難しかった。

誤りのない完全な分析結果を出力する分析器を構築することが困難であったことから、2009年に構築したInteractive GTTM analyzerでは分析器と手動による編集をシームレスに行うことを可能にしたものであった[10]。Interactive GTTM analyzerは現在でも分析データの蓄積のために使用されており無料でダウンロードすることができる[4]。

2008年に構築した $\sigma$ GTTMでは、決定木を用いた統計的学習により自動で局所的グルーピング構造の検出を可能にしていた[11]。FATTAより性能が高かったものの、パラメータ調整後のATTAよりは低い性能であった。一方、 $\sigma$ GTTMIIでは、複数の学習済みの決定木を用意し、それらを手動で切り替えることで、ATTAを超える性能を実現していたが、適切なグルーピング境界を選択するためには、音楽知識が必要で、音楽家でなければ難しい作業であった[12]。

2015年に構築した $\sigma$ GTTMIII[13]では、音楽家による楽曲分析結果を確率文脈自由文法(Probabilistic context-free grammar, PCFG)[14]に基づき統計的に学習することでタイムスパン木の自動分析を可能としていた。 $\sigma$ GTTMIIIでは、GTTMデータベースに収められている300曲を教師データとしてPCFGの生成規則と生成確率を学習し、タイムスパン木分析器としては最も高い性能を示していた。また、同じくPCFGに基づくタイムスパン木分析器であるpGTTM[15]では、教師なし学習を可能としていた。これら、 $\sigma$ GTTMIIIおよびpGTTMの特長は、タイムスパン木の先端付近では拍節の影響が強く、根の近くでは和声の影響が強くなるなど、階層によるコンテキストの違いを学習可能としている点である。

上記述べてきた分析システム・手法では、楽曲中で並列的なメロディを見つけたり、シンメトリーな構造を見つけるような基本的な処理が不足していたり、あるいはそれらの処理があった場合でも、それらの処理と他の処理とがうまく連携されていないという問題があった。本研究では、それらの一連の処理をディープラーニングで実現することを目指す。

### (2) 応用システム

図1のタイムラインより下は、我々がこれまで構築してきた応用システムである。GTTMの分析の結果求まるタイムスパン木は、楽曲の要約[16]や表情付け[17]、生成[18]、メロディモーフィング[19]、作曲支援[20]に用いることができる。これらのシステムを有効に利用するためには、GTTMに基づく楽曲分析を精度高く自動で行うシステムの構築が必要である。

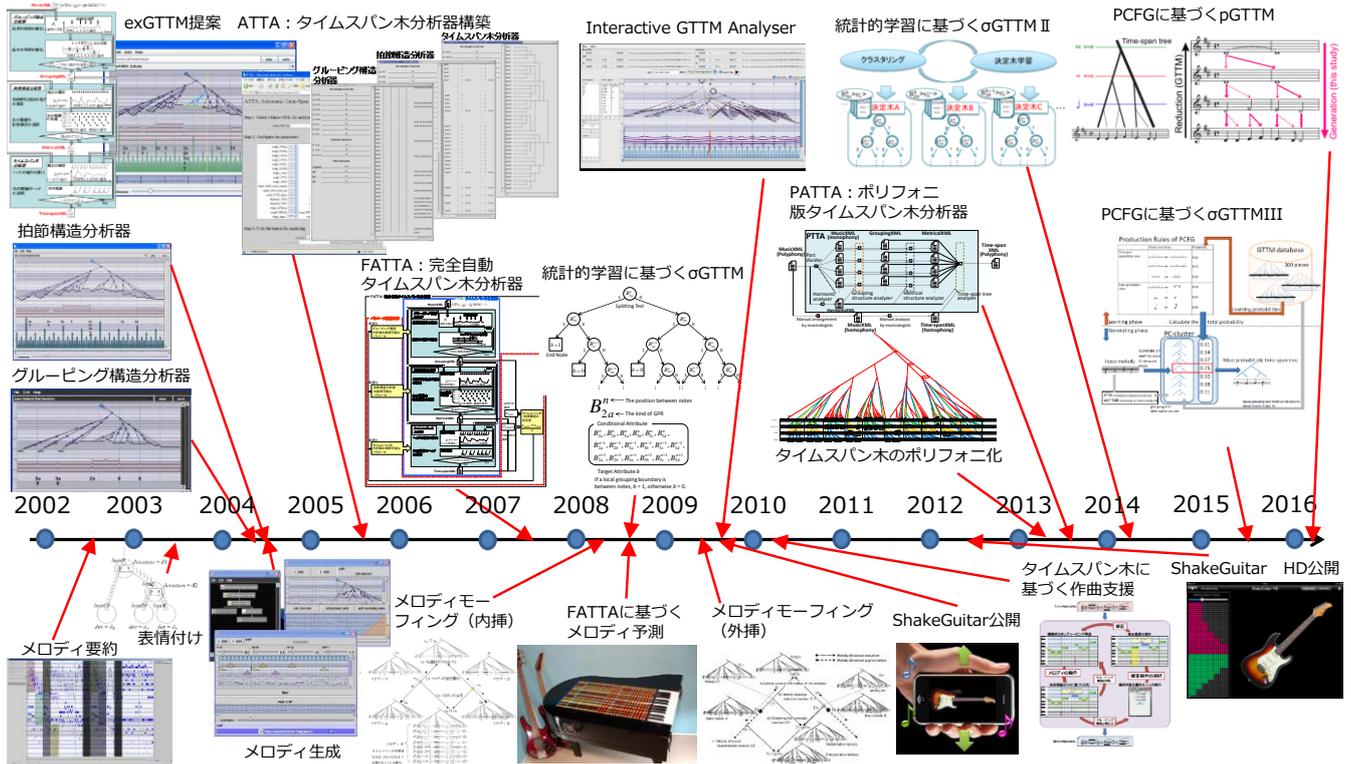


図 1 音楽理論 GTTM に基づく分析システム・応用システム  
 Figure 1 Related work on analysis and application systems for GTTM

(3) ビートトラッキング

拍節構造分析に似通った手法としては、ビートトラッキングが挙げられる。従来のビートトラッキングに関する手法 [21, 22, 23, 24] は、階層的な拍節構造の獲得を実現していたが、そこで扱われたのは小節レベルまでの拍節構造で、本研究で扱うような、それより高次の拍節構造については考慮していなかった。

3. GTTM と実装上の問題

図 2 は、グルーピング構造、拍節構造、タイムスパン木およびプロローゲーション木である。拍節構造分析は、4 分音符/2 分音符/1 小節/2 小節/4 小節などそれぞれの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る。

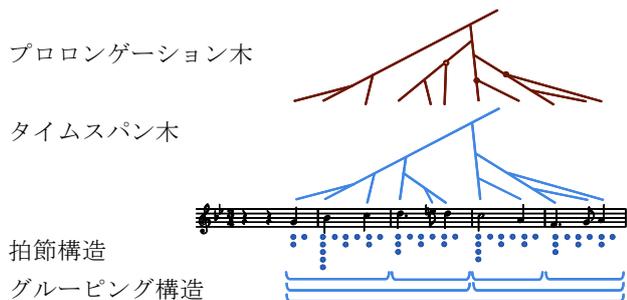


図 2 GTTM の分析結果  
 Figure 2 Analysis results by GTTM

3.1 拍節選好ルール

拍節構造分析は、構成ルールと選好ルールの 2 種類のルールにより定義されている。構成ルールを満たす構造が複数ある場合、選好ルールはそれら複数の構造の中から望ましいものを選び出すためのものである。拍節選好ルールは MPR1 (parallelism), MPR2 (strong beat early), MPR3 (event), MPR4 (stress), MPR5 (length), MPR6 (bass), MPR7 (cadence), MPR8 (suspension), MPR9 (time-span interaction), and MPR10 (binary regularity) の 10 種類からなる。MPR5 はさらに (a) pitch-event, (b) dynamics, (c) slur, (d) articulation, (e) repeated pitches, and (f) harmony の 6 種類に分類される。

3.2 選好ルールの競合

選好ルールの適用順や優先順位は定められていないため、選好ルールの競合はしばしば発生し、分析の曖昧性をもたらしている。図 3 は、MPR5c と 5a が競合している例である。MPR5c は長いスラーの開始点を強拍とするルールで、MPR5a は長い音符の開始点を強拍とするルールである。拍

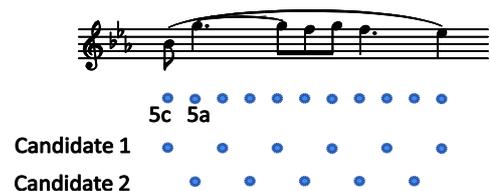


図 3 拍節選好ルールの競合の例  
 Figure 3 Example of conflict between MPRs

節構成ルールでは、強拍は2拍に1回、あるいは3拍に1回繰り返されると定義されていることから MPR5c と 5a の両方を成立させて第1音目と第二音目の両方を強拍とすることはできず、ルールの競合が生じる。

### 3.3 ルールの定義の曖昧さ

GTTM のルールには非常に定義が曖昧なものが存在する。たとえば、MPR5a (Length) は、どちらかといえば長い音のオンセットを強拍とするルールである。しかしこの、「どちらかといえば」という語は曖昧である。一方、MPR1 (Parallelism) は、並行しているフレーズの拍節構造が並行した構造になること選好する。しかし、GTTM では曲の類似度について定義されていないため、フレーズが並行しているかどうかの判断は分析者に委ねられている。

### 3.4 状況依存性

3.2 および 3.3 で述べた問題を解決するため、我々は GTTM を計算機実行可能に拡張した exGTTM およびそれを計算機上に実装した ATTA を提案した[2]。図 4 は、exGTTM および ATTA を用いて MPR4, 5a, 5b, 5c を適用した例である。調節可能なパラメータ  $T_j$  ( $j = 4, 5a, 5b, 5c$ ) を用いて閾値を変化させることで各ルールを成立させるかどうかを決めることができる。しかし、パラメータの適切な値は楽曲や曲の長さ、拍節構造の階層の深さなどに依存して変化するため、適切なパラメータを推定することは難しい。

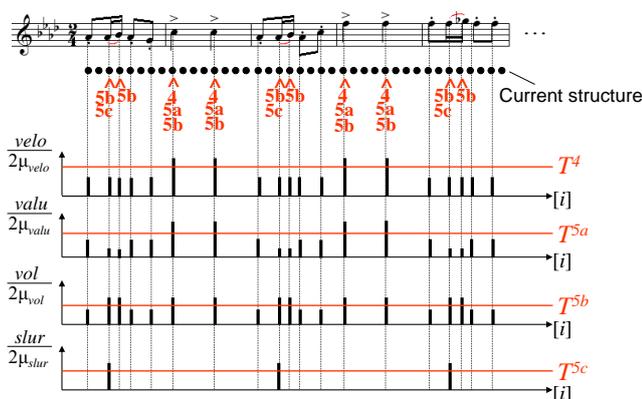


図 4 ATTA を用いた MPR4, 5a, 5b, 5c の適用

Figure 4 Application of MPR4, 5a, 5b, and 5c in ATTA

### 3.5 フィードバックリンクに関する説明の不足

GTTM では、タイムスパン木など高次の構造からグルーピング構造や拍節構造など低次の構造へフィードバックとして働くルールが複数存在する。たとえば、MPR9 (Time-span Interaction) は、タイムスパン簡約と矛盾しないような拍節構造を優先するルールである。しかし、いくつかの例が示されているだけで詳細な説明がなされていない。

また、明示的ではないがフィードバックリンクとなっている場合もある。たとえば、タイムスパン簡約の分析結果は、コード進行に強い関連があるし、コード進行に関連するルールも多くある。

したがって、ディープラーニングを用いて GTTM を完全に

実装するためにはリカレントニューラルネットワーク (RNN) など、フィードバックリンクを扱えるネットワークを用いることが望ましい。本稿では、拍節構造分析の実現に主眼を置いているためフィードバックリンクについては扱わない。

## 4. deepGTTM-II: ディープラーニングに基づく拍節構造分析器

3.2 から 3.4 までの問題を解決するため GTTM の分析にディープラーニングを導入する。

### ● ルールの適用の学習

従来の GTTM 分析器は、研究者・プログラマーによって構築されてきたが、その場合 3.3 で述べた GTTM における曖昧な点は、実装者のごとに解釈が異なってしまう可能性がある。deepGTTM-II の場合、楽譜と拍節選好ルールの適用箇所を学習して、各階層の一つ一つの拍それぞれに GTTM のルールが適用されるか否かを出力する深層のネットワークを構築する。したがって、性能は、学習データの質と量に依存することになる。

### ● ルールの優先度の学習

FATTA は、タイムスパン木の安定性という 1 つの尺度を用いてルールの優先順序を決定していたが、分析精度は低かった。おそらく、ルールの優先度は楽曲の様々なコンテキストに依存しており 1 つの尺度に集約することは困難なためと考えられる。deepGTTM-II で用いる深層ネットワークは、楽曲を入力しルールが適用されるか否かをネットワークの重みとバイアスとして学習するため、曲のコンテキストを反映した分析器を作成することが可能である。

本節では、ディープラーニングを用いて拍節構造の分析を可能にする方法について述べる。

### 4.1 学習・評価データ

ネットワークの学習のため、教師なしデータ、自動教師付けデータ、教師付きデータの 3 種類を用意した (図 5)。

#### (1) 教師なしデータ

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)。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。

#### (4) 自動教師付けデータ

GTTM データベースには 300 曲の MusicXML と、音楽家による拍節構造の分析結果および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、

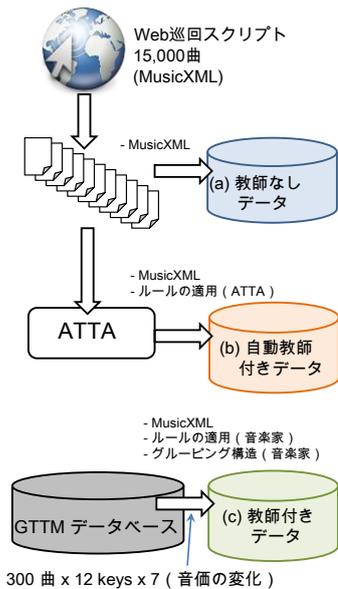


図 5 教師なし・自動教師付け・教師付きデータ

Figure 5 Non-labeled, automatic labeled, and labeled datasets

自動教師付けデータを作成する (図 5b).

自動教師付けデータは, ATTA を用いて 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.

各拍節選好ルールによる拍の強さは  $D_i^j$  ( $j=2, 3, 4, 5a, 5b, 5c, 5d, 0 < D_i^j < 1$ ) で表される. たとえば, MPR4 (Event) は, 強調されて演奏された音符を強拍とするルールであるが, ATTA では次式のように定式化している.

$$D_i^4 = \begin{cases} 1 & \text{velo}_i > 2 \times \mu_{\text{velo}} \times T_j^4 \\ 0 & \text{else} \end{cases} \quad (1)$$

ここで  $\text{velo}_i$  は音符のベロシティ,  $\mu_{\text{velo}}$  はその平均,  $T_j$  ( $0 < T_j < 1$ ) は, ルールを成立させるか ( $D_i^j = 1$ ), 否か ( $D_i^j = 0$ ) を決めるパラメータである. deepGTTM-II では,  $T_j$  ( $j=2, 3, 4, 5a, 5c, 5d$ ) の値は 1 とする.

## (2) 教師付きデータ

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拍節構造分析データおよびグルーピング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トした曲を用意した. 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 (図 5c).

## 4.2 Deep Belief Network

拍節構造を分析する深層ネットワークとして deep belief network (DBN) を採用した (図 6). DBN の入力, 楽譜

(MusicXML) から取得した各音符のオンセット, オフセット, ベロシティ, ピッチおよび, グルーピング構造 (GroupingXML) から取得したグルーピング境界である. グルーピング境界は階層ごとに境界では 1 それ以外では 0 が入力される.

DBN の出力は, 1 つの階層につき, 7 種類の拍節選好ルール (MPR2, 3, 4, 5a, 5b, 5c, 5d) とその階層の拍節構造を接続し, マルチタスク学習を構成する. 拍節選好ルールは, ルールが適用される場合 1, されない場合 0, 拍節構造は強拍の場合 1, 弱拍の場合 0 とする. 階層的な拍節構造は, 上位の階層のネットワークを次々と接続していくことで構成される. 出力とネットワークの最終段の隠れ層との接続はロジスティック回帰を用いた. すべての出力は, 第 1 層から第  $n$  層までのすべての隠れ層を共有し, 以下に述べる 4 ステップで学習を行う. 学習する曲順は Epoch ごとにランダムに変更する.

### (1) 隠れ層 1 から $n$ 層までのプレトレーニング

入力に接続した隠れ層 1 から出力に接続した  $n$  層まで制約付きボルツマンマシン (RBM) を用いて, プレトレーニング (教師なし学習) を行う. プレトレーニングは, 15,000 曲の教師なしデータを用いて 100Epoch 繰り返す.

### (2) MPR2, 3, 4, 5a, 5b, 5c, 5d のファインチューニング

プレトレーニングが終了しているネットワークに対し, 15,000 曲の自動教師付けデータを用いバックプロパゲーションにより MPR2, 3, 4, 5a, 5b, 5c, 5d を 100Epoch ずつ教師付き学習する.

### (3) 1 つのレベルの拍節構造のファインチューニング

MPR のファインチューニングが済んでいるネットワークを用いて, 25,200 曲の教師付きデータを用いてバックプロパゲーションで 1 つのレベルの拍節構造を教師付き学習する.

### (4) 次のレベルの教師なし/付き学習の繰り返し

次のレベルの拍節構造がある場合 (現在のレベルに拍点が 2 個以上ある場合) 隠れ層を 1 層追加し, その層を教師なしデータを用いてプレトレーニングした後, (2) と (3) を行う. そして, 拍点が 1 つになるまで (4) を繰り返す.

## 4.3 マルチタスク学習

4.2 で導入した DBN は非常に複雑で, 1 つのレベルの拍節構造の学習がマルチタスク学習になっている. したがって, 複数のルールの学習は, 多次元のマルチタスク学習に相当する. マルチタスク学習では, ある一つの拍が強拍であるか弱拍であるか, あるいは, ある一つの拍に拍節選好ルールが成立するか否かはそれぞれ別のタスクとして学習される. 学習の手順は以下の通りである. まず, 学習データの曲順をランダムシャッフルし, 1 曲ずつ順に選択する. 選択された曲の学習タスクの順をランダムシャッフルし, 1 タスクごとに学習を行う. 学習は, 強拍やルールが成立する場合には出力を 1, そうでない場合は 0 として, バック

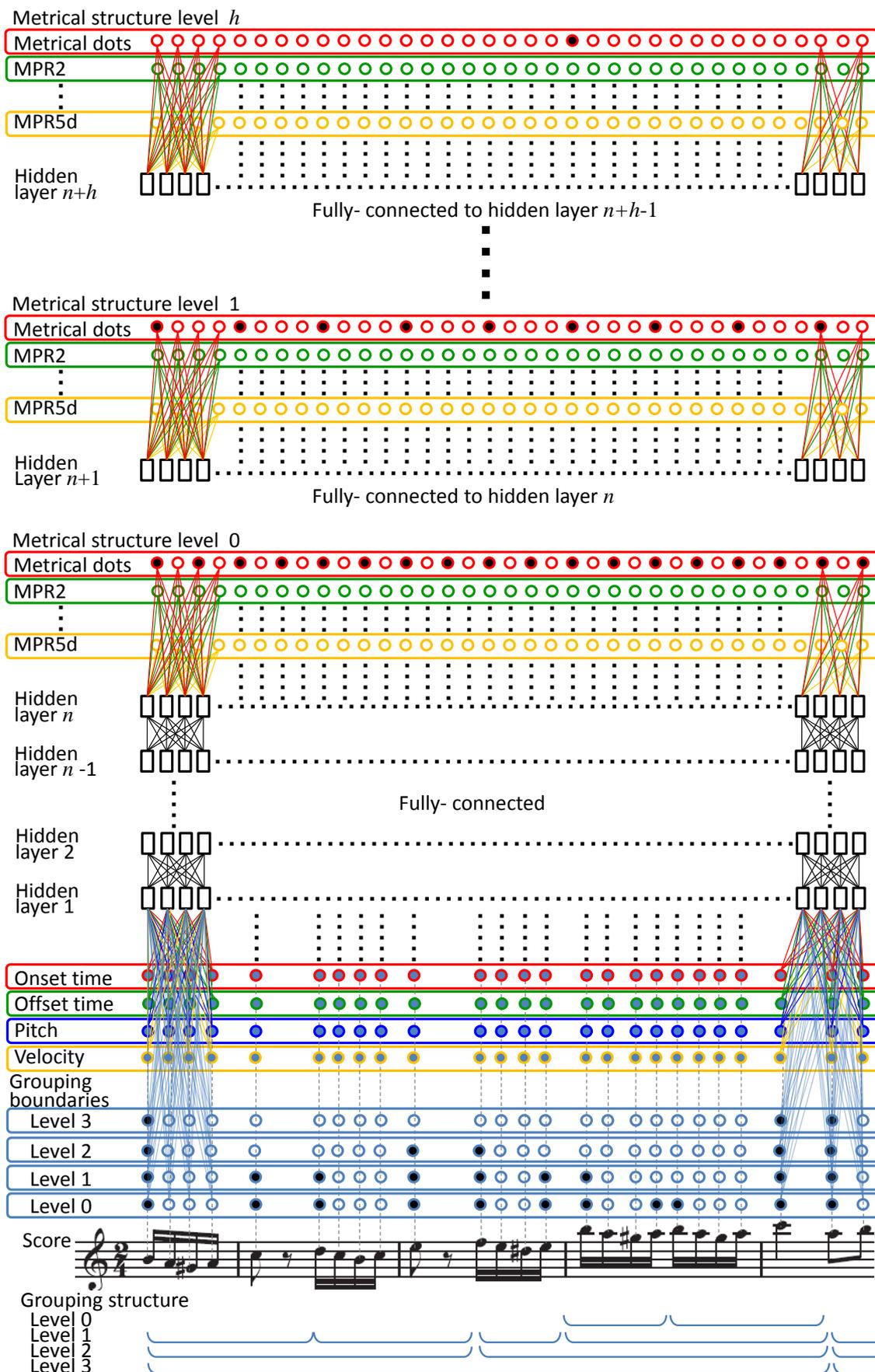


図 6 拍節構造を分析する Deep Belief Network

Figure 6 Deep belief network for generating metrical structure

プロパゲーションで行われる。1つのタスクの学習が100Epoch終了したら、次のタスクあるいは次の曲の学習を行う。

## 5. 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。

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

ただし、

適合率 P: 正解データの拍点と同じ拍点が、システムの入力に含まれている割合。

再現率 R: システムが出力した拍点が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結果である。ATTA および FATTA は、deepGTTM-II で評価に用いたのと同じ100曲での評価結果である。ATTA は、手動調整可能なパラメータがあり、その調整により性能が変化するため、パラメータの調節前後の F 値の両方を載せた。FATTA には調節可能なパラメータはない。その結果、deepGTTM-II は、FATTA およびパラメータ調整前後両方の ATTA の結果よりも性能が高いことが確認できた。

## 6. おわりに

本稿では、ディープラーニングに基づく拍節構造分析器 deepGTTM-II の構築について述べた。本研究の主な意義は次の3点である。

- ディープラーニングに基づく音楽分析器の構築  
ディープラーニングは様々なタスクにおいて強力であることが明らかになっているが、我々は音楽分析においても強力であることを明らかにした。今後、様々な音楽理論をディープラーニングを用いて実装することを試みていく。我々が構築した GTTM 分析データベースには300曲の曲と音楽家による分析結果が収められているが、300曲では深層のニューラルネットワークを学習するためには充分ではなかった。そこで、これまで我々が構築した GTTM 分析器である ATTA を用いて自動教師つきデータを作成することで、ネットワークの学習を可能とした。
- 手動調整なしの高性能な GTTM 分析器の実現  
ATTA や  $\sigma$  GTTM など従来の GTTM 分析器は、手動による調整なしでは、分析精度が非常に低かった。一方、手動による調整が不要な FATTA,  $\sigma$  GTTM,  $\sigma$  GTTMIII や pGTTM などの自動分析器は、性能が低かった (F 値で 0.8 以下)。これらに対し、deepGTTM-II は格段に高い性能を示し、GTTM の分析結果を利用するアプリケーションを実用化で

きる可能性が高まったと言える。今後、GTTM の分析全体をディープラーニングで実装してゆく。

- ルールの適用や強拍/弱拍のマルチタスク学習

GTTM で定義されている複数のルールを深層のニューラルネットワークでマルチタスク学習することが、拍節構造の学習を助けることが明らかになった。学習された各層3000ユニット11層のネットワークは、これまでに構築されたどの GTTM 分析器よりも性能が高いことが確認された。

本研究は、ディープラーニングを用いて GTTM を計算機上に完全実装するための第一歩である。今後、タイムスパン簡約とプロロンゲーション簡約の実装を行っていく。その際、以下の2つの問題が発生する。一つ目の問題は、タイムスパン木やプロロンゲーション木という木構造の分析・生成は、拍節構造に比べて格段に難易度が高くなることである。もう一つの問題は、学習データの不足である。木構造の組み合わせの数は、拍節構造に比べて多大になるため、多くの学習データを用意しなくては学習データがスパースになり学習が難しくなることが予想される。

今後、これらの問題を解決して GTTM のディープラーニング実装を進めていく。現在のところ、我々はなぜディープラーニングが拍節構造分析で高い性能を示しているのか充分検討できていないが、今後、学習済みネットワークを解析し知見を得ていく。

**謝辞** 本研究の一部は JSPS 科研費 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] Hamanaka, M., Hirata, K., Tojo, S.: Implementing 'a generative theory of tonal music', *Journal of New Music Research*, 35(4), 249-277, 2006.
- [3] Hamanaka, M., Hirata, K., Tojo, S.: Fatta: Full automatic time-span tree analyzer, *In: Proceedings of the 2007 International Computer Music Conference (ICMC2007)*, pp. 153-156, 2007.
- [4] Hamanaka, M., Hirata, K., 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.
- [5] Make music Inc. : Music in MusicXML Format, url: <http://www.musicxml.com/music-in-musicxml/>, 参照 2016-6-16.
- [6] Cooper, G. and Meyer, L. B. *The Rhythmic Structure of Music*. The University of Chicago Press, 1960.
- [7] Narmour, E. *The Analysis and Cognition of Basic Melodic Structure*. The University of Chicago Press, 1990.
- [8] Temperley, D. *The Cognition of Basic Musical Structures*. MIT press, Cambridge, 2001.
- [9] Masatoshi Hamanaka, Keiji Hirata, Satoshi Tojo: Implementing Methods for Analysing Music Based on Lerdahl and Jackendoff's Generative Theory of Tonal Music, *Computational Music Analysis* (pp. 221-249), Springer, 2016.
- [10] Hamanaka, M., Hirata, K., Tojo, S.: Interactive GTTM Analyzer,

- In: Proceedings of the 10th International Conference on Music Information Retrieval Conference (ISMIR2009)*, pp.291-296, 2009.
- [11] Miura, Y., Hamanaka, M., Hirata, K., Tojo, S.: Decision tree to detect gttm group boundaries. In: *Proceedings of the 2009 International Computer Music Conference (ICMC2009)*, pp. 125–128, 2009.
- [12] 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.
- [13] Hamanaka, M., Hirata, K., Tojo, S.:  $\sigma$  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.
- [14] Charniak, E.: Tree-bank grammars. In: *Proceeding of Association for the Advancement of Artificial Intelligence (AAAI-96)*, pp. 1032–1036, 1996.
- [15] Nakamura E., Hamanaka M., Hirata K., and Yoshii K.: Tree-Structured Probabilistic Model of Monophonic Written Music Based on the Generative Theory of Tonal Music, In: *proceedings of 41st IEEE International Conference on Acousitcs, Speech and Signal Processing (ICASSP2016)*, 2016.
- [16] 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.
- [17] 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.
- [18] 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.
- [19] Hamanaka, M., Hirata, K., Tojo, S.: Melody morphing method based on gttm. *In: Proceedings of the 2008 International Computer Music Conference (ICMC2008)*, pp. 155–158, 2008.
- [20] 西田 智, 浜中 雅俊, 平田 圭二, 東条 敏: 類似した楽曲構造を持った旋律のインタラクティブな生成方式, 情報処理学会音楽情報科学研究会, 2010-MUS-84(4), pp. 1-6, 2010.
- [21] D. Rosenthal, Emulation of human rhythm perception, *Computer Music Journal*, vol. 16, no. 1, pp. 64–76, 1992.
- [22] M. Goto, An audio-based real-time beat tracking system for music with or without drum-sounds, *Journal of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.
- [23] S. Dixon, Automatic extraction of tempo and beat from expressive performance, *Journal of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.
- [24] M. Davies and S. Bock, Evaluating the evaluation measures for beat tracking, in *Proceedings of the 2014 International Society for Music Information Retrieval Conference (ISMIR2014)*, 2014, pp. 637–642.