A VOCABULARY-FREE INFINITY-GRAM MODEL FOR NONPARAMETRIC BAYESIAN CHORD PROGRESSION ANALYSIS

Kazuyoshi Yoshii **Masataka Goto**
National Institute of Advanced Industrial Science and Technology (AIST), Japan
{k.yoshii, m.goto}@aist.go.jp

ABSTRACT

This paper presents probabilistic n-gram models for symbolic chord sequences. To overcome the fundamental limitations in conventional models—that the model optimality is not guaranteed, that the value of n is fixed uniquely, and that a vocabulary of chord types (e.g., major, minor, $\cdots$) is defined in an arbitrary way—we propose a vocabulary-free infinity-gram model based on Bayesian nonparametrics. It accepts any combinations of notes as chord types and allows each chord appearing in a sequence to have an unbounded and variable-length context. All possibilities of n are taken into account when calculating the predictive probability of a next chord given a particular context, and when an unseen chord type emerges we can avoid out-of-vocabulary error by adaptively evaluating the 0-gram probability, i.e., the combinatorial probability of note components. Our experiments using Beatles songs showed that the predictive performance of the proposed model is better than that of the state-of-the-art models and that we could find stochastically-coherent chord patterns by sorting variable-length n-grams in a line according to their generative probabilities.

1. INTRODUCTION

Chord progression analysis is an important task for content-based music information retrieval (MIR) [1, 2]. Because the chord patterns used in musical pieces are closely related to the composer styles [3] and musical genres [4], it is useful to build statistical models of chord patterns from symbolic chord sequences. In addition, accurate models of chord sequences (called *language models* in analogy with automatic speech recognition) could improve the accuracy of automatic chord recognition for music audio signals [5, 6].

So far, n-gram models have often been used as language models of chord sequences [2–6]. An n-gram is a subsequence of n chords in a given chord sequence, and n-gram models are based on $(n-1)$-order Markovian assumption because chords exhibit strong short-term dependency. In other words, each chord in a given sequence is assumed to depend on its $n-1$ previous chords called a *context*. Using a limited amount of observed data, the goal is to make a statistical model that can calculate the predictive probability of a next chord (n-gram probability), given any context of length $n-1$. However, the observed n-grams are generally a limited subset of all kinds of n-grams, and the number of all kinds of n-grams increases exponentially with increasing n. Therefore, the naive estimates of the probabilities of unobserved n-grams are zero. To avoid such overfitting, various heuristic smoothing methods have been developed [7].

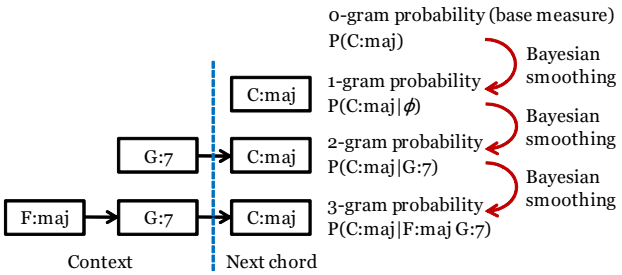

Figure 1. A hierarchical nonparametric Bayesian model for accurately smoothing n-gram probabilities.

In this paper we focus on three fundamental limitations of conventional n-gram models: 1) n-gram models based on heuristic smoothing methods have no solid theoretical foundation, 2) the value of n should be specified uniquely in advance even though each chord depends on a variable-length context, 3) A limited set of chord labels (e.g., major, minor, augmented, diminished, seventh, $\cdots$, and their derivations) should be defined as a vocabulary in advance. Especially, the last limitation has not been discussed so far.

To overcome these limitations, we propose a vocabulary-free infinity-gram model by extending modern nonparametric Bayesian n-gram models [8–10]. Our model is formulated in a hierarchical Bayesian manner (Figure 1) and has the following merits: 1) The predictive distribution of a next chord can be naturally formalized by providing the probabilistic generative model of chord sequences. 2) Each chord in a sequence is allowed to have an unbounded and variable-length context. A posterior distribution of the context length

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.
© 2011 International Society for Music Information Retrieval.

can be estimated. 3) Any combinations of notes can be accepted as chord types. A chord vocabulary is incrementally expanded as needed. These metits enable our model to not only attain the best performance but also find "stochastically-coherent" variable-length chord patterns that are not always simply the ones used most frequently (cf. [11]).

The innovative models of symbolic chord sequences (an infinity-gram model and its vocabulary-free extension) are useful for probabilistic modeling of music audio signals. A typical application is automatic chord recognition, where a vocabulary of chord labels is given. For example, an infinity-gram model could be fused with a joint probabilistic model of keys, chords, and bass notes [12]. Another novel application is automatic music transcription, where a vocabulary is *not* given. We plan to use a vocabulary-free model as a prior distribution on a probabilistic acoustic model for multipitch estimation [13], and jointly optimize the both models. This means that chords and their progressions (now "chords" are combinations of notes, not text labels) are self-organized in an unsupervised manner and are used as a constraint on simultaneous and temporal pitch distributions.

The rest of this paper is organized as follows: Section 2 describes the chord notations used in this study. Section 3 introduces related work on nonparametric Bayesian n-gram models and Section 4 explains our model. Section 5 reports our experiments and Section 6 concludes this paper.

2. CHORD NOTATIONS

We introduce label-based and component-based notations to represent chord sequences (Table 1).

2.1 Label-based Notation

The conventional label-based notation is based on intuitive shorthand labels defined by Harte *et al.* [14]. There are 17 chord labels with an attached root note, which is one of 12 pitch classes.[1] In this paper we do not distinguish C# from Db because they are in the same pitch class. This is a standard treatment used in [2, 3]. For example, C major and Gb diminished seventh chords are respectively represented as C:maj and F#:dim7. The symbol "N" is used to indicate "no chord" (e.g., silence or untuned sounds). The resulting vocabulary size is 205 ($17 \times 12 + 1$).

2.2 Component-based Notation

The component-based notation is based on degrees of note components (relative displacements against a root note). Each chord is represented as a combination of a root note and a 12-dimensional binary vector whose elements indicate the existences of the corresponding degrees. For example, C major chords are written as C:100010010000 and D major chords as D:100010010000, not as D:001000100100. Note that any combinations of notes can be represented even if they are not defined in Table 1. For example, C major chords with an added fourth are written as C:100011010000. Such information is available in Harte's chord annotations [14]. With the additional symbol "N", the resulting vocabulary size is 49153 ($2^{12} \times 12 + 1$). This is finite because we focus on *note existences* in individual pitch classes. Note that a truly vocabulary-free (infinite-vocabulary) notation can be defined by focusing on *note counts* based on musical scores, i.e., by representing note components of each chord as a 12-dimensional nonnegative-integer vector.

[1] The pitch classes are defined as 12 different scales within an octave, i.e., {C, C#, D, D#, E, F, F#, G, G#, A, A#, B}.

Chord type	Label	Components
Major	maj	100010010000
Minor	min	100100010000
Diminished	dim	100100100000
Augmented	aug	100010001000
Major Seventh	maj7	100010010001
Minor Seventh	min7	100100010010
Seventh	7	100010010010
Dim. Seventh	dim7	100100100100
Half Dim. Seventh	hdim7	100100100010
Min. (Maj. Seventh)	minmaj7	100100010001
Major Sixth	maj6	100010010100
Minor Sixth	min6	100100010100
Ninth	9	101010010010
Major Ninth	maj9	101010010001
Minor Ninth	min9	101100010010
Suspended Second	sus2	101000010000
Suspended Fourth	sus4	100001010000

Table 1. Shorthand labels and pitch-class components

3. PROBABILISTIC LANGUAGE MODELS

This section introduces related work on n-gram models. We first identify the purpose of n-gram modeling and then explain several state-of-the-art models based on the probability theory of Bayesian nonparametrics.

3.1 Problem Specification

Suppose we have a chord vocabulary $\boldsymbol{W}$ whose size is V (in this paper, 205 or 49153). Let $w \in W$ be a chord and $\boldsymbol{u} \in W^{n-1}$, where n can be *any* positive integer, be a context consisting of a sequence of $n-1$ chords. We have a limited amount of observed data $\boldsymbol{X}$, which is a sequence of M chords, $x_1 x_2 \cdots x_M$, where $x_m \in W$ $(1 \leq m \leq M)$. We assume for simplicity that we have only one chord sequence. In n-gram modeling, each chord x_m is assumed to depend on the past $n-1$ chords (context).

Given observed data $\boldsymbol{X}$, the goal is to estimate $P_{\boldsymbol{u}}(w|\boldsymbol{X})$, i.e., the predictive probability of chord w following context $\boldsymbol{u}$. Let $c_{\boldsymbol{u}w}$ be the number of occurrences of chord w following context $\boldsymbol{u}$ in training data $\boldsymbol{X}$. The naive maximum likelihood (ML) estimate is given by

$$P^{\mathrm{ML}}_{\boldsymbol{u}}(w|\boldsymbol{X}) = \frac{c_{\boldsymbol{u}w}}{c_{\boldsymbol{u}\cdot}} \tag{1}$$

where the dot $(\cdot)$ means the sum over that index, i.e., $c_{\boldsymbol{u}\cdot} = \sum_{w'} c_{\boldsymbol{u}w'}$. However, if n-gram $\boldsymbol{u}w$ is not observed in $\boldsymbol{X}$

($c_{\boldsymbol{u}w} = 0$), its probability is estimated to be zero. This is called the zero-probability problem.

To solve this problem various smoothing methods have been proposed. The family of Kneser-Ney (KN) smoothing is empirically known as one of the most accurate smoothing techniques [7]. A method called interpolated KN (IKN) estimates $P_{\boldsymbol{u}}(w|\boldsymbol{X})$ by discounting the actual count $c_{\boldsymbol{u}w}$ by a fixed amount $d_{|\boldsymbol{u}|}$ depending on the context length $|\boldsymbol{u}|$ if $c_{\boldsymbol{u}w} > 0$ (otherwise the count remains 0). Furthermore, the discounted n-gram probability of chord w is interpolated with the $(n-1)$-gram probability of chord w. Another important variant is called modified KN (MKN), where the amount of discount is allowed to vary according to the value of $c_{\boldsymbol{u}w}$. MKN is known to slightly outperform IKN.

3.2 Hierarchical Pitman-Yor Language Model

Teh [8] proposed a nonparametric Bayesian n-gram model called a hierarchical Pitman-Yor language model (HPYLM). Interestingly, IKN was proven to be a deterministic approximation of the HPYLM, which can be optimized in a principled way and performs better than IKN.

3.2.1 Pitman-Yor Process and Hierarchical Formulation

We briefly explain the Pitman-Yor process (PY) [15], which is a building block of nonparametric Bayesian models. The PY is a distribution over distributions (e.g., n-gram distributions) over a sample space (e.g., vocabulary $\boldsymbol{W}$). Let d and θ be positive real numbers and G_0 be a distribution over a sample space. The PY is written as

$$G \sim \mathrm{PY}(d, \theta, G_0) \tag{2}$$

where d is called a discount parameter, θ a strength parameter, and G_0 a base measure. G is a random distribution over the sample space. When the value of θ becomes larger, G is more likely to be similar to G_0.

The HPYLM is formulated by layering PYs in a hierarchical Bayesian manner. Suppose we have a unigram distribution G_ϕ over $\boldsymbol{W}$, where ϕ is the empty context and $G_\phi(w)$ is the unigram probability of chord w. A bigram distribution G_u given the last chord u differs from but is somewhat similar to G_ϕ. Here G_u is assumed to be drawn from a PY with base measure G_ϕ as $G_u \sim \mathrm{PY}(d_1, \theta_1, G_\phi)$, where d_1 and θ_1 are discount and strength parameters that are shared among contexts of length 1. Generally speaking, an n-gram distribution $G_{\boldsymbol{u}}$ given a context $\boldsymbol{u}$ of length $n-1$ is drawn from a PY with base measure $G_{\pi(\boldsymbol{u})}$ as follows:

$$G_{\boldsymbol{u}} \sim \mathrm{PY}(d_{|\boldsymbol{u}|}, \theta_{|\boldsymbol{u}|}, G_{\pi(\boldsymbol{u})}) \tag{3}$$

where $\pi(\boldsymbol{u})$ is a shortened context obtained by removing the earliest chord from $\boldsymbol{u}$, and $d_{|\boldsymbol{u}|}$ and $\theta_{|\boldsymbol{u}|}$ are discount and strength parameters depending on the length $|\boldsymbol{u}|$. Since the $(n-1)$-gram distribution $G_{\pi(\boldsymbol{u})}$ is unknown, a PY prior with parameters $d_{|\pi(\boldsymbol{u})|}$ and $\theta_{|\pi(\boldsymbol{u})|}$ and base measure $G_{\pi(\pi(\boldsymbol{u}))}$ is recursively put on $G_{\pi(\boldsymbol{u})}$. Finally, the unigram distribution G_ϕ is given by $G_\phi \sim \mathrm{PY}(d_0, \theta_0, G_0)$ where G_0 is a global base measure (0-gram distribution), which is usually assumed to be uniform, i.e., $G_0(w) = 1/V$.

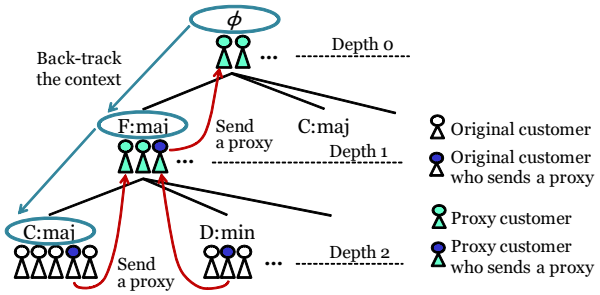

Figure 2. Hierarchical Pitman-Yor language model.

Consequently, the hierarchical structure of the HPYLM can be represented as a suffix tree of depth $n-1$, as shown in Figure 2 where the case of $n=3$ is illustrated. Each node is identified as a context, i.e., descending the tree from the root node to the target node means back-tracking the context.

3.2.2 Stochastic Process for Data Generation

Once the HPYLM is defined, observed data $\boldsymbol{X}$ is generated according to a stochastic process called the Chinese restaurant franchise (CRF), which can be explained by using a metaphor in which contexts are likened to *restaurants*, M observed variables in $\boldsymbol{X}$ are likened to *customers*, and V chord types in $\boldsymbol{W}$ are likened to *dishes*. Each restaurant is allowed to have an unbounded number of *tables* and each table is served a dish. Each customer enters a restaurant, sits at a table, and eats a dish served at that table.

We suppose that $x_1, \cdots, x_M$ are generated sequentially, and consider how the m-th customer x_m behaves, given a seating arrangement of the past customers $\{x_1, \cdots, x_{m-1}\}$. The customer x_m enters restaurant $\boldsymbol{u} = x_{m-(n-1)} \cdots x_{m-1}$ of depth $n-1$. Let $t_{\boldsymbol{u}w}$ be the number of tables serving dish w in restaurant $\boldsymbol{u}$. There are $t_{\boldsymbol{u}\cdot}$ tables in total. Let $c_{\boldsymbol{u}wk}$ be the number of customers sitting at table k and eating dish w ($c_{\boldsymbol{u}wk} = 0$ if table k does not serve dish w). The customer x_m then sits (i) at an existing table k ($1 \le k \le t_{\boldsymbol{u}\cdot}$) and eats a dish w served at the table with probability proportional to $c_{\boldsymbol{u}wk} - d_{|\boldsymbol{u}|}$ or (ii) at a new table $k = t_{\boldsymbol{u}\cdot} + 1$ with probability proportional to $d_{|\boldsymbol{u}|} t_{\boldsymbol{u}\cdot} + \theta_{|\boldsymbol{u}|}$. In the case (i), the value of x_m is set to w and $c_{\boldsymbol{u}wk}$ is incremented. In the case (ii), to order a dish served at the new table k, a proxy customer is sent to the parent restaurant $\pi(\boldsymbol{u})$, where he behaves in a recursive manner. If he eventually eats a dish w in restaurant $\pi(\boldsymbol{u})$, the dish w is also served at the new table k in restaurant $\boldsymbol{u}$ and the customer x_m eats the dish w. Consequently, $t_{\boldsymbol{u}w}$ is incremented, the value of x_m is set to w, and $c_{\boldsymbol{u}wk}$ is incremented. Note that when the proxy customer sits at a new table in restaurant $\pi(\boldsymbol{u})$, a new proxy customer is further sent to the restaurant $\pi(\pi(\boldsymbol{u}))$. Finally, a proxy customer may be sent to the root restaurant ϕ. When he sits at a new table in the root restaurant ϕ, a dish served at the new table is chosen according to the global base measure G_0.

More specifically, given a particular seating arrangement (denoted by $\boldsymbol{S}$), a next chord w following context $\boldsymbol{u}$ is generated according to the following predictive distribution:

$$P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{S}) = \frac{c_{\boldsymbol{u}w\cdot} - d_{|\boldsymbol{u}|}t_{\boldsymbol{u}w}}{c_{\boldsymbol{u}\cdot\cdot} + \theta_{|\boldsymbol{u}|}} + \frac{d_{|\boldsymbol{u}|}t_{\boldsymbol{u}\cdot} + \theta_{|\boldsymbol{u}|}}{c_{\boldsymbol{u}\cdot\cdot} + \theta_{|\boldsymbol{u}|}} P_{\pi(\boldsymbol{u})}^{\text{HPY}}(w|\boldsymbol{S}) \quad (4)$$

where Eqn. (4) is a recursive definition with respect to context $\boldsymbol{u}$ of any length, e.g., $P_{\pi(\boldsymbol{u})}^{\text{HPY}}(w|\boldsymbol{S})$ is given by substituting $\pi(\boldsymbol{u})$ into $\boldsymbol{u}$ in Eqn. (4). Starting with an empty tree ($c_{\boldsymbol{u}wk} = 0$ and $t_{\boldsymbol{u}w} = 0$), a seating arrangement for $\boldsymbol{X}$ is obtained by adding M customers one by one. The IKN was found to be an approximation of the HPYLM (the HPYLM reduces to the IKN when $\theta_{|\boldsymbol{u}|} = 0$ and $t_{\boldsymbol{u}w} = 1$).

3.2.3 Predictive Distribution and Bayesian Inference

The goal is to estimate the predictive distribution $P_{\boldsymbol{u}}(w|\boldsymbol{X})$ in a Bayesian manner. Since a true seating arrangement for $\boldsymbol{X}$ is unknown, the expected value of $P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{S})$ is calculated under the CRF $P(\boldsymbol{S}|\boldsymbol{X})$ as follows:

$$P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{X}) = \sum_{\boldsymbol{S}} P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{S}) P(\boldsymbol{S}|\boldsymbol{X}) \quad (5)$$

Because this sum is analytically intractable, Gibbs sampling is used for approximation. More specifically, we get

$$P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{X}) \approx \frac{1}{L}\sum_{l=1}^{L} P_{\boldsymbol{u}}^{\text{HPY}}(w|\boldsymbol{S}_l) \quad (6)$$

where L is the number of many i.i.d. seating arrangements sampled from $p(\boldsymbol{S}|\boldsymbol{X})$ and l is a sample index.

The Gibbs sampling algorithm is shown in Figure 3. First, a seating arrangement is initialized by adding all customers one by one according to the *posterior* CRF, where each customer $x_m = w$ sits at an existing or new table serving dish w with probability given by the first or second term of Eqn. (4). Then a customer x_m is selected randomly and removed from the tree, from which the related proxy customers and tables that become empty are also removed. Given a seating arrangement of the other customers, the customer x_m is added to the tree again according to the posterior CRF. By iterating this operation, L seating arrangements are sampled with a certain interval. Since the parameters $d_0, \cdots, d_{n-1}$ and $\theta_0, \cdots, \theta_{n-1}$ are unknown, beta and gamma prior distributions are put on them and the values of the parameters are sampled from posterior distributions (see details in [8]).

3.3 Variable-Order Pitman-Yor Language Model

A problem of the HPYLM is that all M customers are forced to enter restaurants of fixed depth $n-1$. To solve the problem, Mochihashi and Sumita [9] proposed a variable-order PY language model (VPYLM) that allows each customer to enter a restaurant of variable depth. Each chord x_m is associated with a latent variable z_m that indicates the value of n (depth$+1$). Since a true value of z_m is unknown, all possible values of z_m are considered (n is marginalized out) for making predictions, resulting in the *infinity*-gram model.

Create an empty tree
for $m = 1 : M$ in random order
  Add customer x_m to the tree at depth n-1
for $i = 1 : \infty$
  for $m = 1 : M$ in random order
    Remove customer x_m from the tree
    Add customer x_m to the tree at depth n-1

Figure 3. Gibbs sampling algorithm for HPYLM.

3.3.1 Stochastic Process for Data Generation

We consider how the value of n-gram length z_m is stochastically determined. The customer x_m descends the tree by following a path $\phi \to x_{m-1} \to x_{m-2} \to \cdots$, i.e., by backtracking the context $\boldsymbol{u}$. When he arrives at restaurant $\boldsymbol{u}_i$ of depth i ($0 \le i \le \infty$), he stops there with probability $\eta_{\boldsymbol{u}_i}$ or passes through with probability $1 - \eta_{\boldsymbol{u}_i}$. The probability of $z_m = n$ ($1 \le n \le \infty$) is therefore given by

$$P_{\boldsymbol{u}}(n|\boldsymbol{\eta}) = \eta_{\boldsymbol{u}_{n-1}} \prod_{i=0}^{n-2} (1 - \eta_{\boldsymbol{u}_i}) \quad (7)$$

Since $\boldsymbol{\eta}$ (a set of parameters) is unknown, beta prior distributions with hyperparameters α and β are put on $\boldsymbol{\eta}$ as follows:

$$p(\boldsymbol{\eta}) = \prod_{\boldsymbol{u} \in \text{tree}} \text{Beta}(\eta_{\boldsymbol{u}}|\alpha, \beta) \quad (8)$$

Given the value of z_m, the value of x_m is stochastically determined according to the CRF described in Section 3.2.2. Note that there are not only proxy customers but also original customers in restaurants other than leaf nodes.

More specifically, given a particular seating arrangement denoted by $\boldsymbol{S}$, a next chord w following context $\boldsymbol{u}$ is generated according to the following predictive distribution:

$$P_{\boldsymbol{u}}^{\text{VPY}}(w|\boldsymbol{S}) = \sum_{n} P_{\boldsymbol{u}}^{\text{VPY}}(w|n, \boldsymbol{S}) P_{\boldsymbol{u}}(n|\boldsymbol{S}) \quad (9)$$

where $P_{\boldsymbol{u}}^{\text{VPY}}(w|n, \boldsymbol{S})$ is obtained in the same way as Eqn. (4) and $P_{\boldsymbol{u}}(n|\boldsymbol{S}) = \int P_{\boldsymbol{u}}(n|\boldsymbol{\eta}) p(\boldsymbol{\eta}|\boldsymbol{S}) d\boldsymbol{\eta}$ is easily calculated by using the conjugacy between Eqns. (7) and (8) (see [9]).

3.3.2 Predictive Distribution and Bayesian Inference

The predictive distribution of a next chord w is obtained in the same way as the HPYLM (Section 3.2.3). The only difference with respect to Gibbs sampling is that the VPYLM needs to sample the value of z_m from its posterior distribution before adding customer x_m to the tree. When $x_m = w$, the posterior probability of $z_m = n$ is given by

$$P_{\boldsymbol{u}}(n|\boldsymbol{S}, w) \propto P_{\boldsymbol{u}}(w, n|\boldsymbol{S}) = P_{\boldsymbol{u}}^{\text{VPY}}(w|n, \boldsymbol{S}) P_{\boldsymbol{u}}(n|\boldsymbol{S}) \quad (10)$$

3.4 Nested Pitman-Yor Language Model

An essential problem of standard n-gram models is that we need to define a finite vocabulary even though in the real world the vocabulary is growing steadily. To solve this problem in the context of *word* sequence modeling, Mochihashi *et al.* [10] proposed a nested PY language model (NPYLM) by formulating a global base measure G_0 over a countably

infinite number of variable-length words. Note that the conventional base measure $G_0(w) = 1/V$ cannot be used because $G_0(w) \to 0$ when $V \to \infty$. Instead, a spelling model based on a *letter-level* VPYLM is given as a global base measure G_0 of a *word-level* VPYLM. More specifically, each word is regarded as a sequence of letters, which are assumed to follow a letter-level CRF. The word length (the number of letters) is assumed to follow a Poisson distribution. Thus, the 0-gram probability of any word w, $G_0(w)$, is given by the product of the probabilities of the letters and their number, resulting in the *infinite*-vocabulary model.

4. VOCABULARY-FREE INFINITY-GRAM MODEL

For *chord* sequence modeling we propose a novel vocabulary-free infinity-gram model similar in spirit to the NPYLM.

4.1 Mathematical Formulation

A critical problem is that we cannot apply the NPYLM to chord sequence modeling. Because words are temporal sequences of letters and chords are simultaneous combinations of notes, we need a different base measure G_0.

To solve this problem, we formulate a probabilistic model based on the component-based notation (Section 2.2) as a global base measure G_0 of a chord-level VPYLM. The base measure G_0 is based on a conjugate model. In general, a chord w can be written as $w_0{:}w_1 \cdots w_{12}$, where w_0 is a root note and the other variables take binary values. When $w = \mathrm{N}$, $w_0 = \mathrm{N}$ and other variables are not used. We assume w_0 to follow a 13-dimensional discrete distribution and the others to follow Bernoulli distributions as follows:

$$G_0(w) = p(w|\boldsymbol{\pi}, \boldsymbol{\tau}) = \pi_{w_0} \prod_{i=1}^{12} \tau_i^{w_i} (1-\tau_i)^{1-w_i} \qquad (11)$$

where $\boldsymbol{\pi} = \{\pi_{\mathrm{C}}, \pi_{\mathrm{C\#}}, \cdots, \pi_{\mathrm{B}}, \pi_{\mathrm{N}}\}$ indicates the probabilities of the respective pitch classes and "N" and $\boldsymbol{\tau} = \{\tau_1, \cdots, \tau_{12}\}$ indicates the existence probabilities of the respective degrees. If $w = N$, $G_0(w) = \pi_{\mathrm{N}}$. Since the values of $\boldsymbol{\pi}$ and $\boldsymbol{\tau}$ are unknown, we put prior distributions as follows:

$$p(\boldsymbol{\pi}, \boldsymbol{\tau}) = \mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{a}_0) \prod_{i=1}^{12} \mathrm{Beta}(\tau_i|b_0, c_0) \qquad (12)$$

where $\boldsymbol{a}_0$, b_0, and c_0 are hyperparameters (set to 0.5).

4.2 Bayesian Inference

Given a seating arrangement $\boldsymbol{S}$, the posterior distribution of $\boldsymbol{\pi}$ and $\boldsymbol{\tau}$ can be easily calculated as follows:

$$p(\boldsymbol{\pi}, \boldsymbol{\tau}|\boldsymbol{S}) = \mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{a}_0 + \boldsymbol{n}) \prod_{i=1}^{12} \mathrm{Beta}(\tau_i|b_0 + n_i, c_0 + \bar{n}_i) \quad (13)$$

where n_v (v is one of the pitch classes or "N") is the number of tables serving dishes with root note v ($w_0 = v$) in the root restaurant ϕ, n_i is the number of tables serving dishes with the i-th note ($w_i = 1$) in ϕ, and $\bar{n}_i$ is the number of tables serving dishes without the i-th note ($w_i = 0$) in ϕ.

The predictive distribution of a next chord w can be calculated in the same way as the VPYLM (Section 3.3.2). The Gibbs sampling algorithm of the VPYLM is modified as follows: When a (proxy) customer sits at a new table (a new table is added) in the root restaurant ϕ, the values of n_v and n_i or $\bar{n}_i$ are incremented according to the components of the target chord (a dish served at that table). When a table is removed from the root restaurant ϕ, the values of n_v and n_i or $\bar{n}_i$ are decremented. The values of $\boldsymbol{\pi}$ and $\boldsymbol{\tau}$ are sampled from the posterior distribution given by Eqn. (13).

5. EXPERIMENTS

This section reports our comparative experiments.

5.1 Experimental Conditions

We used a standard dataset of chord sequences for 180 Beatles songs collected from 12 albums (13 CDs) [14]. Because the choice of chords depends on the musical key, we selected 137 major-scale non-transposition songs and transposed them to C major. The total number of chords was 10,761, where 103 chord types were observed in the label-based notation (the vocabulary size was 205) and 149 chord types were observed in the component-based notation (the vocabulary size was 49153). The entropies of both data were 3.79 [bits] and 3.92 [bits], respectively.

In the first experiment using the label-based notation, the effectiveness of infinity-gram modeling was evaluated by comparing six existing methods: Good-Turing (GT), Witten-Bell (WB), IKN, MKN, HPYLM, and VPYLM, where GT and WB are classical smoothing methods [7]. In the second experiment using the component-based notation, the effectiveness of vocabulary-free modeling was evaluated. In addition to the existing methods, we tested our models that incorporate the vocabulary-free base measure G_0 into HPYLM and VPYLM (denoted by prefix "VF-"). To evaluate the predictive performance, we conducted 10-fold cross validation and measured *perplexity*, which indicates the average number of next-chord candidates (a degree of uncertainty), given a context. A lower perplexity means better performance.

5.2 Experimental Results

We found in the first experiment that VPYLM yielded the lowest perplexity (Table 2) and that, as shown in Figure 4, a posterior distribution over n can be estimated for each chord. To obtain better predictive performance, it is important to marginalize out n (take all possibilities into account) rather than use a maximum-a-posteriori (MAP) estimate of n. The training time and memory usage of the VPYLM were two times shorter and five times smaller than those of the 10-gram HPYLM because unnecessarily-longer contexts (deep nodes) do not need to be considered (expanded). We could discover stochastically-coherent chord patterns (Table 3) by calculating $P_{\boldsymbol{u}}(w, n|\boldsymbol{X}) = \sum_{\boldsymbol{S}} P_{\boldsymbol{u}}(w, n|\boldsymbol{S}) P(\boldsymbol{S}|\boldsymbol{X})$, which indicates how likely chord w is to follow context $\boldsymbol{u}$ of length

n	GT	WB	IKN	MKN	HPYLM	VPYLM
1	16.8	15.6	16.0	15.7	15.8 (±0.03)	
2	20.3	14.2	15.2	15.8	14.5 (±0.10)	n: posterior sample
3	23.5	15.4	16.0	16.3	16.0 (±0.18)	13.4 (±0.33)
4	25.5	16.8	17.7	15.5	13.9 (±0.25)	
5	26.3	17.5	16.2	14.1	13.7 (±0.23)	n: MAP estimate
6	27.0	17.8	15.1	13.5	13.6 (±0.23)	12.9 (±0.35)
7	27.3	18.0	14.5	13.3	13.6 (±0.23)	
8	27.3	18.0	14.2	13.2	13.6 (±0.22)	n: marginalized out
9	27.3	18.0	14.1	13.1	13.5 (±0.23)	**11.9** (±0.22)
10	27.3	18.0	14.0	13.1	13.5 (±0.23)	

Table 2. Perplexities in label-based notation.

$P_{\boldsymbol{u}}(w, n\|\boldsymbol{X})$	Stochastically-coherent chord pattern ($n \geq 3$)
0.701	$n = 3$: C:7 F:7 C:7
0.682	$n = 3$: B:maj F:maj G:maj
0.656	$n = 3$: A:min C:7 F:maj
0.647	$n = 3$: F:min G:maj C:maj
0.645	$n = 4$: F:maj F:maj G:maj C:maj
0.632	$n = 3$: E:min C:7 F:maj
0.630	$n = 3$: C:maj7 D:min7 E:min7
0.627	$n = 4$: B:maj F:maj G:maj C:maj
0.622	$n = 3$: D:min7 G:sus4 G:maj
0.620	$n = 5$: D:min G:maj C:maj F:maj C:maj

Table 3. Stochastically-coherent chord patterns.

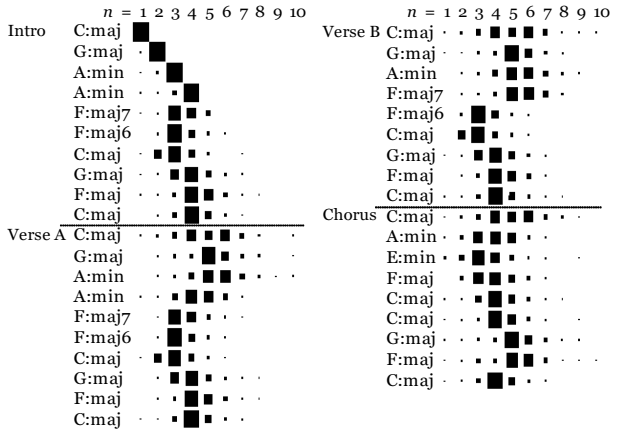

Figure 4. Hinton-diagram representation of posterior distributions over n at the beginning of the Beatles' "Let It Be."

$n - 1$. For example, C:7 F:7 C:7 is a typical blues-rock pattern that was popularized by the Beatles. We can see that the Beatles liked to use chord patterns including (major/minor) seventh chords, which were not so common at that time.

In the second experiment, VF-VPYLM, the vocabulary-free infinity-gram model, yielded a perplexity significantly lower than the other models did (Table 4). The performance advantage was larger than that in the first experiment. This proves that our model is robust to the data sparseness (large- or infinite-vocabulary situation).

6. CONCLUSION

We presented a nonparametric Bayesian n-gram model for chord sequences that requires neither a vocabulary of chord types nor a predefinition of n. We showed that it performed significantly better than the state-of-the-art models.

This study opens up a new research direction. We plan to let computers acquire the concept of "chords" in an unsupervised manner from a large amount of music scores and, ultimately, from a large amount of musical audio signals. We know that certain combinations of notes can form chords. Is this learned from experience? How reasonable is a definition of chords? To explore ways to answer this question we need to consider an infinite number of note combinations as chord candidates. Bayesian nonparametrics is a promising generative approach to such kinds of meta-level problems.

n	GT	WB	IKN	MKN
10	38.3	24.4	18.5	17.5

n	HPYLM	VF-HPYLM
10	18.0 (±0.29)	16.5 (±0.60)

n	VPYLM	VF-VPYLM
∞	15.8 (±0.29)	**14.6** (±0.55)

Table 4. Perplexities in component-based notation.

Acknowledgment: This study was partially supported by KAKENHI 23700184. We thank Dr. Daichi Mochihashi (ISM).

7. REFERENCES

[1] J.-F. Paiement, D. Eck, and S. Bengio. A probabilistic model for chord progressions. *ISMIR*, pp.312–319, 2005.

[2] R. Scholz, E. Vincent, and F. Bimbot. Robust modeling of musical chord sequences using probabilistic N-grams. *ICASSP*, pp.53–56, 2009.

[3] M. Ogihara and T. Li. N-gram chord profiles for composer style representation. *ISMIR*, pp.671–676, 2008.

[4] C. Pérez-Sancho, D. Rizo, and J. M. Iñesta. Genre classification using chords and stochastic language models. *Connection Science*, vol.21, no.2-3, pp.145–159, 2009.

[5] H.-T. Cheng, Y.-H. Yang, Y.-C. Lin, I-B. Liao, and H. H. Chen. Automatic chord recognition for music classification and retrieval. *ICME*, pp.1505–1508, 2008.

[6] M. Khadkevich and M. Omologo. Use of hidden Markov models and factored language models for automatic chord recognition. *ISMIR*, pp.561–566, 2009.

[7] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. Tech. Repo. TR-10-98, Computer Science Group, Harvard University, 1998.

[8] Y. W. Teh. A Bayesian interpretation of interpolated Kneser-Ney. Tech. Repo. TRA2/06, NUS School of Computing, 2006.

[9] D. Mochihashi and E. Sumita. The infinite Markov model. *NIPS*, pp.1017–1024, 2007.

[10] D. Mochihashi, T. Yamada, and N. Ueda. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. *ACL-IJCNLP*, pp.100–1008, 2009.

[11] M. Mauch, S. Dixon, C. Harte, B. Fields, and M. Casey. Discovering chord idioms through Beatles and real book songs, *ISMIR*, pp.255–258, 2007.

[12] M. Mauch and S. Dixon. Simultaneous estimation of chords and musical context from audio. *IEEE Trans. ASLP*, vol.18, no.6, pp.1280–1289, 2010.

[13] K. Yoshii and M. Goto. Infinite latent harmonic allocation: A nonparametric Bayesian approach to multipitch analysis. *ISMIR*, pp.309–314, 2010.

[14] C. Harte, M. Sandler, S. Abdallah, and E. Gómez. Symbolic representation of musical chords: A proposed syntax for text annotations. *ISMIR*, pp.66–71, 2005.

[15] J. Pitman and M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, vol.25, no.2, pp.855–900, 1997.