MELODY HARMONIZATION USING ORDERLESS NADE, CHORD BALANCING, AND BLOCKED GIBBS SAMPLING

Chung-En Sun, Yi-Wei Chen, Hung-Shin Lee, Yen-Hsing Chen, Hsin-Min Wang

Institute of Information Science, Academia Sinica, Taiwan

ABSTRACT

Coherence and interestingness are two criteria for evaluating the performance of melody harmonization, which aims to generate a chord progression from a symbolic melody. In this study, we apply the concept of orderless NADE, which takes the melody and its partially masked chord sequence as the input of the BiLSTM-based networks to learn the masked ground truth, to the training process. In addition, class weighting is used to compensate for some reasonable chord labels that are rarely seen in the training set. Consistent with the stochasticity in training, blocked Gibbs sampling with proper numbers of masking/generating loops is used in the inference phase to progressively trade the coherence of the generated chord sequence off against its interestingness. The experiments were conducted on a dataset of 18,005 melody/chord pairs. Our proposed model outperforms the state-of-the-art system MTHarmonizer in five of six different objective metrics based on chord/melody harmonicity and chord progression. The subjective test results with more than 100 participants also show the superiority of our model.

Index Terms— Melody harmonization, orderless NADE, blocked Gibbs sampling, sample weighting.

1. INTRODUCTION

Automatic melody harmonization aims to build up a learning machine that can generate chord sequences to accompany a given melody [1, 2]. In music, a chord is any harmonic set composed of three or more notes that are heard as if sounding simultaneously [3]. However, melody harmonization is a highly subjective task that is difficult for machines to learn to imitate humans or fit human preferences. While some harmonization may be considered appropriate in some cases, it can be changed according to different contexts.

Many approaches have been proposed for automatic melody harmonization [3], such as hidden Markov models (HMMs) [4, 5, 6] and genetic algorithm (GA)-based methods [7]. Recently, with the prevalence of deep learning models, some deep learning methods have emerged to deal with the melody harmonization problem [8]. Lim et al. [9] first proposed a model composed of bidirectional long short-term memory (BiLSTM) layers [10]. The model predicts a chord for each bar from 24 major and minor chords. Based on the same neural structure, Yeh et al. extended the number of chords to 48 by considering major, minor, augment and diminish chords. In addition, they integrated information of chord functions [11] into the loss function to help chord label prediction, and predicted a chord every half bar [12]. Experiments showed that their model, called MTHarmonizer, could generate more creative and elaborated chord progressions than the previous model. Not only the expanded diversity of chord labels, the reason also lies in that MTHarmonizer can handle the transition between tonic (T), subdominant (S), and dominant (D) chords due to the use of chord functions.

However, there are several drawbacks in the above models. For example, Lim’s model overuses common chords and has an incorrect phrasing problem [12]. Yeh’s model tried to deal with these problems, but still could not appropriately assign altered chords through chord function prediction, because the chord function (TSD) easily loses its function when a secondary or borrowed chord event occurs.

In this study, we propose a BiLSTM-based model stemming from the structure of orderless NADE [13, 14], which takes the melody and its partially masked chord sequence as the input to learn the masked ground truth. Instead of chord functions, class weighting is used to compensate for some reasonable chord labels that are rarely seen in the training set. In addition, we perform blocked Gibbs sampling during inference [15], similar to CoCoNet used in Google’s Bach Doodle [16, 17]. As a result, our model can predict 96 chords, including major, minor, augment, diminish, suspend, major7, minor7, and dominant7. Through the chord balancing process in training, we finally achieve a generative model that can produce rich but still reasonable chord progressions. Such interesting progressions with sufficient tension have a close quality to that of human-composed progressions.

2. PROPOSED MODEL

2.1. Neural Structure

Similar to Lim’s [9] and Yeh’s [12] models, the neural structure of our model is mainly composed of two BiLSTM layers, followed by a fully connected layer, as shown in Fig. 1. In the training phase, dropout with a probability of 0.2 is applied
Finally, the task of our model is to reconstruct \( x_{\text{chord} \neq C} \), the complement of \( x_{\text{chord} = C} \), from \( x_{\text{input}} \).

2.2. Loss Function

The loss function with negative log-likelihood is expressed by

\[
\mathcal{L}(x_{\text{melody}}, x_{\text{chord}}, C, \theta) = -\sum_{t \not\in C} \sum_{s} \log p_{\theta}(x_{\text{chord}_{t,s}} | x_{\text{melody}}, x_{\text{chord} = C}, C)
\]

(3)

where \( w_s \) is the weight of chord \( s \) [22], \( x_{\text{chord}_{t,s}} \) is the \( s \)-th element of the one-hot vector \( x_{\text{chord}_{t,s}} \), \( C \) is the context given, and \( \theta \) denotes the model parameters. Chord weighting can greatly solve the problem of excessive use of common chords in Lim’s model [9] like Yeh’s chord function [12]. However, our model can further manage to assign appropriate altered chords when the melody requires a secondary or borrowed chord to elaborate the whole piece, but Yeh’s model [12] cannot. The weight matrix is derived by

\[
w_s = |S| \times \frac{1/(1000 + n_s)}{\sum_{s'} 1/(1000 + n_{s'})}
\]

(4)

where \( n_s \) is the number of times chord \( s \) appears in the training data, and \( n_s \) is added by 1000 for count smoothing. This process is also called inverse class frequency weighting.

The model is trained by minimizing

\[
\mathbb{E}_{x_{\text{chord}} \sim p(x_{\text{chord}})} \mathbb{E}_{C \sim p(C)} \frac{1}{|C|} \mathcal{L}(x_{\text{melody}}, x_{\text{chord}}; C, \theta).
\]

(5)

The expectations are estimated by sampling \( x_{\text{chord}} \) from the training set and selecting a single context \( C \) per sample.

2.3. Inference

The original orderless NADE model uses ancestral sampling in the inference stage [13, 14], which samples one variable at a time in any order. Therefore, we can randomly choose an order, and then sample all the variables according to that order. However, this process may cause errors to propagate through that specific order. Furthermore, it is shown in [16] that some \( p_{\theta}(x_{\text{chord}_{t,s}} | x_{C}, C), t \not\in C \), may be poorly modeled, especially when \( C \) is small.

Therefore, we use blocked Gibbs sampling [15], a similar sampling procedure in [16, 17]. The blocked Gibbs sampling is an orderless sampling process that perfectly matches the training process of our model. There is no need to select an order first; instead, we randomly choose the blocks to be masked and ask the model to reconstruct them in each iteration. There is an annealing probability to determine the proportion of the variables that remain unchanged. At first, all the variables will be erased and sampled again. As the iteration goes on, more and more variables will be kept unchanged. Eventually, the output will converge and the joint
probabilty can be estimated elaborately. Algorithm 1 elaborates our sampling process in the inference stage. The number of iterations $n$ is set to the average length of chord sequences, $P_{\text{max}}$ is set to 1, and $P_{\text{min}}$ is set to 0.05.

### 3. EXPERIMENTS AND EVALUATIONS

#### 3.1. Dataset

The HookTheory Lead Sheet Dataset (HLSD) is used in this study. The dataset is collected through a community-based platform for users to upload lead sheets. It contains high-quality, human-arranged melodies with chord progressions and other information. After keeping song tracks containing melody/chord pairs, we constructed a dataset of 18,005 samples for this study. We divided the dataset into a training set containing 17,505 samples and a test set containing 500 samples. The dataset contains chords that are beyond our chord space, such as 9th, 11th, 13th, half-diminished, and slashed chords. We transferred these chords into one of the 96 chords according to their chord functions. The slashed chords were returned to their root positions. All the pieces were transferred into either C major or C minor. In addition, by sampling chords at the beginning and middle of a bar, we made the chords in the dataset last at least half a bar.

#### 3.2. Metrics

For objective evaluation, we use six different objective metrics introduced in [12].

- **Pitch consonance score (PCS):** The consonance score that is computed through every note in a given chord.
- **Melody-chord tonal distance (MCTD):** The tonal distance between a note and a chord in 6-D feature vectors.

The first three metrics measure the quality of the chord progression, and the others measure the coherence between the melody and chords. For details about the metrics, see [12].

For subjective evaluation, we randomly selected 50 melodies from our test set. Each human subject needs to evaluate the chord progressions of five melodies. For each melody, the subjects listen to the melody without chords first, and then three different harmonizations composed by human, MTHarmonizer, and our model in random order.

We design four criteria to obtain feedback from subjects:

- **Coherence:** how well the chord progression matches with the melody in terms of harmonicity and phrasing.
- **Chord progression:** how reasonable or smooth the chord progression is on its own, independent from the melody.
- **Interestingness:** how surprising or innovative the chord progression is on its own, independent from the melody.
- **Overall:** how well the whole harmonization is in comprehensive consideration.

A total of 102 subjects participated in the test. The ratings of 9 subjects who always gave the same score regardless of different harmonization were discarded. After the cleaning step, 93 subjects remained. 57 of them know the definition of chord progression. In addition, if the overall score contradicted the ranking result, the ratings for a melody were discarded.

#### 3.3. Results

**3.3.1. Objective tests**

The results are shown in Table 1. Before we get into the discussion, it should be noted that none of the objective metrics mentioned in Section 3.2 can directly reflect the quality of the composed chords. It is difficult to quantify which metrics are better for which purposes and how accurate these metrics are.

The human ear is still the best criterion for all music pieces.

In terms of the melody/chord harmonicit metrics (i.e., CTrCTR, PCS, and MCTD), we can see that the human composed pieces are worse than the automatically composed pieces. We think that this happens because human composers use many 7th, 9th, and 11th chords to produce more interesting chord progressions. However, these modern chords may degrade the performance in the melody/chord harmonicit metric. As a result, we should not conclude that in terms of melody/chord harmonicit, the performance of human composers is worse than that of melody harmonization systems.

Comparing our model (cf. BGS w/o B, $|S| = 48$) with MTHarmonizer, we find that the introduction of the orderless NADE training process and blocked Gibbs sampling can improve the performance in all three melody/chord
In this paper, we proposed a framework based on the BiLSTM structure and the orderless NADE training process for automatic melody harmonization. We applied blocked Gibbs sampling in inference. In addition, we applied chord balancing and augmented the chord space to 96 to increase the diversity of chord progressions. The objective and subjective test results demonstrate that our model is better than Yeh’s model and is comparable in performance to human composers.

In our future work, we will try to make our neural model learn Roman numeral analysis, which is how humans learn how to compose chords, and one of the reasons why our proposed model cannot surpass human composers.

### Table 1. Objective evaluation scores of different models. BGS denotes blocked Gibbs sampling and B denotes chord balancing. Higher values in CTnCTR and PCS and lower values in CTD indicate better melody/chord harmonicity. Higher values in CHE and CC and lower values in CTD implicitly mean better interestingness of chord progression.

| M/C Harmonicity | CTnCTR↑ | PCS↑ | MCTD↓ |
|-----------------|---------|------|--------|
| Humans          | 0.726   | 0.515 | 1.276  |
| MTHarmonizer, | 0.804   | 0.576 | 1.152  |
| $|S| = 48$          |         |       |        |
| BGS w/o B, | 0.850   | 0.652 | 1.087  |
| $|S| = 48$          |         |       |        |
| BGS w/o B, | 0.853   | 0.657 | 1.053  |
| $|S| = 96$          |         |       |        |
| BGS w/ B, | 0.876   | 0.653 | 1.068  |
| $|S| = 48$          |         |       |        |
| BGS w/ B, | 0.887   | 0.652 | 1.052  |
| $|S| = 96$          |         |       |        |

| Chord Progression | CHE↑ | CC↑ | CTD↓ |
|-------------------|------|-----|------|
| Humans            | 1.266 | 4.344 | 0.628 |
| MTHarmonizer, | 0.859   | 3.104 | 0.512 |
| $|S| = 48$          |         |       |        |
| BGS w/o B, | 0.755   | 2.926 | 0.521 |
| $|S| = 48$          |         |       |        |
| BGS w/ B, | 0.726   | 2.862 | 0.540 |
| $|S| = 48$          |         |       |        |
| BGS w/ B, | 1.209   | 4.698 | 0.690 |
| $|S| = 96$          |         |       |        |
| BGS w/ B, | 1.280   | 4.900 | 0.730 |
| $|S| = 96$          |         |       |        |

| Method              | Votes | Rate of the Best % |
|---------------------|-------|-------------------|
| Humans              | 87    | 31.9              |
| MTHarmonizer        | 66    | 24.2              |
| Proposed            | 86    | 31.5              |
| No preference        | 34    | 12.5              |

3.3.2. Subjective tests

Fig. 2 shows the results of the subjective test. Our model (BGS w/ B, $|S| = 96$) surpasses MTHarmonizer in all criteria and has close results to human composers. Our model has excellent performance in terms of coherence, which is consistent with the high melody/chord harmonicity in the objective evaluation. The high score also indicates that with blocked Gibbs sampling, our model can produce more reasonable and smoother chord progressions.

The subjects were also asked to choose the best from three chord progressions of a melody. Table 2 shows that the music pieces composed by humans are the most favorable, but the percentage of being selected as the best is only slightly higher than that of our model. We conclude that our model can arrange music pieces very well.

Several examples are available at https://chord-generation.herokuapp.com/demo. They are randomly rendered from all 500 test samples in HLSD.

### 4. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a framework based on the BiLSTM structure and the orderless NADE training process for automatic melody harmonization. We applied blocked Gibbs sampling in inference. In addition, we applied chord balancing and augmented the chord space to 96 to increase the diversity of chord progressions. The objective and subjective test results demonstrate that our model is better than Yeh’s model and is comparable in performance to human composers.

In our future work, we will try to make our neural model learn Roman numeral analysis, which is how humans learn how to compose chords, and one of the reasons why our proposed model cannot surpass human composers.
5. REFERENCES

[1] Ching-Hua Chuan and Elaine Chew, “A hybrid system for automatic generation of style-specific accompaniment,” in *Proc. IJWCC*, 2007.

[2] Ian Simon, Dan Morris, and Sumit Basu, “MySong: Automatic accompaniment generation for vocal melodies,” in *Proc. CHI*, 2008.

[3] Dimos Makris, Ioannis Kayrdis, and Spyros Sioutas, “Automatic melodic harmonization: An overview, challenges and future directions,” in *Trends in Music Information Seeking, Behavior, and Retrieval for Creativity*, pp. 146–165. IGI Global, 2016.

[4] Jean François Paiement, Douglas Eck, and Samy Bengio, “Probabilistic melodic harmonization,” in *Proc. Canadian AI*, 2006.

[5] Hiroaki Tsushima, Eita Nakamura, Katsutoshi Itoyama, and Kazuyoshi Yoshii, “Function- and rhythm-aware melody harmonization based on tree-structured parsing and split-merge sampling of chord sequences,” in *Proc. ISMIR*, 2017.

[6] Hiroaki Tsushima, Eita Nakamura, Katsutoshi Itoyama, and Kazuyoshi Yoshii, “Generative statistical models with self-emergent grammar of chord sequences,” *J. New Music Res.*, vol. 47, no. 3, pp. 226–248, 2018.

[7] Tetsuro Kitahara, Sergio Giraldo, and Rafael Ramírez, “JamSketch: Improvisation support system with GA-based melody creation from user’s drawing,” in *Proc. CMMR*, 2018.

[8] Jean-Pierre Briot, Gaétan Hadjeres, and François-David Pachet, “Deep learning techniques for music generation: A Survey,” 2017.

[9] Hyungui Lim, Seungyeon Rhyu, and Kyogu Lee, “Chord generation from symbolic melody using BLSTM networks,” in *Proc. ISMIR*, 2017.

[10] Sepp Hochreiter and Jürgen Schmidhuber, “Long short-term memory,” *J. Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] Tsung-Ping Chen and Li Su, “Functional harmony recognition of symbolic music data with multi-task recurrent neural networks,” in *Proc. ISMIR*, 2018.

[12] Yin-Cheng Yeh, Wen-Yi Hsiao, Satoru Fukayama, TeTsuro Kitahara, Benjamin Genchel, Hao-Min Liu, Hao-Wen Dong, Yian Chen, Terence Leong, and Yi-Hsuan Yang, “Automatic melody harmonization with triad chords: A comparative study,” 2020.

[13] Benigno Uria, Iain Murray, and Hugo Larochelle, “A deep and tractable density estimator,” in *Proc. ICML*, 2014.

[14] Benigno Uria, Marc-Alexandre Côté, Karol Gregor, Iain Murray, and Hugo Larochelle, “Neural autoregressive distribution estimation,” *J. Mach. Learn. Res.*, vol. 17, pp. 1–37, 2016.

[15] Li Yao, Sherjil Ozair, Kyunghyun Cho, and Yoshua Bengio, “On the equivalence between deep NADE and generative stochastic networks,” in *Proc. ECML*, 2014.

[16] Cheng-Zhi Anna Huang, Tim Cooijmans, Adam Roberts, Aaron Courville, and Douglas Eck, “Counterpoint by convolution,” in *Proc. ISMIR*, 2017.

[17] Cheng-Zhi Anna Huang, Curtis Hawthorne, Adam Roberts, Monica Dinculescu, James Wexler, Leon Hong, and Jacob Howcroft, “The Bach doodle: Approachable music composition with machine learning at scale,” in *Proc. ISMIR*, 2019.

[18] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, and Ruslan Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[19] Sebastian Ruder, “An overview of gradient descent optimization algorithms,” 2016.

[20] Diederik P. Kingma and Jimmy Lei Ba, “Adam: A method for stochastic optimization,” in *Proc. ICLR*, 2015, pp. 1–15.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” in *Proc. ICCV*, 2016.

[22] Yin Cui, Menglin Jia, Tsung Yi Lin, Yang Song, and Serge Belongie, “Class-balanced loss based on effective number of samples,” in *Proc. CVPR*, 2019.

[23] Christopher Harte, Mark Sandler, and Martin Gasser, “Detecting harmonic change in musical audio,” in *Proc. AMCM*, 2006.