



HAL
open science

The Chordinator: Chord progression modeling and generation using transformers

David Dalmazzo, Ken Déguernel, Bob L. T. Sturm

► **To cite this version:**

David Dalmazzo, Ken Déguernel, Bob L. T. Sturm. The Chordinator: Chord progression modeling and generation using transformers. International Society for Music Information Retrieval Conference, 2023. hal-04289026

HAL Id: hal-04289026

<https://hal.science/hal-04289026v1>

Submitted on 16 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

THE CHORDINATOR: CHORD PROGRESSION MODELING AND GENERATION USING TRANSFORMERS

David Dalmazzo

KTH-Royal Institute of Technology
Stockholm, Sweden
dalmazzo@kth.se

Ken Déguernel

Univ. Lille, CNRS, Centrale Lille,
UMR 9189 CRISTAL, F-59000 Lille, France
ken.deguernel@cnrs.fr

Bob L. T. Sturm

KTH-Royal Institute of Technology
Stockholm, Sweden
bobs@kth.se

ABSTRACT

This paper presents a transformer model trained with a large dataset of chord sequences. The dataset includes several styles, such as jazz, rock, pop, blues, or music for cinema. We apply three consecutive tokenization/encoding strategies: 1) All chords are treated as unique elements. 2) Chords dynamically formatted as a tuple describing roots, nature, extensions, and slash chords. 3) An extension of model 2 with a style token and extension of the positional embedding layer of the transformer architecture. We analyze sequences generated by comparing them with the training dataset using trigram, which reveals common chord progressions and source duplications. We compare the generated sequences from a musical perspective, rating their plausibility in regard to the training data. The third strategy reported lower validation loss and better musical consistency in the suggested progressions.

1. INTRODUCTION

There has been a remarkable surge of applications that use artificial intelligence (AI) to assist in creative processes, from synthesizing graphics using text prompts¹ or the language-based models such as Generative Pre-trained Transformer (GPT)². The domain of music has also seen a variety of applications, from symbolic music generation [1, 2] to sound synthesis [3, 4].

Neural Network based systems suggesting chord progressions are not new. ChordRipple [5] is based on a WordToVec network learning embeddings from a corpus of chord progressions. Long Short-Term Memory [6] has been applied for symbolic music generation; however, transformers have reported an improvement in encoding long-term dependencies. Li and Y. Sung [7] propose an encoder-decoder (Seq2Seq) architecture to gener-

¹ See stable diffusion: <https://huggingface.co/spaces/stabilityai/stable-diffusion>

² See copilot: <https://github.com/features/copilot>

ate chord progressions given a melody. Later proposed models are based on adapting the transformer, extending the embedding layer with musical features [8, 9]. For instance, modeling Jazz styles [10], creating a system for melody and basic chord completion [8] [11], chords classifications methods [12]; however, those examples highly reduces the chord vocabulary for simplifications purposes.

Modeling symbolic music notation encoding form, structure, and harmonic information without losing vocabulary is one of the current challenges in the field. Hence, this paper focuses on modeling chord progressions of several popular music styles (see Figure 1). Such models can be applied to music creation by proposing plausible paths in form and harmony to a user, from which melodies can be derived and arrangements made. The contribution of the proposed system is 1) A dataset with chord sequences from several popular music styles. 2) A tokenization method to reduce the token vocabulary without reducing the chord vocabulary. 3) An update of the transformer network to better suit music information encoding.

2. THE CHORDINATOR

The database was formed by exporting 4300 chord progressions included in the *iReal Pro* application. The exported format is musicXML, which includes metadata and information regarding the form, tempo, bars, and chords. We apply data augmentation by transposing all 4/4 songs into the 18 enharmonic keys, ending with a shuffled dataset of 70,812 pieces. We train a minGPT³ transformer architecture with a fixed-length sequence with 512-time steps, adding the token `<pad>` to those sequences smaller than the reference size.

While encoding chord sequences, we apply three subsequent strategies.

Model 1: We treat all chords as unique elements, having a vocabulary size of 5202 tokens.

Model 2: We split the chord unities into sections, simplifying the tokenization range without reducing chord vocabulary. The chord is composed of a tuple with dynamic content. The array contains **root**, **nature**, and **extensions**, starting the chord element with an identifier token "." (dot).

Model 3: We extend Model 2 by adding a context token of the songs with a style identifier (see Figure 1). We complement the transformer architecture by adding a MIDI

³ <https://github.com/karpathy/minGPT>

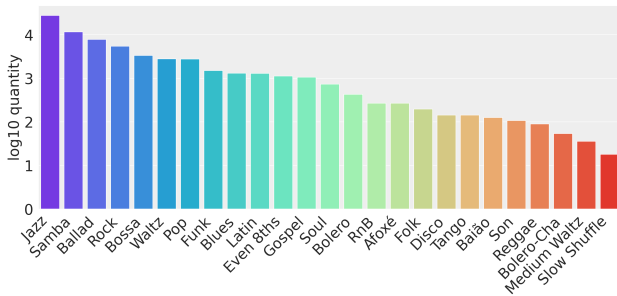


Figure 1. List of musical styles, quantity is scaled to log10.

Embedding relative position (see Figure 2) array into the positional embedding layer. The array is composed of eight values exported by translating ABC chord notation to MIDI (using libraries *music21* and *librosa*). The first element is the MIDI root; the following three values describe the **nature**, and the next three available values are used for **extensions**. Seven MIDI values express the chord’s quality, whereas the eighth value is an identifier id (127) to depict a slash chord.

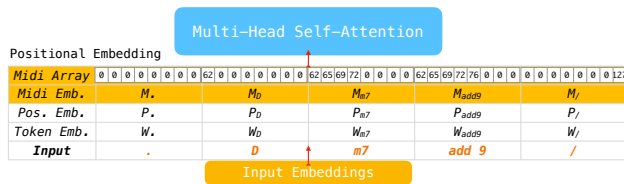


Figure 2. MIDI Embeddings: Formatted as arrays of eight values for each time step containing chord notes

3. PRELIMINARY RESULTS

Trigrams: They are composed of a minimal cell of three chords. The most common trigrams found in the dataset are shown in Figure 3. Nevertheless, the most common functional harmonies found in the dataset are shown in Figure 4.

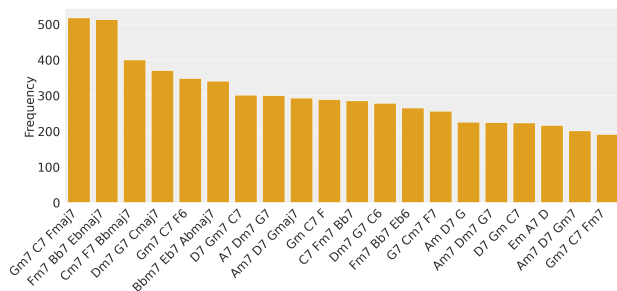


Figure 3. Most common trigrams found in the dataset

	Train Loss	Val. Loss	Perplexity
Model 1	0.2805	0.3802	1.505
Model 2	0.25176	0.32327	1.363
Model 3	0.18609	0.17451	0.2435

Table 1. Metrics comparing the three proposed models

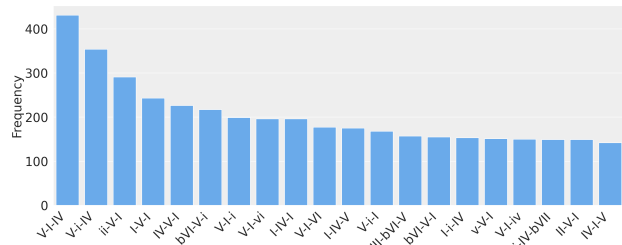


Figure 4. Most common harmonic functions found in the dataset

We compute perplexity to quantify the model’s uncertainty. The resulting metrics, loss, validation curves, and perplexity are expressed in Table 1. Ten chord sequences are exported as suggestions using the three models, creating a reference example based on 30 randomized ones no longer than 24 chords per suggestion. As a first attempt at musical evaluation, we asked five colleagues with a music education background to score their plausibility from 0 to 5, where 0 is not helpful for a potential song, and 5 reflects the suggested sequence is suitable for starting a composition. The overall score of the models is Model 1: **2.425**, Model 2: **2.825**, and Model 3: **4.125**. An example of chord suggestions made by the Model 3 is shown in Figure 5

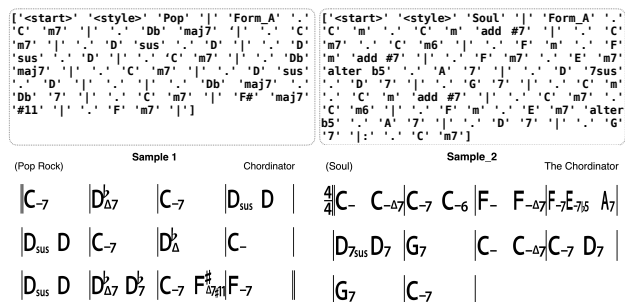


Figure 5. Two samples from Model 3 using Pop and Soul context token.

4. CONCLUSION

Model 3 reports a more consistent suggested progression and lower loss rates. Nevertheless, the perplexity metrics do not reflect musicality per se; further analysis is needed to define its plausibility in musical terms. Model 3, compared to Models 1 and 2, is not prone to generating *ii V I* progressions when the style token is not precisely Jazz or Bossa. Based on the metrics, adding style tokens and complementing the transformer’s positional embeddings with MIDI chord information seems to be a correct subsequent update to modeling chord progressions.

For future work, a web application as an interface to play with the model and generate suggestions is under development; it will open its usability to a broader audience. We will include a melody generator that follows the chord progressions. We will perform further human evaluation from expert musicians to test the impact and usability of the system.

5. REFERENCES

- [1] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. Dai, M. Hoffman, and D. Eck, "Music transformer: Generating music with long-term structure (2018)," *arXiv preprint arXiv:1809.04281*, 2018.
- [2] J.-P. Briot, G. Hadjeres, and F. Pachet, *Deep Learning Techniques for Music Generation*. Springer, 2019.
- [3] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," *ArXiv e-prints (1609.03499)*, Sep. 2016.
- [4] A. Caillon and P. Esling, "Rave: A variational autoencoder for fast and high-quality neural audio synthesis," *ArXiv e-prints*, 2021.
- [5] C.-Z. A. Huang, D. Duvenaud, and K. Z. Gajos, "Chordriple: Recommending chords to help novice composers go beyond the ordinary," in *Proceedings of the 21st international conference on intelligent user interfaces*, 2016, pp. 241–250.
- [6] K. Choi, G. Fazekas, and M. Sandler, "Text-based lstm networks for automatic music composition," in *Proc. 1st Conference on Computer Simulation of Musical Creativity*, Huddersfield, UK, 2016.
- [7] S. Li and Y. Sung, "Transformer-based seq2seq model for chord progression generation," *Mathematics*, vol. 11, no. 5, p. 1111, 2023.
- [8] M. Zeng, X. Tan, R. Wang, Z. Ju, T. Qin, and T.-Y. Liu, "Musicbert: Symbolic music understanding with large-scale pre-training," *arXiv preprint arXiv:2106.05630*, 2021.
- [9] Z. Guo, J. Kang, and D. Herremans, "A domain-knowledge-inspired music embedding space and a novel attention mechanism for symbolic music modeling," *arXiv preprint arXiv:2212.00973*, 2022.
- [10] M. Pfeleiderer, K. Frieler, J. Abeßer, W.-G. Zaddach, and B. Burkhardt, Eds., *Inside the Jazzomat - New Perspectives for Jazz Research*. Schott Campus, 2017.
- [11] S. Li and Y. Sung, "Mrbert: Pre-training of melody and rhythm for automatic music generation," *Mathematics*, vol. 11, no. 4, p. 798, 2023.
- [12] T.-P. Chen and L. Su, "Attend to chords: Improving harmonic analysis of symbolic music using transformer-based models," *Transactions of the International Society for Music Information Retrieval*, vol. 4, no. 1, 2021.