Latent Chords: Generative Piano Chord Synthesis with Variational Autoencoders

Agustín Macaya, Rodrigo F. Cádiz, Manuel Cartagena, Denis Parra*
{aamacaya,rcadiz,micartagena}@uc.cl,dparra@ing.puc.cl
Pontificia Universidad Católica de Chile
Santiago, Chile

ABSTRACT

Advances in the latest years on neural generative models such as GANs and VAEs have unveiled a great potential for creative applications supported by artificial intelligence methods. The most known applications have occurred in areas such as image synthesis for face generation as well as in natural language generation. In terms of tools for music composition, several systems have been released in the latest years, but there is still space for improving the possibilities of music co-creation with neural generative tools. In this context, we introduce *Latent Chords*, a system based on a Variational Autoencoder architecture which learns a latent space by reconstructing piano chords. We provide details of the neural architecture, the training process and we also show how *Latent Chords* can be used for a controllable exploration of chord sounds as well as to generate new chords by manipulating the latent representation. We make our training dataset, code and sound examples open and available at https://github.com/CreativAI-UC/TimbreNet

CCS CONCEPTS

• **Applied computing → Sound and music computing**; • **Computing methodologies** → *Neural networks.*

KEYWORDS

Visual Analytics, Explainable AI, Automated Machine Learning

ACM Reference Format:
Agustín Macaya, Rodrigo F. Cádiz, Manuel Cartagena, Denis Parra. 2020. Latent Chords: Generative Piano Chord Synthesis with Variational Autoencoders . In *IUI '20 Workshops, March 17, 2020, Cagliari, Italy.* ACM, New York, NY, USA, 4 pages.

1 INTRODUCTION

The promise of Deep Learning (DL) is to discover rich and hierarchical models that represent probability data distributions encountered in artificial intelligence applications, such as natural images or audio [6]. This potential of DL, when carefully analyzed, makes music and ideal application domain, being in essence very rich, structured and also hierarchical information encoded in either a symbolic score format or as audio waveforms.

It is no surprise then that the spectacular growth of DL has also greatly impacted the world of the arts. Classical tasks that can be addressed through DL are tasks that have to do with classification and estimation of numerical quantities. But perhaps one of the most interesting aspects that these networks can do now is the generation of content. In particular, there are network architectures that are capable of generating images, text or artistic content such as paintings or music [2]. Different authors have designed and studied networks capable of classifying music, recommending new music, learning the style of a visual work, among other things. Perhaps one of the most relevant and recognized efforts at present is the Magenta project [1], carried out by Google Brain, one of the branches of the company in charge of using AI in its processes. According to their website, the goal of Magenta is to explore the role of machine learning as a tool in the creative process.

DL models have been proven useful even in very difficult computational tasks, such as to solve reconstructions, deconvolutions and inverse problems with increasing accuracy over time [6, 12]. However, this great capacity of neural networks for classification and regression is not what interests us the most. It has been shown that deep learning models can now generate very realistic visual or audible content, fooling even the most expert humans. In particular, variational auto-encoders (VAEs) and generative adversarial networks (GANs) have produced shocking results in the last couple of years, as we discuss now.

One of the most important motivations for using DL to generate musical content is its generality. As [2] emphasize: "*As opposed to handcrafted models, such as grammar-based or rule-based music generation systems, a machine learning-based generation system can be agnostic, as it learns a model from an arbitrary corpus of music. As a result, the same system may be used for various musical genres. Therefore, as more large scale musical datasets are made available, a machine learning-based generation system will be able to automatically learn a musical style from a corpus and to generate new musical content*". In summary, as opposed to structured representations like rules and grammars, DL excels at processing raw unstructured data, from which its hierarchy of layers will extract higher level representations adapted to the task. We believe that this capacities make DL a very interesting technique to be explored for the generation of novel musical content. Among all the potential tasks in music generation and composition which can be supported by DL models, in this work we focus on chord synthesis. In particular we leverage Variational Autoencoders in order to learn a compressed latent space which allows controlled exploration of piano chords as well as generation of new chords unobserved in the training dataset.

[*] Also with IMFD.

[1] https://magenta.tensorflow.org/

The contributions of this work are the following. First, we constructed a dataset of 450 chords recorded on the piano at different levels of dynamics and pitch ranges (octaves). Second, we designed a VAE which is very similar in architecture as the one described in GanSynth [5], the difference being that they use a GAN while we implemented a VAE. We chose a VAE architecture to decrease the chance of problems such as training convergence and mode collapse present in GANs [11, 13]. Third, we train our model in such a way to obtain a two dimensional latent space that could adequately represent all the information contained in the dataset. Fourth, we explored this latent space in order to study how the different families of chords were represented and how both dynamic and pitch content operate on this space. Finally, we explored the generation of both new chords and harmonic trajectories by sampling points in this latent space.

2 RELATED WORK

Generative models have been extensively used for musical analysis and retrieval. We now discuss a few of the most relevant work with generative models from music from the last couple of years to get an idea of the variety of applications that these techniques offer.

In terms of content generation, there are many recent works that are very interesting. One of them is DeepBach [7], a neural network that is capable of harmonizing Bach-style chorals in a very convincing way. MidiNet [21] is a convolutional adversary generation network that generates melodies in symbolic format (MIDI) by generating counterexamples from white noise. MuseGAN [4] is network based on an adversary generation of symbolic music and accompaniment, specifically targeted for the rock genre. Wavenet [14] is a network that renders audio waves directly, without going through any kind of musical representation. Wavenet has been tested in human voice and speech. NSynth [5] is a kind of timbre interpolation system that can create new types of very convincing and expressive sounds, by morphing between different sound sources. In [19], the authors introduced a DL technique to autonomously generate novel melodies that are variations of an arbitrary base melody. They designed a neural network model that ensures, with high probability, that the melodic and rhythmic structure of the new melody would be consistent with a given set of sample songs. One important aspect of this work is that they propose to use Perlin noise instead of the widely use white noise in VAEs. [20] proposed a DL architecture called Variational Recurrent Autoencoder (VRASH), supported by history, that uses previous outputs as additional inputs. The authors claim that this model *listens* to the notes that it has composed already and uses them as additional "historic" input. In [16] the authors applied VAE techniques to the generation of musical sequences at various measure scales. In a further development of this work, the authors created MusicVAE [17], a network with a self-coding structure that is capable of generating latent spaces through which it is possible to generate audio and music content through interpolations in these latent spaces.

Generative models have also been used for music transcription problems. In [18], the authors designed generative long short-term memory (LTSM) networks for music transcription modelling and composition. Their aim is to develop transcription models of music that can be of help in musical composition situations. For the specific case of chords, there is a quite large number of research devoted to chord recognition (some notable examples are [3, 9, 12, 22]), but much less work has been devoted to chord generation. Our work is based on GanSynth [5], a GAN model that can generate an entire audio clip from a single latent vector, allowing for a smooth control of features such as pitch and timbre. Our model, as we specify below, works in a similar fashion but is was customized for the specific case of chord sequences.

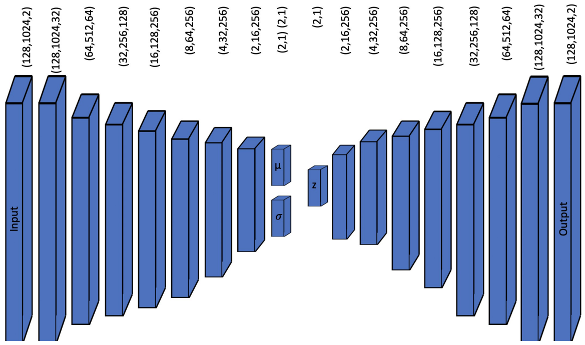

Figure 1: Arquitecture of our VAE model for chord synthesis.

3 NETWORK ARCHITECTURE

The network architecture is presented in Figure 1. Our design goal was not only content generation and latent space exploration, but also to generate a tool useful for musical composition. A VAE based model has the advantage over a GAN model of having an encoder network that can accept inputs from the user and a decoder network that can generate new outputs. Although it is possible to replicate these features with a conditional GAN, we prefer using a VAE since GANs have known problems of training convergence and mode collapse [11, 13] we prefer to avoid in this early stage of our project. Still, we based the encoder architecture from the discriminator of GanSynth [5] and the decoder architecture from the generator of GanSynth.

The encoder takes a (128,1024,2) MFCC (Mel Frequency Cepstral Coefficients) image and passes it through one conv 2D layer with 32 filters generating a (128,1024,32) output that then passes through a series of 2 conv2D layers with the same size padding and a Leaky ReLU non-linear activation function followed by 2x2 downsampling layers. This process keeps halving the images' size and duplicating the number of channels until a (2,16,256) layer is obtained. Then, a fully connected layer outputs a (4,1) vector that contains the two means and the two standard deviations for later sampling.

The sampling process takes a (2,1) mean vector and a (2,2) standard deviation diagonal matrix and using those parameters we sample a (2,1) latent vector z from a normal distribution.

The decoding process takes the (2,1) z latent vector and passes it throw a fully connected layer that generates a (2,16,256) output that then is followed by a series of 2 transposed convD layers followed by an 2x2 upsampling layer that keeps doubling the size of the image and halving the number of channels until a (128,1024,32) output is obtained. This output passes through a last convolutional layer that outputs the (128,1024,2) MFCC spectral representation

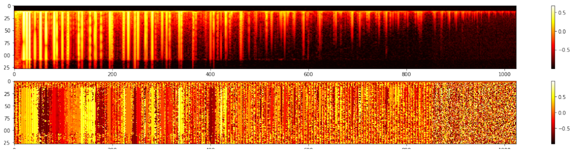

Figure 2: MFCC representation of a *forte* chord

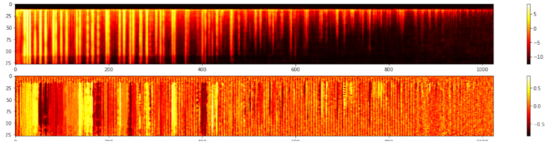

Figure 3: MFCC representation of the forte chord generated by the network

of the generated audio. Inverse MFCC and STFT are then used to reconstruct a 4 second audio signal.

4 DATASET AND MODEL TRAINING

Our dataset consists on 450 recordings of 15 piano chords played at different keys, dynamics and octaves, performed by the main author. Each recording has a duration of 4 seconds, and were recorded with a sampling rate of 16 kHz in Ableton Live in the wav audio format. Piano keys were pressed for three seconds and released at the last second. The format of the dataset is the same as used in [5].

Tbe chords that we included in the dataset were: C2, Dm2, Em2, F2, G2, Am2, Bdim2, C3, Dm3, Em3, F3, G3, Am3, Bdim3 and C4. We used three levels of dynamics: f (forte), mf (mesoforte), p (piano). For each combination, we produced 10 different recordings, producing a total of 450 data examples. This dataset can be downloaded from the github repository of the project[2].

Input: MFCC representation. Instead of using the raw audio samples as input to the network, we decided to use an MFCC representation, which has proven to be very useful for convolutional networks designed for audio content generation [5]. In consequence, the input to the network is a spectral representation of a 4-second window of an audio signal, by means of the MFCC transform. The calculation of MFCC is done by computing a short-time Fourier Transform (STFT) of each audio window, using a 512 stride and a 2048 window size, obtaining an image of size (128,1024,2). Magnitude and unwrapped phase are coded in different channels of the image.

Figure 2 displays the MFCC transform of a 4-second audio recording of a piano chord performed *forte*. Magnitude is shown on top while unwrapped phase is displayed at the bottom. The network outputs a MFCC audio representation as well. Figure 3 displays the MFCC representation of a 4-second audio recording of a the same forte chord of figure 2 but in this case, the chord was generated by the network by sampling the same position in the latent space where the original chord lays.

Model training. We used tensorflow 2.0 to implement our model. For training, we split our dataset leaving 400 examples for train-validation, and 50 examples for testing. We used an Adam optimizer with default parameters and learning rate of 3×10^{-5}. We chose a batch size of 5, and the training was performed for 500 epochs, the full training was done in about 6 hours using one GPU, a nvidia GTX 1080Ti. We used the standard cost function in VAE networks that has one term corresponding to the reconstruction loss and a second term corresponding to the KL divergence loss, but in practice the model was trained to maximize the ELBO (Evidence Lower BOund) [10, 15]. We tested different β weights for the KL term to find out how it does affects the clustering of the latent space [8]. The best results were obtained with $\beta = 1$.

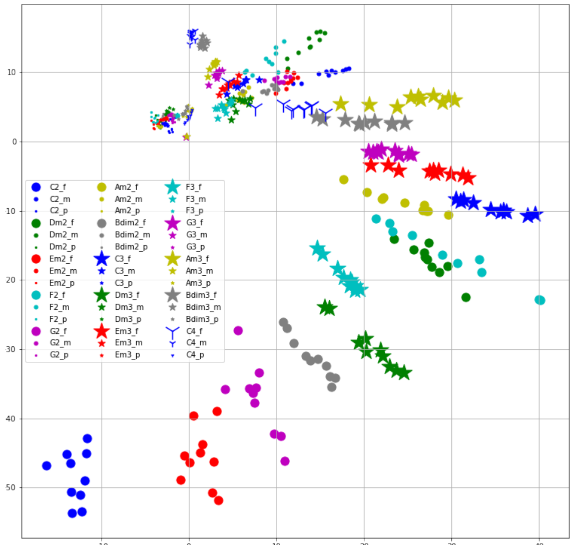

Figure 4: Two dimensional latent space representation of the dataset. Chords are arranged in a spiral pattern, and chords are arranged from forte to a piano dynamic.

5 USE CASES

Latent space exploration. Figure 4 displays a two dimensional latent space generated by the network. Chords are arranged in a spiral pattern following dynamics and octave position. Louder chords are positioned in the outer tail of the spiral while softer sound are in close proximity to the center. Chords are also arranged by octaves, lower octaves are towards the outer tail while softer octaves tend to be closer to the center. In this two dimensional space, the x coordinate seems to be related mainly to chroma, i.e. different chords, while the y coordinate is dominated by octave from lower to higher and dynamics from louder to softer. A remarkable property of this latent space is that different chords are arranged by thirds, following the pattern C, E, G, B, D, F, A. This means that neighboring chords share the largest number of common pitches. In general, this latent space is able to separate type of chords.

Chord generation. One of the nice properties of latent spaces is the ability to generate new chords by selecting positions in the plane that have not been previously trained by the network. In figure 5 we show the MFCC coefficients of a completely new chord generated by the network.

Chord sequencer. Another creative feature of our network is the exploration of the latent space with predefined trajectories, which allows for the generation of sequence of chords, resulting in a certain harmonic space. These trajectories not only encompass

[2] https://github.com/CreativAI-UC/TimbreNet/tree/master/datasets/

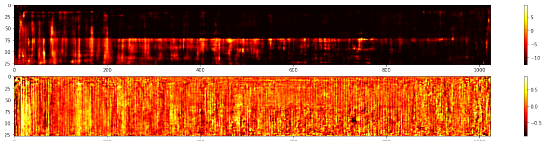

Figure 5: MFCC of a new chord generated by the network

different chord chromas, but different dynamics and octaves as well. In figure 6, one possible trajectory is shown. In this case, we can navigate from piano to forte, and from the thirds octave to the first, and at the same time we can produce different chords, following a desired pattern.

6 CONCLUSIONS AND FUTURE WORK

We have constructed *Latent Chords*, a VAE that generates chords and chord sequences performed at different level of dynamics and in different octaves. We were able to represent the dataset in a very compact two-dimensional latent space where chords are clearly clustered based on chroma, and where the axes correlate by octave and dynamic level. Contrary to many previous works reported in the literature, we used audio recordings of piano chords with musically meaningful variations such as dynamic level and octave positioning. We presented two use cases and we have shared our dataset, sound examples and network architecture to the community.

We would like to extend our work to a larger dataset, including new chords chromas, more levels of dynamics, more octave variation and include different articulations. We would also like to explore the design of another neural network devoted to explore the latent space in musically meaningful ways. This would allow us to generate a richer variety of chord music and to customize trajectories according to the desires and goals of each composer. We will also attempt to build an interactive tool such as Moodplay [1] to allow user exploratory search on a latent music space, but with added generative functionality.

ACKNOWLEDGMENTS

This research was funded by the *Dirección de Artes y Cultura, Vicerrectoría de Investigación* from the *Pontificia Universidad Católica de Chile*. Also, this work is partially funded by Fondecyt grants #1161328 and #1191791 ANID, Government of Chile, as well as by the Millenium Institute Foundational Research on Data (IMFD).

REFERENCES

[1] Ivana Andjelkovic, Denis Parra, and John O'Donovan. 2016. Moodplay: Interactive mood-based music discovery and recommendation. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*. ACM, 275–279.

[2] Jean-Pierre Briot, Gaëtan Hadjeres, and François-David Pachet. 2019. Deep learning techniques for music generation. *Sorbonne Université, UPMC Univ Paris* 6 (2019).

[3] Jun-qi Deng and Yu-Kwong Kwok. 2016. A Hybrid Gaussian-HMM-Deep Learning Approach for Automatic Chord Estimation with Very Large Vocabulary.. In *ISMIR*. 812–818.

[4] Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang. 2018. MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[5] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. 2017. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1068–1077.

[6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

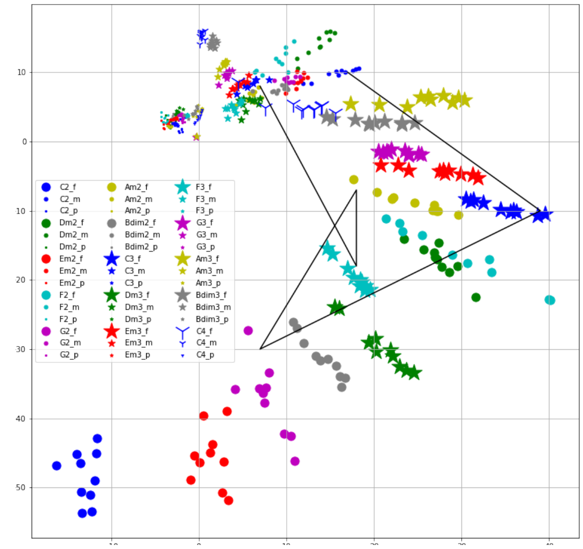

Figure 6: One possible trajectory for an exploration of the latent space. Trajectories consist on different chords, but also on different octaves and dynamics.

[7] Gaëtan Hadjeres, François Pachet, and Frank Nielsen. 2017. Deepbach: a steerable model for bach chorales generation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1362–1371.

[8] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. 2017. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *ICLR* 2, 5 (2017), 6.

[9] Eric J Humphrey, Taemin Cho, and Juan P Bello. 2012. Learning a robust tonnetz-space transform for automatic chord recognition. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 453–456.

[10] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[11] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. 2017. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215* (2017).

[12] Filip Korzeniowski and Gerhard Widmer. 2016. Feature learning for chord recognition: The deep chroma extractor. *arXiv preprint arXiv:1612.05065* (2016).

[13] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. 2018. Which training methods for GANs do actually converge? *arXiv preprint arXiv:1801.04406* (2018).

[14] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* (2016).

[15] Rajesh Ranganath, Sean Gerrish, and David Blei. 2014. Black box variational inference. In *Artificial Intelligence and Statistics*. 814–822.

[16] Adam Roberts, Jesse Engel, and Douglas Eck. 2017. Hierarchical variational autoencoders for music. In *NIPS Workshop on Machine Learning for Creativity and Design*.

[17] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, and Douglas Eck. 2018. A hierarchical latent vector model for learning long-term structure in music. *arXiv preprint arXiv:1803.05428* (2018).

[18] Bob L Sturm, Joao Felipe Santos, Oded Ben-Tal, and Iryna Korshunova. 2016. Music transcription modelling and composition using deep learning. *arXiv preprint arXiv:1604.08723* (2016).

[19] Aline Weber, Lucas Nunes Alegre, Jim Torresen, and Bruno C. da Silva. 2019. Parameterized Melody Generation with Autoencoders and Temporally-Consistent Noise. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Marcelo Queiroz and Anna Xambó Sedó (Eds.). UFRGS, Porto Alegre, Brazil, 174–179.

[20] Ivan P Yamshchikov and Alexey Tikhonov. 2017. Music generation with variational recurrent autoencoder supported by history. *arXiv preprint arXiv:1705.05458* (2017).

[21] Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. 2017. MidiNet: A convolutional generative adversarial network for symbolic-domain music generation. *arXiv preprint arXiv:1703.10847* (2017).

[22] Xinquan Zhou and Alexander Lerch. 2015. Chord detection using deep learning. In *Proceedings of the 16th ISMIR Conference*, Vol. 53.