Listen to Your Data: Turning Chemical Dynamics Simulations into Music

Austin Atsango[1] (atsango), Soren Holm[1] (sorenh), and K. Grace Johnson[1] (kgjohn)

[1] Department of Chemistry, Stanford University

Abstract

Our goal is to translate simulation data into a musical form in order to present a different way to interact with data. Specifically, the goals are 1) to **generate music**, i.e. melodies that are indistinguishable from those composed by humans, and 2) to have those melodies **reflect trends in the underlying data**.

We take two approaches: 1) We use a supervised model (either **softmax regression** or an **LSTM RNN** trained on composed melodies) to predict the next note in a song, biased by the trajectory values. 2) We cluster snippets of a trajectory using a Gaussian Mixture Model (**GMM**) with the EM algorithm to discover motifs within a trajectory, then match these motifs to similar ones from a composed melody.

We evaluate the success of these approaches with a survey designed to assess the two goals of the project.

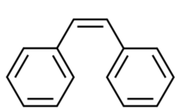
Datasets

Music

- MIDI data format
- 312 classical piano pieces
- 93 piano pieces from Final Fantasy video game
- Simplified using music21 and mido packages in Python to represent as pitch (with value 0 to 127) vs time

Most piano pieces have melodies in pitch range 50-90

Chemical dynamics



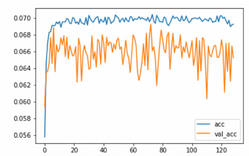
(Z)-Stilbene

- Quantum dynamics simulations of stilbene decaying from excited to ground state
- 200 trajectories of potential energy vs. time (femtoseconds)
- Potential energy normalized to 50-90 pitch range

Models

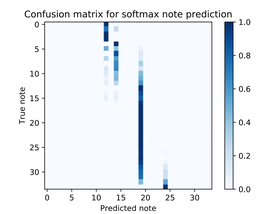
Predictive models

- Convolution over each musical piece
- One-hot encoding
- Supervised: predict next note based on previous 50 notes

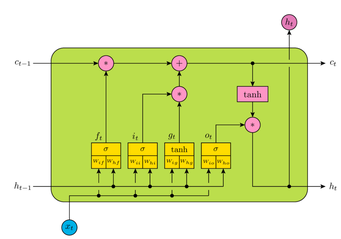


Softmax Regression

$$l(\theta) = \sum_{i=1}^{n} \log \prod_{l=1}^{k} \left(\frac{\exp(\theta_l^\top x^{(i)})}{\sum_{j=1}^{k} \exp(\theta_j^\top x^{(i)})} \right)^{1\{y^{(i)}=l\}}$$



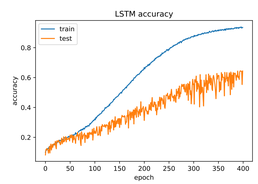
LSTM RNN

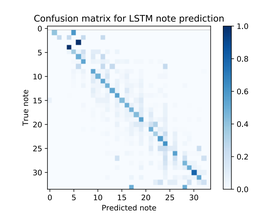


$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_C)$$
$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t \times \tanh(C_t)$$

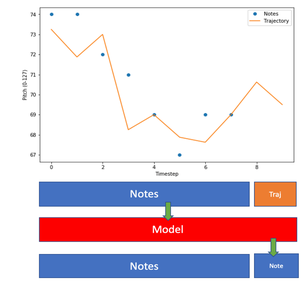
Architecture:

- LSTM with 256 hidden units
- LSTM with 38 hidden units
- Dense layer with softmax activation



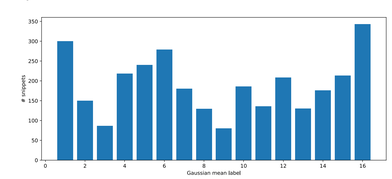


Music generation from trajectory:



GMM

- use a Gaussian Mixture Model with the EM algorithm to cluster snippets of all trajectories based on distance and gradient
- Match snippets to motifs in a given musical piece



Population of trajectory snippets for EM with 16 GMM means

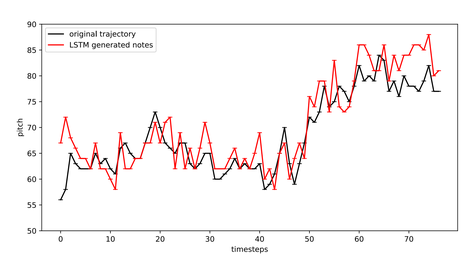
Results

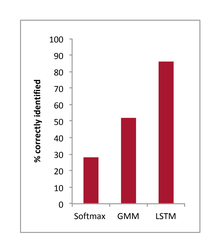
Goal 1: Turing test

Generated	
LSTM full	56%
LSTM subset	64%
Generated based on trajectory	
Softmax regression	13%
GMM	45%
LSTM subset	42%
Real music (control)	57%

Survey results with 40 participants showing % responding the sample was composed by a human.

Goal 2: matching generated music to trajectory





Percentage of participants matching correct trajectory

Discussion

We explored training predictive models with several architectures and on several subsets of the music data. We found the best training and validation accuracy using a subset of the full dataset: the pieces composed by Clementi.

Of all models tested, the LSTM RNN was most successful at generating music that reflected trends in a given dynamics trajectory. Softmax regression produced samples with the same note repeated, which were neither musical nor reflective of trajectory data. The GMM approach had roughly the same success as the LSTM, but cannot truly be considered music generation, as it sampled snippets from composed pieces. To more fully analyze the success of the models in achieving both goals outlined, we would need a survey with a much larger sample size both in number participants and number of audio clips.

Future Work

Future efforts include curating a larger dataset with distinctive melodies and exploring other generative models such as GANs or GRUs. The control of the Turing test shows that reducing a piece to simply pitch and time removes much of the musicality. We would also want to extend the model to train not just on pitch, but also on rhythm, chords, and other expressive information, then explore methods of interpreting the trajectory data with these additional features.

References

[1] Weir, H., Williams, M., Parrish, R., and Martinez, T.J. Nonadiabatic dynamics of photoexcited cis-Stilbene using ab initio multiple spawning. *In prep*. (2019).
[2] Classical piano midi page. Retrieved from http://www.piano-midi.de/
[3] Skli, Sigurur. How to Generate Music Using a LSTM Neural Network in Keras. Data Science, 7 Dec. 2017. Retrieved from towardsdatascience.com
[4] Holzner, A. LSTM Cells in Pytorch. Retrieved from https://medium.com/@andre.holzner/lstm-cells-in-pytorch-fab924a78b1c