Generative Lyric Composition via Transfer Learning

Jacob Wagner, Abraham L. Apellanes
Department of Computer Science
Stanford University
{jtwagner, aapellan}@stanford.edu

Overview

Motivation

Artists spend their careers honing their abilities to construct expressive, relatable, and creative lyrics. Our project explores modern deep learning techniques and the potential they have to supplement the songwriting process by learning from artists and their extensive work. We build on a neural network pre-trained on Kanye West's discography by tuning the model's hyper parameters with our own dataset of 4,000 songs from prominent hip-hop artists and rappers from the past 25 years.

Models

- Recurrent Neural Networks (RNNs)
- Long Short-Term Memory (LSTMs)

Results

Our experimental survey results show that participants could not discern our model's lyrics from the lyrics of trending SoundCloud rappers up to 20% of the time.

Datasets and Features

- MetroLyrics Dataset via Kaggle[3] [ID, Release Year, Artist, Genre, Lyrics]
- From the original MetroLyrics dataset, we found all the hip hop artists, and pared down to the top 35.
- We cleaned and standardized the entries in the dataset as it initially contained poorly structured and flawed entries.
- We placed a total of 4,000 songs into a list and wrote them to a text file for use in our model. This text file was a simple concatenation of the lyrics with no signification of the other features from the original dataset.
- Songs appear in the text file as follows: [*"Song 1 \\ Song 2 \\ Song 3 \\ ... \\ Song 4000"*].
- From this RNN model, there aren't many features extracted outside of the text lyrics and the input/output scheme because of the recurrent nature.

Results

```
But what.. Can I qualify for government grants? /
When it's drama I'ma dummy, look don't make lots of
sand, two blonds, implants / Yeah that my girl Shantz
/ Tell me I gotta talk rose petals on the name belts
```

Figure 3. Two bars generated by model with ~8% loss

Quiz #	Learning Rate	Decay	Epsilon	Rho	Max Syllables	Epochs	Batch size	Loss
6, 9	0.001	0	0	0.99	20	100	128	0.0519
8, 10	0.001	0	0	0	20	100	128	0.0525
4, 5	0.01314260933	0	0	0	16	10	64	0.0847
1, 2	0.008321415106	0	0	0	16	3	64	0.0845

Figure 4. Best performing model hyperparameters

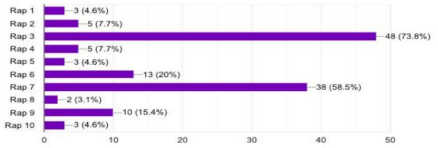

Figure 5. Distribution of ID experiment results where 3 and 7 are the true samples from trending artists on SoundCloud

- The training data that we used contained 4,000 songs split on newline characters.
- Models produce coherent sentences at 5-8% loss with some semblance of creativity (**Figure 3**).
- Hyperparameter tuning of (1) learning rate, (2) batch size, (3) number of epochs, (4) max syllables, (5) Rho (shown in **Figure 4**) yields a loss of ~5%.
- To qualitatively test our results, we surveyed 64 participants across campus. Participants were asked to rate ten randomly sampled lyrics, eight of which were AI-generated and two of which were from trending SoundCloud artists. They were then asked to identify which two were the natural ones. **Figure 5.** shows songs 3 and 7 were correctly ID'ed by 73.8% and 58.5% of participants respectively.

Models

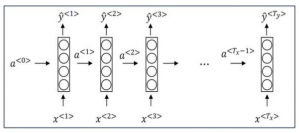

Figure 1. RNN Representation[4]

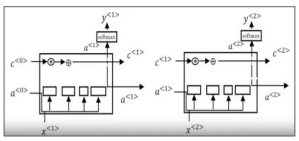

Figure 2. LSTM Representation[4]

We found a pretrained model[1] that uses an LSTM RNN architecture. This RNN model follows a recurrent mathematical formula that is very well detailed by Dr. Graves[2]. The hidden LSTM layers are used to create the predictions—refer to Graves for the formulas for RNNs and LSTMs[2]. We apply transfer learning using Barrat's model in order to train on and produce a more diverse set of lyrics. To do so, we feed the model a dataset that is 20X larger, comprising songs from prominent artists from the past 25 years. We then tuned hyperparameters such as: learning rate, decay, epsilon, Rho, max syllables per line, number of layers, epochs, and batch size. This gave us a large variation of losses and created a large range of lyrics. We applied the "Pandas" approach for tuning our hyperparameters based on initial successful metrics from our models.

Discussion

We tuned our model's learning rate, batch size, and number of layers. Our model plateaued at a point where it produced coherent lyrics but lacked contextual consistency in samples of more than 2-3 bars. We achieved a loss of 8% while the pretrained model we used was achieving 4%. We discovered that tuning the maximum number of syllables in a given line was key to further improvement; by going from 16 - 20 max syllables, we were able to achieve a loss of 5%. Tuning max syllables gave more room for the RNN to generate lyrics and gave a more generous cap that allowed for the network to better fit the data. Even with our relatively low loss, we find our model's performance to be less than optimal. Our experiment shows that people can still relatively easily discern our AI-generated lyrics from those of trending SoundCloud artists. As expected, feedback from participants indicates that our lyrics lack a thematic structure and a varied and creative flow.

Future Work

- Apply a fish approach to hyperparameter tuning.
- Work through the vanishing gradient problem.
- Experiment more with average line length of data and the max syllables per bar.
- Conduct different user trials.

References

[1] Barrat, R. (n.d.). Rapping Neural Network. Retrieved June 8, 2019, from https://github.com/robbiebarrat/rapping-neural-network
[2] Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.
[3] GyanendraMishra. (2017). 380,000 lyrics from MetroLyrics. Retrieved June 8, 2019.
[4] Andrew Ng slide deck C5M1