ERwEM: Events Represented with Emotive Music Using Topic-Filtered Tweets

Makayla Harris, Hunter Harris, Paul Bodily
Computer Science Department
Idaho State University
Pocatello, ID 83209 USA
{thommaka, harrhunt, bodipaul}@isu.edu

Abstract

We present ERwEM (Events Represented with Emotive Music), a system that generates cross-domain artefacts between text and music by drawing from emotional sentiments. Our system uses keywords extracted from a news article to filter live-stream tweets containing these keywords and analyze the tweet collection's emotion data. It then uses the emotion sentiment (a scale from positive to negative) to influence the mood of the music it writes using an adapted bi-axial long short term memory (BALSTM) model. From this, our system determines which mode the generated song will be transposed to. We use the FACE model to characterize the creative aspects of our system as a collection of generative acts. We provide examples of the artefacts our system generates as well as the response we received from external sources who reviewed the generated songs.

Introduction

Those in the field of computational creativity are consistently exploring new domains to apply CC and methods by which systems can produce creative artefacts. One such domain that continues to grow is music composition. In this domain, songwriters often draw on their emotional sentiment towards a concept or event to write music that reflects the way they feel about it. This is often reflected in the lyrics produced or the song's mode, which is the combination of its scale and tonal center. Our system explores this experience based method of songwriting by writing emotive songs that represent events or topics found in the news. Our system uses the these news articles and emotion values extracted from tweets to express emotion in its songs. This method for music creation led us to name our system ERwEM (Events Represented with Emotive Music).

In this paper, we leverage the correlation between modality and emotion to allow our system to generate emotive music. There is a strong correlation between the speed and modality of a song and the emotion that is attributed to the song (Fritz et al. 2009). This was found specifically for western style music and modes which are the same modes we employ in our system.

Related Works

There have been many forms of music generation ranging from statistical models (Conklin 2003) and Markov models (Schulze and Van Der Merwe 2010), to neural networks (Yang, Chou, and Yang 2017). There have also been studies on making emotive music from text, including the study on TransProse (Davis and Mohammad 2015). This study uses emotional data (a numerical measurement of the emotions found in text using the NRC Word-Emotion Association Lexicon) from literature to create music using parameters for pitch, tempo, and key; however, the music lacks a consistent vertical structure because it uses two independently written melodic lines played simultaneously. We intend to use this same emotional data to transform songs generated by a model able to create vertical structure in a song.

A study comparing the differences between neural networks and Markov model music generation (Cruz 2019), shows that neural networks generate music that more closely matches its training set than Markov models. A paper by Johnson describes one such neural network: a biaxial long short-term memory network (BALSTM) (Johnson 2017). There are other long short-term memory networks (LSTM) used for music generation (Sturm et al. 2019; Eck and Schmidhuber 2002; Boulanger-Lewandowsk et al. 2012), but the BALSTM is efficient in encoding both temporal and pitch patterns from music. Transposition invariance is a unique attribute of the BALSTM which allows training and generation in any key. It also allows for it to sample from a dataset containing songs with varying keys. Since MIDI files are often missing key information, this makes the key invariant model more convenient to use. Otherwise, key identification and transposition to a common key would be necessary for each song in the training data. We adapt Johnson's BALSTM as the music generation component of ERwEM.

The Nottingham Database[1] is a set of 1200 American and British folk songs. The MIDI files that make up this dataset are simple songs that consist of a melody played over chords. We found this dataset to generate song-like results more consistently and with fewer training iterations than other datasets we used. As a result, we use the parame-

[1] https://ifdo.ca/~seymour/nottingham/nottingham.html

Proceedings of the 11th International
Conference on Computational Creativity (ICCC'20)
ISBN: 978-989-54160-2-8

ters created from using this training dataset.

To extract emotion from text, we use the NRC Word-Emotion Association Lexicon (Mohammad and Turney 2013). Using this lexicon and Twitter, our system finds inspiration for the emotions it will use in its generation process.

There are several papers on how creativity can be measured and attributed to a system such as Ritchie's metrics (Ritchie 2007), and Colton's FACE model (Colton, Charnley, and Pease 2011) which we relate to our system. Ventura also describes a general structure for CC systems (Ventura 2017) as well as what it means for a system to be merely generative (Ventura 2016). He discusses the spectrum of stages for a creative system that we use to show where our system is located in the spectrum of creativity and what is necessary for higher levels of creativity.

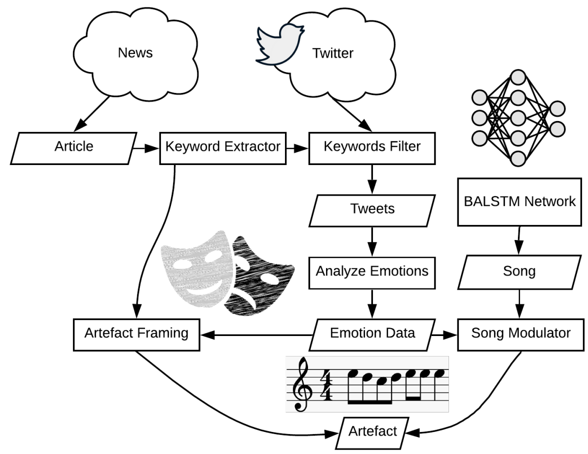

Figure 1: An overview of the ERwEM system. From The Guardian news API, we select one article. From that article we extract keywords and use the keywords to filter Twitter for live tweets containing those keywords. We use an emotion analyzer to extract the emotion data from the collected tweets. The emotion data is used in the artefact framing as well as in the song modulator. After the BALSTM generates a song, we transpose it to a new mode decided from the emotion data. This paired with the artefact framing gives us the artefact generated by ERwEM.

Methods

In this section we describe the design and construction of ERwEM. We first give an overview of our system and the task it is designed to accomplish. We then provide a detailed description of the components that make up our system and the processes they use to reach our goal.

ERwEM System Overview

ERwEM is designed to generate emotive music from events it finds from The Guardian news outlet. Our system begins by selecting a news article and extracting the keywords from it. Using those keywords, the system filters Twitter for live tweets that contain any of the keywords gathered from the news article. Emotions are gathered from the collected tweets using the NRC Word-Association Lexicon and mapped to a musical mode. From there, our system generates a new song using a BALSTM and transposes the song to the aforementioned mode. We trained the BALSTM on the Nottingham Database for its simplicity and consistent structure which generated the most songlike pieces. The complete artefact generated by the system consists of the final composition and its framing (see Figure 1).

To show the components of our system that contribute to its creativity, we analyze it using the FACE model. The FACE model defines the creative components of a system as a tuple of generative acts. We argue the tuple representing ERWEM to be $< F^g, C^g, E^g >$. These generative acts are briefly outlined below.

- F^g: Framing generated from news, tweets, and emotions
- C^g: Concepts created from extracted keywords and their perceived emotion
- E^g: Expression developed by transposing a generated song to a mode determined by emotion

Concept ERwEM generates concepts for its creative process from news and tweets. It begins by selecting a news article $a \in A$ where A is the set of all articles available in The Guardian's API. From a we extract n number of keywords k where the set of keywords for a given article is a_k. We arbitrarily chose $n = 3$ in our system. We extract the keywords from the article using a chi-square test for independence to find the words that are statistically significant. We use approximately twenty years worth of articles from The Guardian news outlet to build the statistical model. Using those keywords a_k we filter live tweets t from Twitter giving the set a_{k_t} (see Figure 2). We obtain the emotions ε from a_{k_t} by using the NRC Word-Emotion Association Lexicon where ε is a set of emotion and emotion score pairs (see Table 1). This process gives us the concept $c^g = (a, a_k, a_{k_t}, \varepsilon)$.

Expression The expression, or artefact, is generated by using a concept c^g. The system begins by composing a song s in the form of a MIDI file. Then, using the emotion data ε from the c^g, we calculate the overall brightness b of the expression as:

$$b = \frac{e_{pos} - e_{neg}}{e_{pos} + e_{neg}}$$

Where ε_{pos} is the count of words with the positive attribute marked and ε_{neg} is the count of words with the negative attribute marked. This gives b a range of $[-1, 1]$. Our system determines the mode m for the expression by mapping the set of the seven common modes M to the range $[-1, 1]$. This is accomplished by spacing the seven modes equally across the range such that the brightest mode is equal to 1 and the darkest mode is equal to -1 (see Figure 3). The order of these modes reflects the order described in music theory. Locrian is the darkest mode and each mode is subsequently brighter than the last such that lydian is the brightest

Proceedings of the 11th International
Conference on Computational Creativity (ICCC'20)
ISBN: 978-989-54160-2-8

Figure 2: An example tweet, collected by ERwEM.

Emotion	Value	Emotion	Value
Positive	1 (20%)	Negative	4 (80%)

Emotion	Value	Emotion	Value
Anger	0 (0%)	Anticipation	0 (0%)
Disgust	0 (0%)	Fear	2 (50%)
Joy	0 (0%)	Sadness	2 (50%)
Surprise	0 (0%)	Trust	0 (0%)

Table 1: An example of the emotional data extracted from the tweet pictured above. This was accomplished using the NRC Word-Emotion Association Lexicon which totals the emotion score for each word in the tweet. The relative distribution of these is represented in parantheses.

mode. Using our brightness and our scale of modes, we are able to select the mode m closest to our brightness b.

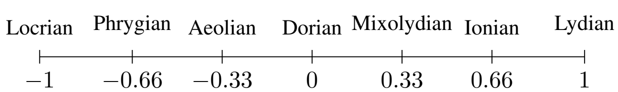

Figure 3: The mapping of the seven common modes to the range $[-1, 1]$ where the relative brightness of the mode increases from left to right (e.g., locrian=darkest, lydian=brightest).

We will denote the current mode of our song as s_{m_0} and the mode it is to be transposed to as s_m. To programmatically transpose $s_{m_0} \xrightarrow{to} s_m$, we created a series of vectors that will act as transformations from one mode to the next closest mode in descending order (see Table 2). Using this model, we are able to construct a transformation vector from any one mode to any other mode. As an example, we show the process for creating a transformation vector from ionian (major) to aeolian (minor). Consider the following equation.

$$\underset{\text{Io}\xrightarrow{to}\text{Mi}}{\begin{bmatrix}0\\0\\0\\0\\0\\0\\-1\end{bmatrix}} + \underset{\text{Mi}\xrightarrow{to}\text{Do}}{\begin{bmatrix}0\\0\\-1\\0\\0\\0\\0\end{bmatrix}} + \underset{\text{Do}\xrightarrow{to}\text{Ae}}{\begin{bmatrix}0\\0\\0\\0\\0\\-1\\0\end{bmatrix}} = \underset{\text{Io}\xrightarrow{to}\text{Ae}}{\begin{bmatrix}0\\0\\-1\\0\\0\\-1\\-1\end{bmatrix}}$$

By adding the transformation vectors for each successive transposition from ionian to aeolian, we are left with a new

Transposition from mode to mode		Scale Degree 1	2	3	4	5	6	7
Lydian$\xrightarrow{to}$Ionian	(Ly$\xrightarrow{to}$Io)	0	0	0	-1	0	0	0
Ionian$\xrightarrow{to}$Mixolydian	(Io$\xrightarrow{to}$Mi)	0	0	0	0	0	0	-1
Mixolydian$\xrightarrow{to}$Dorian	(Mi$\xrightarrow{to}$Do)	0	0	-1	0	0	0	0
Dorian$\xrightarrow{to}$Aeolian	(Do$\xrightarrow{to}$Ae)	0	0	0	0	0	-1	0
Aeolian$\xrightarrow{to}$Phrygian	(Ae$\xrightarrow{to}$Ph)	0	-1	0	0	0	0	0
Phrygian$\xrightarrow{to}$Locrian	(Ph$\xrightarrow{to}$Lo)	0	0	0	0	-1	0	0
Locrian$\xrightarrow{to}$Lydian	(Lo$\xrightarrow{to}$Ly)	0	1	1	1	1	1	1

Table 2: The transforms to transpose from one mode to another mode. Ordered top to bottom from brightest mode to darkest mode. Scale degree represents the seven notes that make up a heptatonic scale. The value of each scale degree for a given row represents the number of semitones to lower or raise each scale degree to transpose between the two modes labeled on the left.

transformation vector that will take any song in ionian and transpose it to aeolian (this process of transposing maintains the current key the song is in and only affects the mode). An example of this on the key of C in mode ionian is shown below.

$$\underset{\text{C Ionian}}{\begin{bmatrix}0\\2\\4\\5\\7\\9\\11\end{bmatrix}} + \underset{\text{Io}\xrightarrow{to}\text{Ae}}{\begin{bmatrix}0\\0\\-1\\0\\0\\-1\\-1\end{bmatrix}} = \underset{\text{C Aeolian}}{\begin{bmatrix}0\\2\\3\\5\\7\\8\\10\end{bmatrix}}$$

This process gives us our resulting expression $e^g = (m, s_m)$ where s_m is the transposed version of the originally generated song s. To tie our concept and expression together, our system proceeds to construct framing for the expression from its associated concept.

Framing Our system creates framing f^g for e^g by using the information in c^g. We create the title for the current expression e^g_{name} by joining the three keywords a_k with underscores, and we choose the top two emotions from ε (excluding positive and negative) to be ε_1 and ε_2. By using the article title a_{title}, the keywords a_k, the top two emotions ε_1 and ε_2, and the mode m, our system constructs the framing using the following template.

`While working on my project` e^g_{name}`, I found a news article titled: "`a_{title}`". I checked Twitter to see what other people had to say relating to the topics` a_k `that I found in the article. What I read from the tweets made me feel` ε_1 `and` ε_2`. As a result, I decided to write a song in` m

This framing f^g is displayed alongside the e^g to enhance the creativity of our system.

Figure 4: An example song generated by ErwEM.

Results

In this section, we will describe the results from building and testing our system as well as some of the ways we evaluated its artefacts.

ERwEM describes the context from which it gathered its inspiration as well as the decision it made from the context. This helps our system explain why and how it generated the songs that it does and enhances its creativity by acting as the F^g generative act. This helps people who listen to the songs connect with the process by which ERwEM created its artefacts and understand the inspiration behind them. We believe this can improve the perceived creativity of our system.

A snippet of a song generated by our system can be seen notated musically in Figure 4. Audio examples of the generated artefacts can be listened to on ERwEM's SoundCloud[2]. The resulting artefacts had a moderate level of variance, generating songs in the following modes: aeolian, dorian, mixolydian, and ionian. We are led to believe the reason our system has not written in the darkest and brightest modes is due to the rarity of finding extreme emotion values from a large sample of tweets. We have determined this not to be a flaw in our system because songs are rarely written by human composers in the darkest and brightest modes.

Conclusion

In our efforts to create a system that can express emotive music, we attempt to simulate the way songwriters incorporate personal and perceived experiences about concepts through our system's use of emotional analysis applied to music. Our system is currently in between Generalization and Filtration in Ventura's spectrum of generative systems. Our system could be attributed a higher level of creativity if we implement a fitness evaluation function for the works produced by our system. This is a candidate for future work with ERwEM.

Although our system struggles in creating songs that sound vastly different from each other, we argue this is something that human musicians also struggle to do. We recognize the limited dataset and training time to be limitations of our system and propose more advanced methods of training and generation for future work. We also note the BALSTM model creates music with vertical structure, but does not have horizontal structure. We recognise this as a limitation of our system and would explore different models of music generation that have greater capabilities in creating horizontal structure throughout songs. We would also use the emotion data to influence tempo, instrumentation, pitch, etc., in its song generation.

[2]`https://soundcloud.com/anonymized`

References

Boulanger-Lewandowsk, N.; YoshuaBengio, P. V.; Gray, P.; and Naguri, C. 2012. Modeling temporal dependencies in high-dimensional sequences. *ICML*.

Colton, S.; Charnley, J.; and Pease, A. 2011. Computational creativity theory: The FACE and IDEA descriptive models. *Proceedings of the 2nd International Conference on Computational Creativity, ICCC 2011* 90–95.

Conklin, D. 2003. Music generation from statistical models. In *Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*, 30–35. Citeseer.

Cruz, J. 2019. Deep learning vs markov model in music generation. Undergraduate honors thesis, Pace University.

Davis, H., and Mohammad, S. 2015. Generating Music from Literature. 1–10.

Eck, D., and Schmidhuber, J. 2002. A first look at music composition using lstm recurrent neural networks. *Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale* 103:48.

Fritz, T.; Jentschke, S.; Gosselin, N.; Sammler, D.; Peretz, I.; Turner, R.; Friederici, A. D.; and Koelsch, S. 2009. Universal recognition of three basic emotions in music. *Current Biology* 19(7):573 – 576.

Johnson, D. D. 2017. Generating polyphonic music using tied parallel networks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10198 LNCS:128–143.

Mohammad, S. M., and Turney, P. D. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence* 29(3):436–465.

Ritchie, G. 2007. Some empirical criteria for attributing creativity to a computer program. *Minds and Machines* 17(1):67–99.

Schulze, W., and Van Der Merwe, B. 2010. Music generation with markov models. *IEEE MultiMedia* (3):78–85.

Sturm, B. L.; Ben-Tal, O.; Monaghan, U.; Collins, N.; Herremans, D.; Chew, E.; Hadjeres, G.; Deruty, E.; and Pachet, F. 2019. Machine learning research that matters for music creation: A case study. *Journal of New Music Research* 48(1):36–55. Publisher: Routledge _eprint: https://doi.org/10.1080/09298215.2018.1515233.

Ventura, D. 2016. Mere generation: Essential barometer or dated concept? In Pachet, F.; Cardoso, A.; Corruble, V.; and Ghedini, F., eds., *Proceedings of the Seventh International Conference on Computational Creativity, UPMC, Paris, France, June 27 - July 1, 2016*, 17–24. Sony CSL Paris, France.

Ventura, D. 2017. How to Build a CC System. *Eighth International Conference on Computational Creativity (ICCC'17)* 253–260.

Yang, L.-C.; Chou, S.-Y.; and Yang, Y.-H. 2017. Midinet: A convolutional generative adversarial network for symbolic-domain music generation. *arXiv preprint arXiv:1703.10847*.