ScripTONES: Sentiment-Conditioned Music Generation for Movie Scripts

Vishruth Veerendranath, Vibha Masti, Utkarsh Gupta, Hrishit Chaudhuri, Gowri Srinivasa
PES Center for Pattern Recognition, Department of Computer Science and Engineering
PES University
`{vishruthnath, vsmasti, utkarsh348, hrishitchaudhuri}`@gmail.com,
gsrinivasa@pes.edu

Abstract

Film scores are considered an essential part of the film cinematic experience, but the process of film score generation is often expensive and infeasible for small-scale creators. Automating the process of film score composition would provide useful starting points for music in small projects. In this paper, we propose a two-stage pipeline for generating music from a movie script. The first phase is the Sentiment Analysis phase where the sentiment of a scene from the film script is encoded into the valence-arousal continuous space. The second phase is the Conditional Music Generation phase which takes as input the valence-arousal vector and conditionally generates piano MIDI music to match the sentiment. We study the efficacy of various music generation architectures by performing a qualitative user survey and propose methods to improve sentiment-conditioning in VAE architectures.

1 Introduction

Film scores are a critical part of the cinematic experience and in high-production environments, they require a significant amount of skill and time to produce. The challenge lies in being able to correctly identify abstract musical themes and motifs. The current state of musical score composition relies on human knowledge of these properties, with the pipeline often starting with the composers seeing a rough cut of the script, analyzing the mood of the scene, and applying this knowledge of motifs to compose music that matches the sentiment of the scene. The music is then ornamented with instrumentalization, and the score is edited to match the film shots.

In this paper, we present **ScripTONES** (**S**cript **TO N**otes with **E**motional **S**ignals), an automated two-stage pipeline to conditionally generate music for movie scenes. Unlike [34], the goal of ScripTONES is to generate music for a short movie scene, and not to retrieve music to fit the sentiment of a book chapter. The first stage of the pipeline is the *Sentiment Analysis* phase where the sentiment of the movie scene is extracted from the script text. After preprocessing the scene text, we capture its sentiment by extracting its Valence and Arousal [30] values according to the NRC VAD lexicon [21]. The second stage is conditional music generation. Here, we generate polyphonic piano MIDI music to match the valence and arousal of the scene extracted in the first stage. We explore the current transformer-based [16, 13] and VAE-based [32, 39, 18] methodologies that allow for sentiment-conditioned music generation. To allow for finer control from the creator in inspiring the music generated, we propose regularisation losses for sentiment modification. We also compute attribute vectors for each quadrant of the Valence-Arousal space (henceforth VA space) and perform attribute vector arithmetic to modify the sentiment of a music piece. The VAE would also allow for interpolation to match the changes in the sentiment of a scene. A detailed discussion of related work and data used can be found in Appendices D and E respectively.

Figure 1: Illustration of the ScripTONES pipeline, which consists of 2 major stages - Sentiment Analysis of Scripts and Conditional Music Generation. Music Generation is achieved either with MusicVAE with attribute vector arithmetic or EMOPIA-CWT

2 ScripTONES Pipeline

This section describes the pipeline of score generation process which has been illustrated in Fig. 1.

2.1 Sentiment Analysis of Scripts

A movie script contains dialogue as well as additional information that is used to produce and direct the movie. We parse the script to only extract meaningful words that contribute to the sentiment of the scene, as well as segment a script into individual scenes. For this, we use the ScreenPy parser [40] which relies on formal grammar rules that conform to the *Hollywood Standard format*.

For sentiment analysis, we use the NRC VAD [21] lexicon to find the *unweighted average* of valence and arousal values from every single word in the parsed movie scene. The value obtained is in the range [0, 1], and further normalized to be in the range [-1, 1]. The movie scene S_i is thus associated with a Valence-Arousal (VA) tuple value $VA_i = (V, A)$, where $-1 \leq V, A \leq 1$.

2.2 Conditional Music Generation

Below we describe the two models used in the pipeline. We propose and experiment with ways (regularization losses) to improve conditional generation with VAEs in Appendix A and C.1.

2.2.1 EMOPIA-CWT (Transformer-based)

We experimented with the Compound Word Transformer (CWT) trained on EMOPIA [16] for music generation (henceforth referred to as **EMOPIA-CWT**). This phase is labelled 2.b in Fig. 1. Here, sentiment is not represented as a continuous (V, A) tuple, but as a quadrant number Q of the valence-arousal space in the range $[1, 4]$. The EMOPIA-CWT generates music that fits the sentiment of the corresponding quadrant. While the discrete quadrant method does not allow for latent space sampling, the music generated is quite realistic, polyphonic and fits the sentiment of movie scenes rather well.

2.2.2 MusicVAE tuned with Attribute Vector Arithmetic

Inspired by [32], we extend the idea of attribute vector arithmetic in VAEs to the sentiment attributes of valence and arousal in music. To formalize this, we define 4 emotional attribute vectors, namely High Valence (z_{vh}), Low Valence (z_{vl}), High Arousal (z_{ah}) and Low Arousal (z_{al}) by averaging latent vectors z (encoded using MusicVAE [32]) of all MIDI samples in the EMOPIA dataset [16]. To conditionally generate the music, we scale the attribute vectors based on threshold value α defined on the V & A values extracted from the script (empirically α = -0.25), as per Eqn. 1. This results in an Emotionally Conditioned latent vector z_{ec}, which is then decoded back to music as a MIDI file. This methodology enables conditioning music in a continuous space.

2

$$z_{ec} = \begin{cases} |V| * z_{vh} + |A| * z_{ah} & (V \geq 0, A \geq \alpha) \\ |V| * z_{vh} + |A| * z_{al} & (V \geq 0, A < \alpha) \\ |V| * z_{vl} + |A| * z_{ah} & (V < 0, A \geq \alpha) \\ |V| * z_{vl} + |A| * z_{al} & (V < 0, A < \alpha) \end{cases} \tag{1}$$

We introduce stochasticity by randomly sampling a point in the latent space (z_r) and tuning the random point based on our emotional condition vector as per $z_{tuned} = z_r + z_{ec}$. We further enable finer control for artists by replacing z_r with the encoding for artist's inspiration music piece (z_{insp}).

3 Evaluation and Results

Additional experiments are detailed in Appendix C. Demonstrations & samples are on our webpage [1].

3.1 Attribute Vector Arithmetic

To demonstrate the validity of the attribute vector arithmetic, we show an example of conditional music generation with the pre-trained MusicVAE [32] model with attrbite vector arithmetic. A sample from the EMOPIA dataset [16] (Fig 2(a)) has been passed in as an inspiration piece and is encoded to latent vector z. The emotional characteristics of this piece have then been modified using the High Valence (z_{vh}) and High Arousal (z_{ah}) attribute vectors. When z_{vh} is added to z and then converted back to MIDI, the density of notes, as seen in Fig 2(b), is much higher than the original. Similarly, adding z_{ah} to z and converting back to MIDI (Fig 2(c)) causes the music to follow a similar note sequence as the original inspiration music, but intersperses it with some fast-paced staccato notes between the original note structure.

(a) Original music   (b) Modified with increased valence   (c) Modified with increased arousal
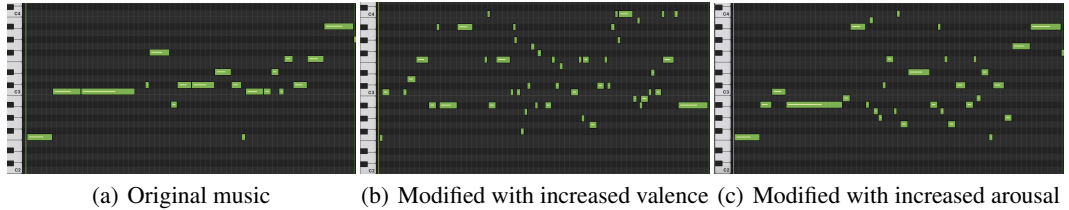
Figure 2: Original music (a) modified with Increased Valence (b) and Increased Arousal (c)

(a) Discrete Latent Regularization   (b) Regularization loss during finetuning
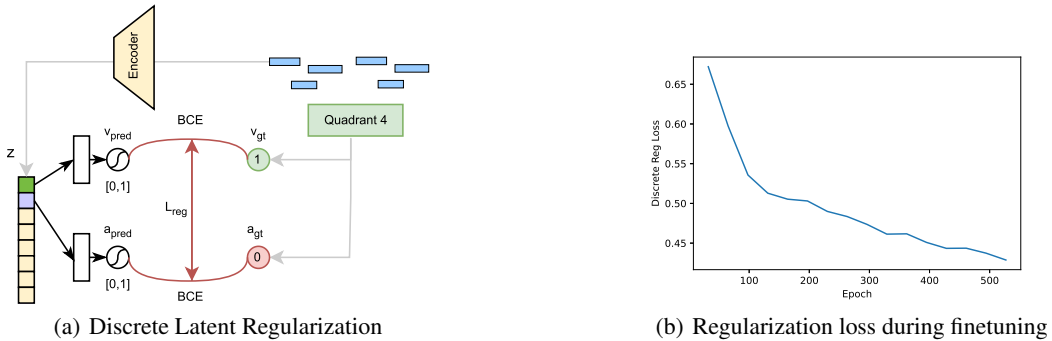
Figure 3: Discrete Regularization and loss plot while finetuning FIGARO on EMOPIA data

3.2 User Study Evaluation of ScripTONES

A group of 31 users (with varying music knowledge as seen in Fig. 4(b)) were surveyed to compare the quality of music generated by the different models for 3 movie scenes. For each scene, two different

---

[1]bit.ly/scriptones-script-music

Table 1: Average ratings of match between generated music and film scene

Attribute Rated	E-CWT	MVAE
Valence	**2.62**	1.96
Arousal	**2.44**	1.92
Overall Mood Fit	**2.48**	1.86

Table 2: User evaluated scene-wise overall mood fit ratings on a scale of 1-4

Scene Number	E-CWT	MVAE
Scene 1	**2.52**	1.58
Scene 2	1.87	**2.17**
Scene 3	**3.06**	1.84

music pieces were played – one generated by MusicVAE and the other generated by EMOPIA-CWT as the music generators in the ScripTONES pipeline. For each piece of music, on a scale of 1-4, we rate the valence/positivity, the arousal/excitement and the overall mood fit of the music with respect to the mood of the scene.
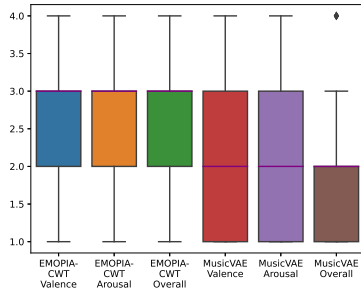
(a) Box-plot of user ratings for E-CWT & MVAE models
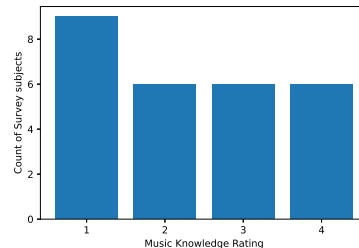
(b) Music Knowledge rating of survey subjects

Figure 4: User preferences of music generation model and their music knowledge

Table 1 presents the average valence, arousal and overall mood fit ratings (on a 4-point Likert scale), across all users and all scenes in the study. A detailed box-plot of the ratings are shown in Fig. 4(a). The EMOPIA-CWT model receives a better rating across all attributes. This can be attributed to the compound word architecture of EMOPIA-CWT, which can generate more coherent and nuanced music with polyphony.

We analyse the scene-wise consensus (details in Appendix B) among users by averaging the overall mood fit ratings for each scene across all users in Table 2. We find that user's prefer EMOPIA-CWT in scenes that require nuanced and sophisticated music with polyphony whereas MusicVAE is preferred in negative and action scenes.

3.3 Discrete Regularization

To improve sentiment-conditioning further in VAEs we propose a Discrete Regularization loss as per Eqn. 2. An illustration can be seen in Fig. 3(a) and the regularization process is described in more detail in Appendix A.2.2. We use this loss to finetune FIGARO's [38] pretrained VAE model, on the EMOPIA dataset [16]. This helps FIGARO's VAE model learn a better latent space representation for sentiment as seen in Fig. 3(b) and detailed further in Appendix C.1.

$$L_{reg_{disc}} = BCE(v_{pred}, v_{gt}) + BCE(a_{pred}, a_{gt}) \tag{2}$$

4 Conclusion and Future Work

In this paper, we present ScripTONES, which introduces new directions for research on sentiment-conditioned music generation for movie scripts. We obtain the sentiment encoding of movie scripts using the NRC VAD lexicon for English words and use it to conditionally generate music using both VAE and Transformer architectures, which we qualitatively evaluate by surveying users. We present the ideas of attribute vector arithmetic and latent regularisation for sentiment modifications in the latent space of the VAE. In future work, variations of encoder-decoder architectures (GANs, GRU-VAEs) can be explored to improve the music generation capabilities of the model.

References

[1] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

[2] Dmitry Bogdanov, Xavier Lizarraga Seijas, Pablo Alonso-Jiménez, and Xavier Serra. Musav: a dataset of relative arousal-valence annotations for validation of audio models. 2022.

[3] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

[4] Sven Buechel and Udo Hahn. EmoBank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 578–585, Valencia, Spain, April 2017. Association for Computational Linguistics.

[5] Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. Goemotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, 2020.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[7] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[8] Paul Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.

[9] Jesse Engel, Matthew Hoffman, and Adam Roberts. Latent constraints: Learning to generate conditionally from unconditional generative models. *arXiv preprint arXiv:1711.05772*, 2017.

[10] Lucas N Ferreira and Jim Whitehead. Learning to generate music with sentiment. *arXiv preprint arXiv:2103.06125*, 2021.

[11] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

[12] Wen-Yi Hsiao, Jen-Yu Liu, Yin-Cheng Yeh, and Yi-Hsuan Yang. Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 178–186, 2021.

[13] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer. *arXiv preprint arXiv:1809.04281*, 2018.

[14] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 709–727. Springer, 2020.

[15] Yu-Siang Huang and Yi-Hsuan Yang. Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2020.

[16] Hsiao-Tzu Hung, Joann Ching, Seungheon Doh, Nabin Kim, Juhan Nam, and Yi-Hsuan Yang. Emopia: A multi-modal pop piano dataset for emotion recognition and emotion-based music generation. *arXiv preprint arXiv:2108.01374*, 2021.

[17] IMSDb. The internet movie script database (imsdb).

[18] Junyan Jiang, Gus G Xia, Dave B Carlton, Chris N Anderson, and Ryan H Miyakawa. Transformer vae: A hierarchical model for structure-aware and interpretable music representation learning. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 516–520. IEEE, 2020.

[19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[20] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150, 2011.

[21] Saif M. Mohammad. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Proceedings of The Annual Conference of the Association for Computational Linguistics (ACL)*, Melbourne, Australia, 2018.

[22] Saif M Mohammad and Peter D Turney. Crowdsourcing a word–emotion association lexicon. *Computational intelligence*, 29(3):436–465, 2013.

[23] Surabhi S Nath, Vishaal Udandarao, and Jainendra Shukla. It's levasa not leviosa! latent encodings for valence-arousal structure alignment. In *Proceedings of the 3rd ACM India Joint International Conference on Data Science & Management of Data (8th ACM IKDD CODS & 26th COMAD)*, pages 238–242, 2021.

[24] Fabien Chhel Amal El Fallah Seghrouchni Nicolas Gutowski Nathan Fradet, Jean-Pierre Briot. Miditok: A python package for midi file tokenization. In *Extended Abstracts for the Late-Breaking Demo Session of the 22nd International Society for Music Information Retrieval Conference*, 2021.

[25] Pedro Neves, Jose Fornari, and João Florindo. Generating music with sentiment using transformer-gans. *arXiv preprint arXiv:2212.11134*, 2022.

[26] Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 188–197, 2019.

[27] Sageev Oore, Ian Simon, Sander Dieleman, Douglas Eck, and Karen Simonyan. This time with feeling: Learning expressive musical performance. *Neural Computing and Applications*, 2018.

[28] Ashis Pati and Alexander Lerch. Attribute-based regularization of latent spaces for variational auto-encoders. *Neural Computing and Applications*, 33:4429–4444, 2021.

[29] Robert Plutchik. *The emotions*. University Press of America, 1991.

[30] Jonathan Posner, James A Russell, and Bradley S Peterson. The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and psychopathology*, 17(3):715–734, 2005.

[31] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

[32] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, and Douglas Eck. A hierarchical latent vector model for learning long-term structure in music. In *International conference on machine learning*, pages 4364–4373. PMLR, 2018.

[33] Klaus R Scherer and Harald G Wallbott. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310, 1994.

[34] Jaidev Shriram, Makarand Tapaswi, and Vinoo Alluri. Sonus texere! automated dense sound-track construction for books using movie adaptations.

[35] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[36] Hao Hao Tan and Dorien Herremans. Music fadernets: Controllable music generation based on high-level features via low-level feature modelling. *arXiv preprint arXiv:2007.15474*, 2020.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[38] Dimitri von Rütte, Luca Biggio, Yannic Kilcher, and Thomas Hofmann. Figaro: Generating symbolic music with fine-grained artistic control. *arXiv preprint arXiv:2201.10936*, 2022.

[39] Ziyu Wang, Yiyi Zhang, Yixiao Zhang, Junyan Jiang, Ruihan Yang, Junbo Zhao, and Gus Xia. Pianotree vae: Structured representation learning for polyphonic music. *arXiv preprint arXiv:2008.07118*, 2020.

[40] David R Winer and R Michael Young. Automated screenplay annotation for extracting story-telling knowledge. In *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2017.

[41] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

[42] Yelp. Yelp open dataset.

[43] Mingliang Zeng, Xu Tan, Rui Wang, Zeqian Ju, Tao Qin, and Tie-Yan Liu. Musicbert: Symbolic music understanding with large-scale pre-training, 2021.

A   Improving Sentiment-Conditioning of Music with VAEs

We propose a few methods to improve the affective characteristics of the music generated by a variational autoencoder (VAE) to match the sentiment of the script more closely. These methods are enabled by three properties of VAEs, which we describe below.

A.1   Attribute Vector Arithmetic

As described in Section 2.2.2, we can extract the characteristics for each type of sentiment in the embedding space and use it to tune the music generated. While we demonstrate the use of attribute vector arithmetic with the four quadrants of the valence-arousal space as our distinct sentiments, this can also be used for other *distinct sentiment models* like positive-negative (2 categories), Plutchik's wheel of emotions 8 categories) [29] and Ekman's model (6 categories) [8]. However, we cannot extract attribute vectors for all sentiments in a continuous sentiment model.

A.2   Latent Space Regularization

To handle continuous sentiment models, we consider the work done by Pati et al. [28], which extended the ideas of latent space regularization [9] and disentanglement [11] for music. The regularization loss is capable of constraining a particular dimension of the latent space to reflect monotonicity in a particular attribute of the data. Regularizing one dimension of the latent space each for the attributes of valence and arousal will enable direct control of valence and arousal by simply increasing or decreasing the corresponding dimension. This is much like the faders described in [36] but for high-level sentiment attributes.

We propose regularizing the latent space in the following two ways.

A.2.1 Continuous Regularization

The first, which we call *Continuous Regularization* is closely related to [28] where we define regularization losses for valence and arousal as per the equations

$$L_{reg_v} = MSE(tanh(D_{z^1}) - sgn(D_v)) \tag{3}$$

$$L_{reg_a} = MSE(tanh(D_{z^2}) - sgn(D_a)) \tag{4}$$

and $L_{reg_{cont}} = L_{reg_v} + L_{reg_a}$, where D_{z^1} and D_{z^2} are the sum of differences in z^1 (1st dimension for valence) and z^2 (2nd dimension for arousal) respectively, computed pairwise over all pairs of data points in the mini-batch. D_v and D_a are the *attribute distances* i.e. the sum of differences in the valence and arousal values of the pairs of data points. $sgn(.)$ refers to the sign (positive or negative).

Due to the unavailability of a dataset of music annotated with continuous valence and arousal values, computation of the *attribute distances* is not directly possible. We suggest the use of the valence-arousal regressors introduced in MusAV [2] to predict the valence and arousal values of each piece of music, thereby enabling the computation of D_a and D_v. However, the valence-arousal predictions from the regressors can only be considered noisy labels (true labels in [2] are relative valence-arousal and not absolute values) and might lead to inaccurate regularization.

A.2.2 Discrete Regularization

To use true labels instead of noisy predictions as labels for regularization, we propose a second method for regularization called *Discrete Regularization*, inspired by [23]. Here, we use the discrete VA quadrant annotations in the EMOPIA dataset [16] as an approximation for the continuous VA values.

Here again, we regularize z^1 for valence and z^2 for arousal. This is done by passing the z^1 and z^2 through a simple single hidden layer neural network with a single output neuron. Sigmoid activation is applied to the output neuron to predict the probability of whether the corresponding attribute (valence or arousal) is high (1) or low (0). The regularization loss is formulated as

$$L_{reg_{disc}} = BCE(v_{pred}, v_{gt}) + BCE(a_{pred}, a_{gt}) \tag{5}$$

where $BCE(.)$ refers to Binary Cross Entropy Loss, v_{pred} and a_{pred} are the sigmoid probabilities for valence and arousal values. v_{gt} and a_{gt} are ground truth values for valence and arousal based on the quadrant annotations. For instance, for a music piece with annotation as quadrant 4, the valence is high and the arousal is low. This implies $v_{gt} = 1$ and $a_{gt} = 0$. An illustration of Discrete Regularization can be seen in Fig. 3(a).

Once either Discrete or Continuous regularization has been performed during training, the valence and arousal attributes can be manipulated by simply changing the values of the dimensions z^1 and z^2 for any piece of music. This is done either as per VA_i extracted from the script or a slider, by performing a simple addition $z^1 = z^1 + V$ and $z^2 = z^2 + A$ before decoding the music.

A.3 Latent Space Interpolation

Since attributes of the data are encoded within the latent space by the VAE, interpolation in the latent space of the VAE would allow us to smoothly change the attributes in a piece of music. This would be particularly useful when the sentiment changes over the course of a scene. For instance, the start of the scene could have low arousal while the end of the scene has high arousal. The changes in sentiment can be captured by extracting VA values at a sentence level and manipulating z for the start and end of a scene using VA_{start} and VA_{end} which refer to the VA values for the first and last sentence of a scene respectively. The pieces of music for the other parts of the scene can be filled using interpolation between z_{start} and z_{end}, as can be seen in Fig. 5.

B Scene-Wise Analysis of User Survey

We analyse the scene-wise consensus among users by averaging the overall mood fit ratings for each scene across all users in Table 2. Scene 1 — a dramatic, high-impact and large-scale sequence
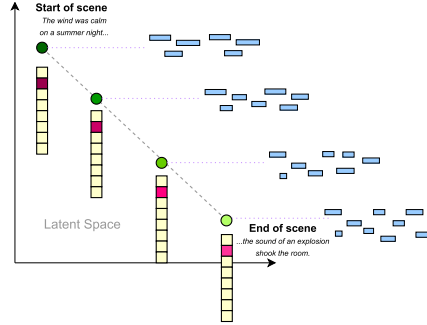
Figure 5: Latent Space Interpolation

— requires nuanced and sophisticated music to enhance the experience, which EMOPIA-CWT is able to do better with its polyphony. In Scene 2 — which is a negative, action-packed fight scene — MusicVAE does better, as EMOPIA-CWT is known to struggle with music with negative sentiment. Additionally, the action in the scene is paralleled better in MusicVAE due to attribute vector arithmetic, where the addition of the high arousal vector (z_{ah}) induces fast-paced notes in music. EMOPIA-CWT again does better in Scene 3 as it is an emotional scene with almost no action sequence.

C   Additional Experiments

C.1   Finetuning VAE with Discrete Regularization

FIGARO [38] is a recent work that uses VAEs to extract high-level "descriptions" at a bar-level, and then uses the sequence of descriptions to generate coherent music with a Seq2Seq Transformer model. We finetune FIGARO's [38] pretrained VAE model, on the EMOPIA dataset [16]. This is done by adding a discrete regularization component to the VAE training loss as per Eqn 5, to improve sentiment conditioning (described in A.2.2). This helps FIGARO's VAE model learn a better latent space representation for sentiment, in addition to the reconstruction properties carried over from the pretraining.

In Fig. 3(b), the discrete regularization loss continues to decrease during finetuning which indicates that the model is able to predict the Valence and Arousal values of the piece of music better after finetuning. However, the finetuning process is very slow and takes many epochs to converge as the gradient from the discrete regularization loss component is much smaller relative to the reconstruction and KL divergence loss components of the VAE.

C.2   Polyphonic Recurrent VAE

Experiments were conducted to train a simple Recurrent VAE baseline for polyphonic music with a similar architecture to [3]. We use a bidirectional LSTM as the encoder and a vanilla LSTM as the decoder, with one layer each. During training, the input music is fed into the bi-LSTM encoder. The hidden vector of the last state of the bi-LSTM is mapped to the latent dimensions through a fully connected layer, which is then fed into the LSTM decoder to output music. We additionally concatenate the latent vector z with the input embedding at every time step of the decoder.

The polyphonic recurrent VAE however suffers from posterior collapse, where the LSTM decoder simply ignores the latent vector z and learns to output a repeating sequence of notes. This is especially the case when teacher forcing rate [2] is set to 0 in the decoder. When the teacher forcing rate is set to 1, the problem of repeating notes is solved, but the reconstruction for all pieces of music remains the same due to posterior collapse. We detail the results of all our experiments, as well as the hyperparameters chosen in the Supplementary Material.

---

[2]percentage of times ground truth is fed back as input to decoder instead of previous decoder output

D Related Work

D.1 Sentiment Analysis

One-dimensional sentiment polarity analysis, commonly referred to as sentiment analysis, refers to identifying whether textual data conveys a positive or negative sentiment. Many benchmark datasets for sentence-level and phrase-level binary classification [35, 20, 42, 26] have been developed. Pre-trained Transformer-based models [31, 6, 41, 19] have been fine-tuned on the benchmark datasets to obtain state-of-the-art binary sentiment classification.

To capture more complex sentiments, sentiment analysis techniques based on discrete sentiment categories have been explored. Few public datasets exist for multi-emotion classification. EmoLex [22] maps English language words to 8 discrete emotion categories. ISEAR [33] and GoEmotions [5] are multi-emotion sentence classification datasets with 28 and 7 emotion classes respectively.

The NRC VAD [21] lexicon maps 20,000 English words to a continuous space of real-valued valence, dominance and arousal scores. For sentence-level continuous emotion regression, the EmoBank [4] dataset contains 10,000 sentences labelled with valence, arousal and dominance values. The drawback of this dataset is that there is an imbalance wherein most sentences are labelled with VA values in the range of 2 to 3.

D.2 Music Generation

Music generation has been approached in two major ways. The first is to generate music as raw audio (a .wav, for instance) [7, 1]. The second is to generate music in the symbolic space either as a probability over all possible notes for each time step [32, 13] or as a sequence of tokens in one of many tokenized representations of music [27, 15, 43, 12], which can then be converted to a MIDI (.mid) file. We discuss approaches for symbolic music generation due to the complexities associated with raw audio domain music generation.

Variational autoencoders for generating music were popularised by MusicVAE [32], a hierarchical variational autoencoder with recurrent neural networks (RNNs) as encoder and decoder. To decompress the latent representation while avoiding posterior collapse, a hierarchical RNN decoder was used. However, the output produced by the MusicVAE architecture is limited to *monophonic music* and is unable to produce *polyphonic music* [3]. The authors also proposed using attribute vector arithmetic in the latent space but limited it to simpler attributes like note density. More recent VAE approaches for music like [39, 18, 38] have handled polyphonic music in a VAE. In addition to VAEs, transformers [37] have also been used widely to generate music [13, 12].

Various approaches have recently been proposed specifically for sentiment-conditioned music generation. Lucas et al. [10] propose a combination of mLSTM and a logistic regression model. The MIDI files are represented as a series of events, relevant to features such as timbre, harmony, tempo, etc. The sentiment is interpreted from these features to create the partially annotated VG-MIDI dataset (annotated with a valence-arousal pair). The generative mLSTM is trained on the unlabeled dataset and the logistic regressor uses the hidden state to encode MIDI phrases and predict statements. However, it fails to accurately generate music for negative sentiments and cannot handle inputs more detailed than a given set of labels/emotions.

Hung et al. [16] introduced the EMOPIA dataset, a collection of labelled pop-piano midi files based on the valence-arousal model of sentiment. The authors provided baselines for music classification and sentiment-conditioned music generation. The compound word transformer (CWT) [12] was the baseline proposed as a sentiment-conditioned music generator. This however is limited to generating music conditioned only by a given quadrant of the VA space.

Other recent approaches to this problem include Transformer-GANs [25] and Music FaderNets [36]. The first proposes using an additional adversarial loss component to the usual negative log-likelihood (NLL) loss utilized to train transformers for music generation, as a way to prevent the degradation of quality in longer sequences of music. However, the conditioning control is again only limited to the quadrant number and does not allow for sentiment modifications to a given piece of music. Music FaderNets aimed at bypassing the question of formalizing high-level notions like "sentiment"

---

[3]In monophonic music, only one note can be played at any time. In polyphonic music, multiple notes can be played together simultaneously (like chords)

by trying to force the model to infer them from a dataset of low-level features using semi-supervised clustering. Currently, however, FaderNets are trained only to be conditioned on arousal.

E   Data and Dataset Preprocessing

For movie scripts, we use the IMSDb [17] and MovieNet [14] databases to source movie scripts. For sentiment analysis of text, we use the NRC VAD [21] lexicon, a dataset of valence, dominance and arousal human ratings. For music generation tasks, we use the AILabs Pop1k7 dataset [12] and the EMOPIA dataset [16] which contains MIDI piano pieces categorized into one of the 4 quadrants in the valence-arousal space.

We used the MidiTok [24] library to tokenize the AILabs1k7 and EMOPIA datasets into REMI [15] representations, and chunked them into single-bar representations.