Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

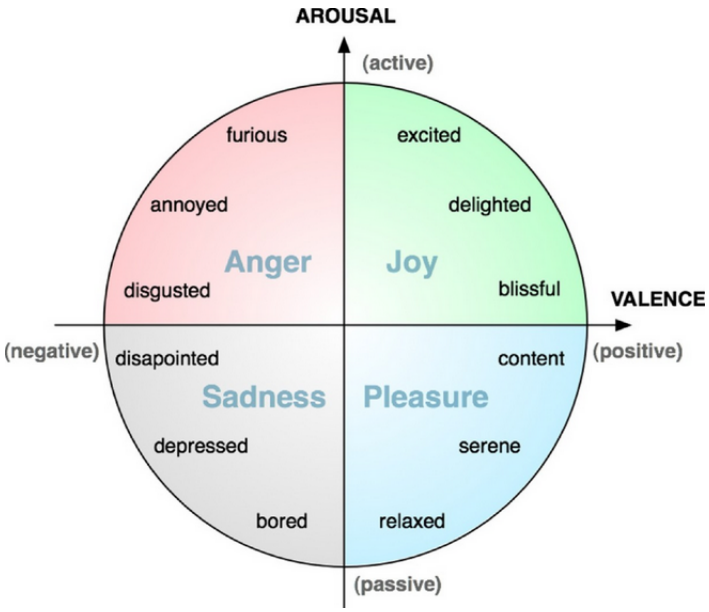
IIC3633 — Sistemas Recomendadores — 2' 2020

Informe Intermedio Proyecto

1 Exploración de Datos

Nuestro proyecto busca generar música a partir del contenido de una imagen, por lo que necesitamos datos de tanto imágenes como de audio. Además, para realizar el matching entre imágenes y audio nos guiaremos por la metodología propuesta en [1] en donde se comparan los valores de VA (Valence - Arousal) de ambos medios, por lo que también necesitaremos anotaciones de VA de las imágenes y audios que tengamos.

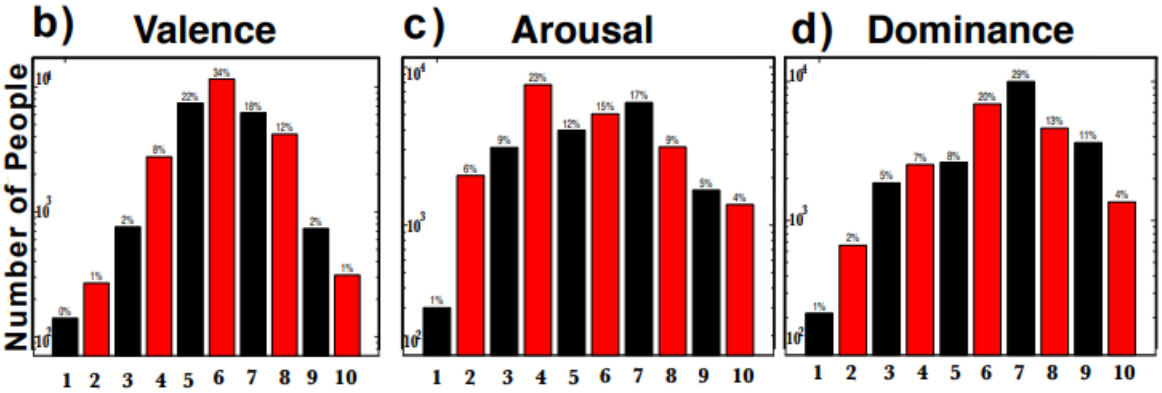
El mapping Valence - Arousal es un método de clasificación de emociones de alto nivel de granularidad. Esto quiere decir que permite diferenciar de manera más específica las emociones ubicándolas en un espacio bidimensional donde los ejes corresponden a los niveles de valencia (indican que tan placentera o positiva es una emoción) y excitación/estimulación. El gráfico a continuación muestra la ubicación de algunas emociones en este espacio:



Por el lado de las imágenes, tenemos actualmente acceso a dos datasets: OASIS [2] y EMOTIC [3].

- OASIS: Este dataset cuenta con 900 imágenes de tópicos muy variados, como personas, animales y objetos (entre otros). Cada una de las imágenes del dataset tiene además un valor de valencia y otro de arousal, los cuales corresponden al promedio de todas las anotaciones que tuvo dicha imagen.

- EMOTIC: Este dataset consiste en 23189 imágenes de personas, las cuales se encuentran en distintos contextos (es decir, pueden estar sentadas, riéndose, durmiendo, etc) y cada una de estas imágenes posee anotaciones de Valence, Arousal. Además de estas 2, también poseen anotaciones para Dominance (otra métrica menos usada) y 26 categorías discretas de emociones. Abajo se puede apreciar un gráfico que describe la distribución de valencia, arousal y dominance en el dataset. Fue obtenido del paper original [3], en donde hay disponible más análisis de este dataset.



Como se puede ver, tenemos una gran cantidad de datos para la parte de las imágenes, sin embargo hay que tener en cuenta que la gran mayoría provienen de EMOTIC, el cual sólo posee imágenes de personas. Esto puede hacer que a nuestra red le cueste identificar emociones en imágenes que no posean personas. Sin embargo, si bien todas las imágenes tienen personas, en la mayoría de las imágenes se puede ver mucho más que solo la persona (incluso en muchas imágenes la persona puede estar de espalda, o haber varias personas, etc). En general son muy variadas.

En cuanto a los datos de audio, se buscaron datos para poder realizar el mapping de audio a emoción y datasets de secuencias MIDI. En cuanto al mapping de emoción, se buscaron datasets análogos a los de imágenes donde se asocian valores de VA a clips de audio. Por otro lado, las secuencias MIDI son la fuente de datos que permitirá alimentar los modelos de generación de música. Se experimentó aplicar el mapping a emoción sobre estas secuencias MIDI para comprobar la posibilidad de generar un dataset que asocie las secuencias a valores de VA. Los datasets usados son:

- DEAM Dataset: Este dataset consiste de 2058 extractos y canciones completas anotadas con sus respectivos valores VA por segundo. También cada clip de audio tiene el promedio y desviación estándar de sus valores de VA. Se entrega el metadata de los audios (Nombre de artista, canción etc) y features extraídos del audio con el programa openSMILE. Los features se guardan en archivos .csv que cada 500ms se asocian un set de features a un timestamp. Los audios están en formato .mp3 y el sampling rate es de 44,1 KHz. Más detalles sobre el dataset se encuentran en el manual [4].
- Musical AI MIDI Dataset: Es una colección de 77153 canciones en formato MIDI obtenida de múltiples datasets. Los datasets son: Big Data Set, Cariart, Download-midi, Guitar midkar, ICS, The Lahk MIDI Dataset y TV Themes de www.tv-timewarp.co.uk.

2 Estado de Avance

Para esta entrega, nos centramos en tanto la exploración de datos y de bibliografía, como también aterrizar nuestra propuesta inicial a un problema concreto con el fin de identificar las distintas tareas que se deben ll-

evar a cabo para realizar el proyecto. La exploración de los datos se puede ver en la sección 1 de este informe.

A partir de nuestra propuesta inicial, decidimos centrarnos en generar música basándonos en una imagen, en donde se espera que la música generada provoque un sentimiento similar al que provoca la imagen que se le entregó al modelo. Para esto, decidimos que tanto las imágenes como el audio serían clasificados en base a los valores de Valence y Arousal propuestos en [5]. De esta manera, tanto imágenes como audio se pueden representar en un espacio de 2 dimensiones de VA, y cuantificar la distancia entre ellos utilizando distancia euclidiana, siguiendo el método propuesto en [1]. Finalmente, nuestro objetivo es minimizar la distancia de la imagen y el audio generado en el espacio VA.

Para lograr realizar lo anterior, hay 3 tareas fundamentales que debemos realizar: Obtener el VA de una imagen arbitraria, generar música tomando como input el VA de la imagen y finalmente obtener el VA de la música generada para compararlo con el de la imagen. A continuación se describirá el estado de avance (tanto de investigación como de implementación) en cada una de estas tareas.

2.1 Mapping de imagen a emoción (Valence - arousal space)

Predecir los valores de VA de una imagen es una tarea muy importante en nuestro proyecto, debido a que se utilizará este valor predicho como input para generar la música. Es por esto que en lo posible esta predicción tiene que ser lo más precisa posible, con el fin de que realmente la música generada se sienta parecido a la imagen en cuestión.

Dado que necesitamos una alta precisión al momento de predecir, buscamos entre los papers más recientes del área con el fin de comprender un poco el estado del arte en esta tarea y poder elegir el mejor método posible. De esta manera, encontramos 3 papers [1] [6] [7] bastante actuales que realizan la tarea de predecir el VA de una imagen. De estos 3, solamente 2 tienen implementaciones open-source y lamentablemente ninguno de estos posee algún modelo preentrenado, por lo cual para testearlos habrá que entrenar dichos modelos.

Finalmente se vio que tan factible es utilizar estos modelos para nuestro proyecto. Si bien existen algunas dificultades como los datos a utilizar o la necesidad de realizar una implementación y/o entrenamiento de la red (las cuales son detalladas en la sección 3 del informe), por lo menos los resultados que reportan indican que nos servirían para nuestro proyecto. Los pasos a seguir a partir de aquí están detallados en la sección 4 del informe.

2.2 Generación de música con VA específico

Esta tarea consiste en generar una pieza musical nueva mediante técnicas de deep learning. Esta pieza debe provocar una emoción lo mas similar posible a la emoción que provoque una imagen seleccionada, lo cual se traduce en minimizar la distancia de la imagen y el audio en el espacio VA. Para lograr esto, nuestra idea es entregarle como input a la red generadora el valor VA de la imagen, de manera que la red sepa cual el es valor objetivo de VA que debe tratar de generar.

Generar música utilizando deep learning no es una tarea fácil, pero por suerte existe mucha investigación en el área sobre una gran variedad de técnicas como el uso de GANs, LSTM, VAE, entre otros. Por ejemplo, Music VAE [11] de Magenta es un Variational Autoencoder (VAE) en donde el encoder consiste en una red LSTM, y el decoder es una red LSTM jerárquica. Esta red permite generar secuencias de notas a partir de un input, o bien interpolar entre 2 secuencias. Otro modelo capaz de generar música es el Wasserstein Autoencoder (WAEs). Dado una colección de datos, el objetivo de las WAEs es ajustar un modelo capaz de generar datos sintéticos que sean similares al conjunto de datos recibido. Existen dos formas de entrenar las WAEs, una de ellas es WAE-GAN que usa adversarial training introduciendo un discriminador que busca clasificar los datos reales de los sintéticos. La otra forma es WAE-MMD que consiste en la utilización de la

métrica Maximum Mean Discrepancy para la elaboración de la función de pérdida. Este segundo método asume que los datos pertenecen a una distribución y que las muestras son i.i.d.

Nuestra idea es basarnos en alguna de las arquitecturas mencionadas anteriormente para crear nuestro modelo generador, pero modificando el input de estas redes para que acepten una tupla VA. De esta manera, el generador tendría la información del VA objetivo al que se quiere llegar. Luego, en la función de pérdida integraríamos un término adicional que midiese la distancia euclidiana entre la tupla VA objetivo y la tupla VA de la canción generada por la red. De esta manera, al minimizar esta pérdida se estaría logrando reducir la distancia emocional entre la canción generada y el VA objetivo.

Sin embargo, para lograr esto tenemos en mente dos posibles enfoques que dependen de que tan difícil sea entrenar la red para que logre generar música con sentido (sin tomar en cuenta el tema emocional):

- El primer enfoque es simplemente entrenar la red desde cero, creando nosotros una pérdida personalizada que tome en cuenta tanto la pérdida original de la red (para que pueda generar música con sentido) como la pérdida por distancia de VA, modificando además las layers de input para agregarle la información de VA objetivo. Este enfoque tiene la ventaja de que es un método bastante directo para lograr lo que queremos y tendríamos control completo sobre los datos que haya visto nuestro modelo para ser entrenado, sin embargo tiene la desventaja de que en algunos casos entrenar estos modelos puede ser muy costoso en cuanto a tiempo y hardware.

- La otra opción que tenemos es trabajar sobre la red con pesos pre-entrenados, de manera que la red ya sepa como generar música, y luego *finetunearla* para condicionar a la red a que genere música con el VA que se requiere. Este *finetuning* sería muy similar al entrenamiento que se haría en el otro enfoque, en donde modificariamos la pérdida para que comience a tomar en cuenta el VA. Trabajos anteriores como [12] ya han explorado la posibilidad de *finetunear* redes generativas de música (aunque en ese caso utilizando Reinforced Learning, nosotros utilizaríamos un enfoque más tradicional para *finetunear*), en donde en ese caso buscaron enseñarla a una red ya entrenada para generar música acerca de teoría musical. En el nuestro, sería una idea parecida pero enseñándole a generar tomando en cuenta la emoción.

Aún no hemos decidido cuál de los dos enfoques vamos a utilizar dado que nos falta primero elegir que red vamos a utilizar de las dos que mencionamos y luego comprobar que tan difícil es entrenarlas. Más detalle acerca de los pasos a seguir están en el plan de avance.

2.3 Mapping de música a emoción (Valence - arousal space)

La búsqueda de bibliografía permitió encontrar este artículo [8] que presenta un modelo de predicción de emociones a partir del mismo dataset DEAM que se exploró, utilizando Triplet Neural Networks. Los autores publicaron el código fuente de su trabajo que posee funciones de preprocesamiento de los datos, el modelo con los pesos ya entrenados y visualizaciones. A continuación se detalla el proceso de experimentación que se realizó con el repositorio del paper:

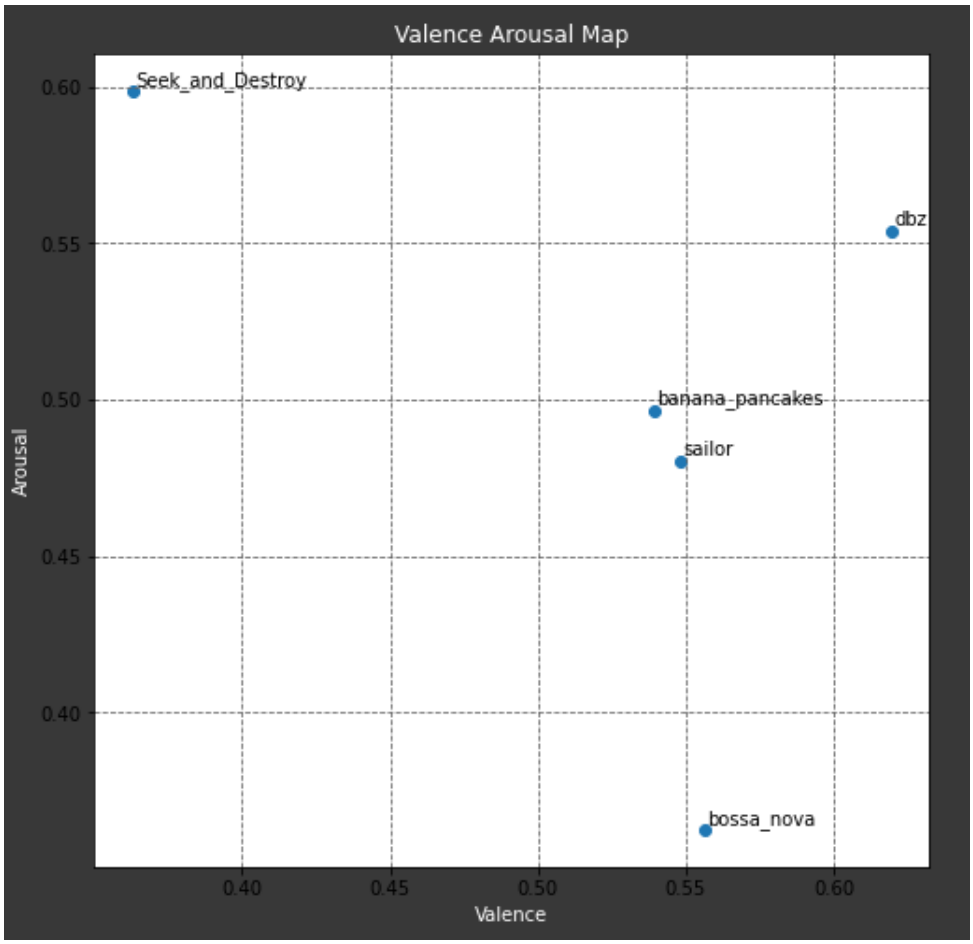
- Lo primero que se realizó fue seleccionar algunos archivos MIDI del dataset Musical AI para posteriormente abrirlos en un Digital Audio Workstation con el fin de exportarlos como archivos de audios en formato .wav ya que se requiere este formato para el siguiente paso del preprocesamiento. De todas formas, el código fuente posee un método que transforma .mp3 a .wav que sería útil si se requiere realizar este proceso con audios .mp3 de datasets como FMA.

- Una vez obtenido el archivo de audio .wav, se tiene que extraer sus features con un software llamado openSMILE. Precisamente se utiliza la herramienta SMILExtract que genera los archivos .csv con las features obtenidas del audio cada 500ms. Para poder utilizar este programa, se tuvo que descargar su código fuente escrito en C++, compilarlo y utilizar el ejecutable SMILExtract con un archivo de configuración encontrado en el repositorio del código fuente del paper antes mencionado.

- El archivo .csv se transforma a un numpy array y se le aplica normalización z-score. En este punto se carga un modelo y sus pesos preentrenados para hacer una regresión con el fin de obtener el valor de valence y otro para obtener el valor de arousal.

El modelo corresponde a una TNN (Triplet Neural Network) que corresponde a una fully connected layer con función de activación ReLu que opera como un reductor de dimensionalidad. La gracia de esta TNN por sobre otras técnicas de reducción de dimensionalidad, es que logra capturar relaciones no lineales entre las features del audio. Al output de la TNN se le aplica Gradient Boosting para realizar la regresión de los valores de VA. Se puede denominar este modelo como GBM-TNN dada sus componentes Gradient Boosting Model y Triplet Neural Network.

Se realizó el procedimiento previamente explicado con los archivos MIDI de las canciones Seek and Destroy - Metallica, Banana Pancakes - Jack Johnson, Dragon Ball Z Theme, Sailor Moon Theme, y un bossa nova (los audios utilizados están disponibles aquí). Se presenta a continuación los resultados obtenidos.



Valores de VA por canción		
Canción	Valence	Arousal
Seek and Destroy	0.363012	0.598811
Banana Pancakes	0.539223	0.496738
bossa nova	0.556183	0.362598
DragonBallZ Theme	0.619157	0.554055
Sailor Moon Theme	0.547877	0.480442

Se puede apreciar que el modelo predictor genera resultados que tienen sentido. Se ubica al bossa nova en la zona de arousal baja y valencia alta en donde se encuentran las emociones de relajo y serenidad. Seek and Destroy tiene una baja valencia y muy alto arousal. En ese sector se ubican las emociones de furia y enojo. Dragon Ball Z Theme tiene alta valencia y alto arousal por lo tanto se acerca a las emociones de alegría. En general las emociones que se asocian a los valores de valence y arousal obtenidos si guardan relación con las canciones.

3 Problemas Encontrados

En esta sección se listarán los problemas encontrados:

- Datasets: Si bien finalmente logramos obtener buenos datasets con una buena cantidad de items, hubieron varios datasets que nos podrían haber servido como IAPS y DAPS, pero que lamentablemente para acceder a ellos había que enviar una solicitud, la cual por lo bajo se demora 30 días en ser respondida, e incluso podría haber sido rechazada. Dado a que no podemos esperar tanto tiempo y tampoco podemos quedarnos sin datos en el caso que nos rechacen, tuvimos que rendirnos con estos datasets. Además, hubieron muchos datasets que utilizaron ciertos papers que vimos que lamentablemente no los liberaron al público y nos hubieran servido bastante.

  En cuanto a datasets de audio, no se encontraron datos de archivos MIDI con sus valores de VA respectivos. Lo que si se obtuvo, fue datasets de secuencias MIDI y un modelo predictor que otorga valores de VA a un archivo de audio, por lo que es posible transformar MIDI a audio para obtener sus valores de VA.

- Implementaciones de papers: Un problema que tenemos en la parte de mapeo de imágenes a VA es que las implementaciones (si es que existen) no incluyen los pesos del modelos entrenado, lo que significa que tendremos que gastar tiempo entrenando nosotros dichos modelos. Además, los datasets utilizados en aquellos trabajos son diferentes a los que tenemos nosotros (ver la dificultad de arriba), por lo que existe la posibilidad no menor de que dichos modelos funcionen mal en nuestros datos. En este caso, probablemente tendríamos que diseñar nosotros un modelo capaz de predecir el VA de una imagen, lo cuál queremos evitar dado a que nos tomaría un tiempo no despreciable.

- Falta de automatización en la predicción MIDI a emoción: Ahora para poder transformar las secuencias MIDI a sonido se esta realizando mediante el uso de un DAW. Se utiliza el DAW de tal modo de asegurar que los instrumentos que ejecuten las secuencias sean los acordes para la canción seleccionada. Faltaría revisar algún programa que permite escribir un script para hacer las conversiones de manera automática sobre un conjunto de archivos MIDI y evaluar que los instrumentos que asigne a las secuencias no hagan perder características de la canción original (al seleccionar instrumentos muy distinto al instrumento original). También se probó la ejecución del programa SMILExtract en Python para automatizar la generación de los archivos con features, pero no se tuvo éxito. Este último problema se puede dar probablemente a una imposibilidad de ejecutar un archivo ejecutable de Google Drive en un Google Colab notebook. Si bien no es un problema que se vea difícil de solucionar, si hace que el proceso

sea más lento. En general el proceso tiene muchas etapas distintas que requieren de generar archivos intermedios que ocupan bastante memoria. Se debe buscar la manera de automatizar el proceso e ir eliminando los archivos intermedios durante la ejecución.

4 Plan de Avance

Para la tarea de mapeo de imágenes a VA, nos quedaría entrenar y testear las implementaciones disponibles de los métodos mencionados anteriormente utilizando nuestro cuerpo de datos. A partir de esto, si los resultados son satisfactorios esta tarea quedaría relativamente terminada, dado que ya tendríamos lista la parte de predecir el VA de la imagen y no habría necesidad de crear un modelo nuevo o seguir entrenando. Si por el contrario los modelos no funcionan bien, tendremos que implementar nuestra propia versión adaptada a nuestro dataset. El problema en si no es muy difícil, pero igual nos tomará un tiempo considerable realizar esto y que funcione bien.

Para la tarea de generación de música, consideramos necesario generar un dataset propio de archivos MIDI y su valor respectivo de valence-arousal dado que no se han encontrado datasets open-source con esta información. La tarea de generar este dataset es factible dado que se posee un dataset MIDI grande y variado, que en conjunto con el predictor de emoción, se puede recopilar los archivos MIDI con su mapping en el espacio VA correspondiente. Además, el mapping que se realiza demuestra que tiene sentido y que logra capturar las cualidades emocionales de las canciones. Este dataset nos servirá para generar un regresor capaz de asignar valores de VA directamente a los archivos MIDI sin tener que convertirlos a audio. Nos podemos basar en el mismo modelo utilizado [8] , cambiando el extractor de features SMILExtract por otro extractor de features de MIDI como jSymbolic [13]. El baseline de rendimiento de nuestro regresor en primera instancia será que logre mapear la emoción del MIDI en el cuadrante correcto (High Valence-Low Arousal, High Valence High Arousal, Low Valence-Low Arousal, Low Valence- High Arousal). Este mínimo de calidad ya es valioso dado que las emociones que se encuentran en un mismo cuadrante son muy similares entre ellas.

Una vez generado este dataset y el regresor, se tendrá que construir un modelo basado en las arquitecturas mencionadas en la sección 2.2 que tenga la particularidad de recibir valores de VA para generar música que evoque emociones asociadas a tales valores. Estos valores de VA de input para el modelo serán obtenidas del output de la predicción de emoción de una imagen seleccionada. Primero pensamos probar entrenar desde cero estos modelos para ver el tiempo que tardan (sin hacer ningún cambio en este punto), o al menos hacernos una idea de cuanto tardarían en entrenarse por completo. Luego, si es factible entrenarlos por nuestra cuenta decidiremos utilizar el primer enfoque detallado en la sección 2.2, en caso contrario utilizaríamos el segundo enfoque en donde *finetunearemos* una de las redes. Además, en este punto decidiremos la red a utilizar para generar música. Una vez definido esto, definiremos formalmente la pérdida personalizada para entrenar la red y modificaremos la entrada de la red para agregarle el valor de VA objetivo. Finalmente, quedaría entrenar la red con nuestro cuerpo de datos y evaluar los resultados obtenidos para asegurarnos que se logre transmitir la emoción deseada a través de la música.

References

[1] Zhao, S., Li, Y., Yao, X., Nie, W., Xu, P., Yang, J., Keutzer, K. (2020, October) *Emotion-Based End-to-End Matching Between Image and Music in Valence-Arousal Space.* In Proceedings of the 28th ACM International Conference on Multimedia (pp. 2945-2954)

[2] Kurdi, B., Lozano, S., Banaji, M. R. (2017). *Introducing the open affective standardized image set (OASIS).* Behavior research methods, 49(2), 457-470.

[3] Kosti, R., Alvarez, J. M., Recasens, A., Lapedriza, A. (2017). *Emotion recognition in context.* In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1667-1675)

[4] Soleymani, M., Aljanakil, A., Yang, Y. (2018, April) *DEAM: MediaEval Database for Emotional Analysis in Music* PLOS ONE.

[5] A. Mehrabian. (1995) *Framework for a comprehensive description and measurement of emotional states.* Genetic, social, and general psychology monographs

[6] Zhao, S., Jia, Z., Chen, H., Li, L., Ding, G., Keutzer, K. (2019, October). *Pdanet: Polarity-consistent deep attention network for fine-grained visual emotion regression.* In Proceedings of the 27th ACM International Conference on Multimedia (pp. 192-201).

[7] Kim, H. R., Kim, Y. S., Kim, S. J., Lee, I. K. (2018). *Building emotional machines: Recognizing image emotions through deep neural networks.* IEEE Transactions on Multimedia, 20(11), 2980-2992.

[8] Cheuk, K., Luo, Y., Balamurali, B., Roig, G., Herremans, D. (2020, July). *Regression-based Music Emotion Prediction using Triplet Neural Networks* ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing.

[9] Giovanni Bindi (2020) *Music generation with Wasserstein Autoencoders* https://w00zie.github.io/post/wae/wae-gan

[10] Yang, L. C., Chou, S. Y., Yang, Y. H. (2017).*MidiNet: A convolutional generative adversarial network for symbolic-domain music generation.* arXiv preprint arXiv:1703.10847.

[11] Roberts, A., Engel, J., Raffel, C., Hawthorne, C., Eck, D. (2018). *A hierarchical latent vector model for learning long-term structure in music.* arXiv preprint arXiv:1803.05428.

[12] Jaques, N., Gu, S., Turner, R. E., Eck, D. (2016). *Generating music by fine-tuning recurrent neural networks with reinforcement learning.*

[13] McKay, C., Fujinaga, I. (2006). *jSymbolic: A Feature Extractor for MIDI Files.* In ICMC.