

Overall Notes

Where to find sources for computer science topic

- Google Scholar
- JSTOR
- Arxiv

“To what extent can neural networks create numerical representations of musical pieces?”

Introduction:

The relevance of the topic

- Can start off by re-examining the nature of humanity. Furthermore, I can delve into the essence of emotions and how music impacts individuals.
- This could go further into the pursuit of Artificial General Intelligence and how it strives to mimic the human brain
- If we can create a numerical representation of music, we can compare, contrast different types of music to better understand this culture overall

Literature Review:

Music and audio files in general are a type of sequential input, similar to text. This has implications with relevant approaches in the domain of natural language processing (NLP).

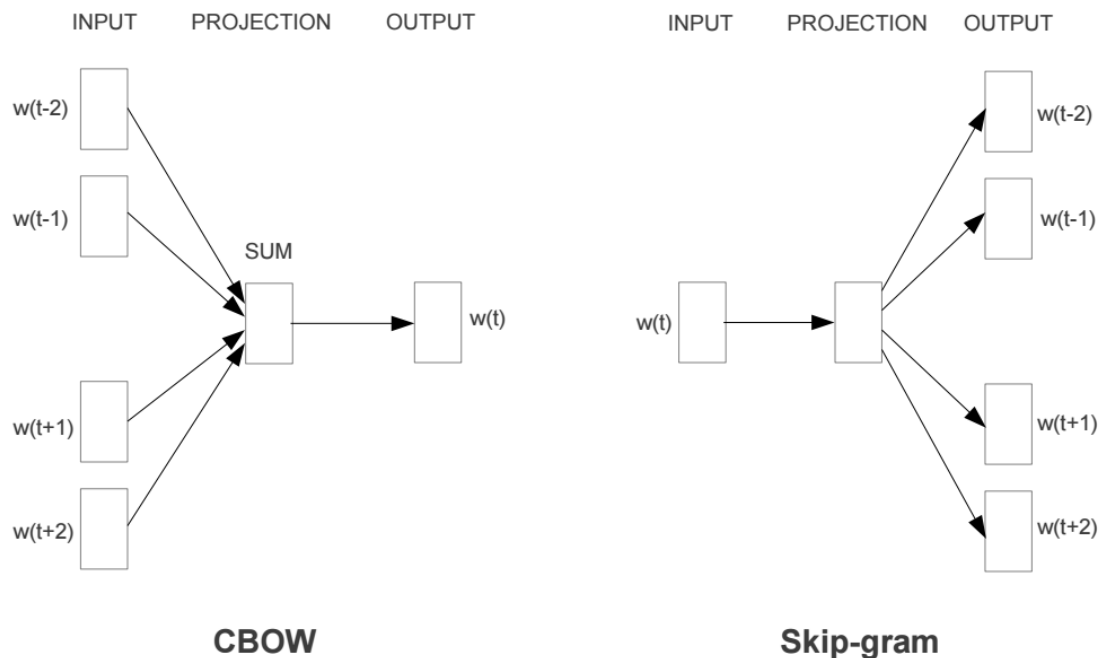
The mapping of abstract concepts like words into a numerical representation -- more concisely, *latent space* -- has already been done.

Annotated Bibliography and Ideas

How are audio files represented in latent space?

word2vec: <https://arxiv.org/pdf/1301.3781.pdf>

This paper introduces word2vec, a neural network algorithm that represents words as continuous vectors of multiple dimensions. There are two flavours of the word2vec model that generates word vectors (more precisely, embeddings). The Continuous Bag of Words CBOW and the Skip Gram model:



The CBOW model predicts the next word given the current word, and creates a projection, while the Skip-gram model uses a “fill in the blank” type of logic, where given a word, the model predicts the context words surrounding it.

I think that we can apply a similar logic to music. This leads to a variation of the doc2vec model that I should read later: <https://arxiv.org/abs/1405.4053>

Literature Review of deep learning in signal processing: <https://arxiv.org/pdf/1905.00078.pdf>

In their analysis section they reviewed tasks like genre classification. One author they mentioned approached the task using 1x1 convolutions in CNNs that were then averaged over the entire

music file to obtain a genre label. Another author used CNNs but with a 3x3 convolution which is eventually reduced to a 1x1 feature then classified.

“However, on the research side, neither within nor across tasks is there a consensus on what input representation to use (log-mel spectrogram, constant-Q, raw audio) and what architecture to employ (CNNs or RNNs or both, 2D or 1D convolutions, small square or large rectangular filters), leaving numerous open questions for further research.”

<https://arxiv.org/pdf/1811.12408.pdf>

Authors discuss the use of the aforementioned word2vec model. They highlight that CNNs are used to typically encode musical files, then processed to a RNN/LSTM model. However, not many have approached the “encoding” aspect using the word2vec model.

Relevant datasets used were: Musedata from CCARH, JSB chorales, classical piano archives, and Nottingham folk tune collection.

The way they processed their data was encoding an entire musical piece into “slices”. Each slice is one beat long and would be labeled an appropriate pitch like below:



Their results showed that the word2vec embeddings can semantically understand chords, keys and analogies (which are similar to chord progressions).

Makes me think: should I narrow my research topic further? I need to think how I am going to encode my music and measure what exactly? Should I quantify human emotion or chords or something?

Modeling Musical Context Using word2vec: <https://arxiv.org/pdf/1706.09088.pdf>

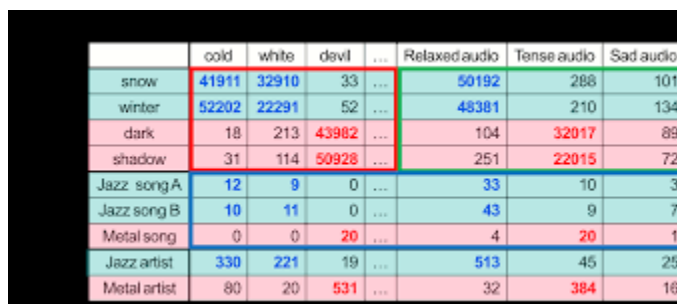
Similar to the previous research paper, they considered musical slices as “chords”. They also truncated their 128 dimension vectors to 2 dimensions using t-SNE, a dimensionality reduction

algorithm. I should probably consider this approach in my extended essay or using the PCA algorithm.

Literature Review Paper to Read: <https://arxiv.org/pdf/2201.02490.pdf>

Query By Blending: <https://archives.ismir.net/ismir2019/paper/000015.pdf>

Instead of encoding one type of feature, they used a co-occurrence matrix that feeded to the neural network. They also encode the artist name, word used and audio track and apply vector addition to find new queries. I could include this in my paper. It seems very interesting. More reading is needed to fully understand this paper.

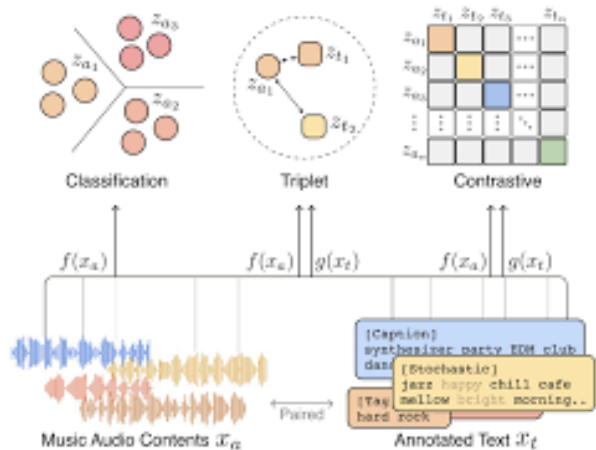


	cold	white	devil	...	Relaxed audio	Tense audio	Sad audio
snow	41911	32910	33	...	50192	288	101
winter	52202	22291	52	...	48381	210	134
dark	18	213	43982	...	104	32017	89
shadow	31	114	50928	...	251	22015	72
Jazz song A	12	9	0	...	33	10	3
Jazz song B	10	11	0	...	43	9	7
Metal song	0	0	20	...	4	20	1
Jazz artist	330	221	19	...	513	45	25
Metal artist	80	20	531	...	32	384	16

This is the co-occurrence matrix they mentioned. Following the Distributional Hypothesis, similar words would have similar context words as well. Each row in the matrix is called the target vector that is a vector representation of the target word. A similarity score is defined as the distance between the rows of this matrix. This logic is applied to create artist and audio embeddings.

<https://arxiv.org/pdf/2211.14558.pdf>

It is a text to retrieval system that encodes audio and text. Their architecture consists of 3 different sections. I only understand the 2 of them:



Classification Model:

It attempts to map an audio embedding with its mood/tone pair. It then tries to minimize the similarity distance between the most similar audio tracks.

Triplet Loss Model:

Attempts to map an audio embedding with its associated text/tag/descriptor pair. It tries to maximize the similarity between associated text-audio pairs and minimize similarity with non-related audio tracks and descriptors.

They also used different datasets that could be relevant for my project such as the “million song dataset” and the MTG-top50s dataset that contains 55k tracks as well as annotated tone, mood and theme labels for each 30s song.

Some ideas after reading the paper and referring to the previous one:

I am thinking of making a model that’s like the following:

1. You can add, subtract emotional word vectors like “energetic”, “sad”, “happy”
2. You can subtract the audio vectors of one song to get another song.
3. How do I measure the accuracy of the model? How do I make sure it has understood the semantic meaning of emotion and song?
4. How do I combine word and audio vectors/input?

Potential Datasets

<https://paperswithcode.com/dataset/musiccaps>

<https://www.kaggle.com/datasets/googleai/musiccaps>

Contains

- 1) A free-text caption consisting of four sentences on average, describing the music and
- 2) A list of music aspects, describing genre, mood, tempo, singer voices, instrumentation, dissonances, rhythm, etc.

<http://millionsongdataset.com/>

MTG-top50s dataset: <https://github.com/MTG/mtg-jamendo-dataset>

Other questions to consider:

- Other than the word2vec model, should I read more about other models or go deep dive into this only methodology?
- What should I measure exactly?
- Makes me think: should I narrow my research topic further? I need to think how I am going to encode my music and measure what exactly? Should I quantify human emotion or chords or something?
- Do this, but for words in poetry to find how authors use words differently

https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=music+representation+in+vector+space&btnG=

Relevant papers I found by this google search:

https://www.google.com/search?q=QUERY-BY-BLENDING%3A+A+MUSIC+EXPLORATION+SYSTEM+BLENDING+LATENT+VECTOR+REPRESENTATIONS+OF+LYRIC+WORD%2C+SONG+AUDIO%2C+AND+ARTIST+dataset+used&rlz=1C1GCEB_enCA1075CA1075&oq=QUERY-BY-BLENDING%3A+A+MUSIC+EXPLORATION+SYSTEM+BLENDING+LATENT+VECTOR+REPRESENTATIONS+OF+LYRIC+WORD%2C+SONG+AUDIO%2C+AND+ARTIST+dataset+used&aqs=chrome..69i57j69i58.5641j0j7&sourceid=chrome&ie=UTF-8

What are the following:

Q log-magnitude spectrogram? Log-mel

URL list from Monday, Oct. 2 2023 2:41 AM

To copy this list, type [Ctrl] A, then type [Ctrl] C.

What's wrong with CNNs and spectrograms for audio processing? | by Daniel Rothmann | Towards Data Science

<https://towardsdatascience.com/whats-wrong-with-spectrograms-and-cnns-for-audio-processing-311377d7ccd>

classifying music based on notes and keys machine learning - Google Search

https://www.google.com/search?q=classifying+music+based+on+notes+and+keys+machine+learning&og=classifying+music+based+on+notes+and+keys+machine+learning&gs_lcrp=EgZjaHJvbWUyBggAEEUYOdIBCTEwMzI0ajFgNKgCALACAA&sourceid=chrome&ie=UTF-8

NLP-based music processing for composer classification | Scientific Reports

<https://www.nature.com/articles/s41598-023-40332-0>

Mahieu-DetectingMusicalKeyWithSupervisedLearning-report.pdf

<https://cs229.stanford.edu/proj2016/report/Mahieu-DetectingMusicalKeyWithSupervisedLearning-report.pdf>

Google's AI Music Datasets: MusicCaps, AudioSet and MuLan

<https://www.audiocipher.com/post/musiccaps-audioset-mulan#:~:text=MuLan%20music%20dataset%3F-,What%20are%20music%20datasets%3F,of%20a%20piece%20of%20music.>

NLP-based music processing for composer classification

file:///Users/catherinebalajadia/Downloads/s41598-023-40332-0.pdf

GitHub - SirawitC/NLP-based-music-processing-for-composer-classification: Code for the paper NLP-based music processing for composer classification

<https://github.com/SirawitC/NLP-based-music-processing-for-composer-classification>

mulan.pdf

<http://www.joonseok.net/papers/mulan.pdf>

Visualizing large

<https://towardsdatascience.com/encoding-data-with-transformers-d14445e96ead>

Keras tutorials:

<https://stackoverflow.com/questions/43715047/how-do-i-get-the-weights-of-a-layer-in-keras>

https://www.google.com/search?q=encoder+decoder+keras&oq=encoder+decoder+keras&gs_lcrp=EgZjaHJvbWUyBggAEEUYOTIICAQABgWGB4yCAGCEAAYFhgeMggIAxAGBYHjIIICAQQABgWGB4yCAGFEAAYFhgeMggIBhAAGBYHjIIICAcQABgWGB4yCAGIEAAYFhgeMgoICRAAGIYDGloF0gEINTgwN2owajmoAgCwAgA&sourceid=chrome&ie=UTF-8

<https://blog.keras.io/a-ten-minute-introduction-to-sequence-to-sequence-learning-in-keras.html>

<https://blog.keras.io/building-autoencoders-in-keras.html>

<https://keras.io/examples/generative/vae/>