Audio Engineering Society

Convention Paper 10470

Presented at the 150th Convention
2021 May 25–28, Online

Sequential Modeling of Temporal Timbre Series for Popular Music Sub-Genres Analyses Using Deep Bidirectional Encoder Representations from Transformers

Shijia Geng[1], Gang Ren[1,2], Xu Pan[2], Joel Zysman[1], and Mistu Ogihara[1,2]

[1] *Institute for Data Science and Computing, University of Miami*
[2] *Department of Computer Science, University of Miami*

Correspondence should be addressed to Shijia Geng (`s.geng@umiami.com`)

ABSTRACT

The timbral analysis from spectrographic features of popular music sub-genres (or micro-genres) presents unique challenges to the field of the computational auditory scene analysis, which is caused by the adjacencies among sub-genres and the complex sonic scenes from sophisticated musical textures and production processes. This paper presents a timbral modeling tool based on a modified deep learning natural language processing model. It treats the time frames in spectrograms as words in natural languages to explore the temporal dependencies. The modeling performance metrics obtained from the fine-tuned classifier of the modified Deep Bidirectional Encoder Representations from Transformers (BERT) model show strong semantic modeling performances with different temporal settings. Designed as an automatic feature engineering tool, the proposed framework provides a unique solution to the semantic modeling and representation tasks for objectively understanding of subtle musical timbral patterns from highly similar musical genres.

1 Introduction

Popular music is signified by its timbre: the soundscape that defines its culture which can be best observed from the timbral factors including the arrangement of musical instrumentation, compositional patterns, vocal techniques, and mixing/production processes [1–8]. Because of the complex integration of these factors, observing sub-genre timbral signatures directly from mixed-down audio is challenging [2, 3]. Figure 1 shows several spectrograms plotted from popular song segments from typical sub-genres. While listening to these songs, we can readily hear the emotion and style connected to their sub-genres. However, when we look at their "busy" spectrograms, the cognition of emotions and stylistic patterns (audio-induced structures from our subjective perception) are very difficult to be attached (located and interpreted) from the spectrogram.

When applying conventional musical timbre analyses tools [9–13] to popular sub-genres, most existing genre-discriminatory spectrographic features cannot represent the sub-genre differences under the umbrella of popular music. Existing timbral features are adequate for distinguishing musical genre categories with broader stylistic separation margins such as the contrast between classical and popular music [11, 12]. Figure 2 shows the spectrograms from several typical classical music sub-genres. All these spectrograms are very different from popular music genres in the spectral energy distribution and in the regularity of rhythmic patterns [2, 3]. Popular music shows more noise-like spectral energy distribution as an evenly spread of sonic energy across frequency bands, while classical music shows stronger energy concentration at lower frequencies [9, 10]. Popular music spectrograms show more steady energy components over the time, while classical music can be quiet for a moment and then active for another moment. Of course, popular music is strongly correlated with strong and regular beat patterns, as sequentially repeating cells observed in the spectrogram [2, 3].

Timbral descriptors derived from these spectrographic signatures are strongly effective for categorizing "broad" genres [14–21]. But when we apply them to sub-genres, their discriminative functions no longer work. Timbral pattern differences among popular music or classical music sub-genres (differences among sub-figures inside Figure 1 or Figure 2) are more difficult to be quantified with existing timbral descriptors because the sub-genres are continuously evolving and absorbing each others [22–24]. The differences among sub-genres are more nuanced and ambiguous and thus a different set of timbral analyses tools is essential for understanding more subtle timbral information structures.

To capture the timbral differences among sub-genres, we implement a framework to sample successive spectral templates from audio and train a modified deep natural language processing model to represent the timbral information encoded in a sequence of spectral templates. The timbral pattern from the spectrogram is sampled as successive timbral templates with fixed time lengths (e.g., 5 seconds or 10 seconds). These timbral templates serve as the basic processing unit for timbral pattern recognition because humans can reliably identify popular music sub-genres from this time span. These timbral templates are treated as natural language sentences or segmented paragraphs and the short-time frames (spectral analysis windows) inside each template are treated as words. Then we use the sub-genre categories as the training labels to a natural language processing model to understand the semantic mapping between sequences of spectral energy distribution in spectrogram time frames and the sub-genre labels.

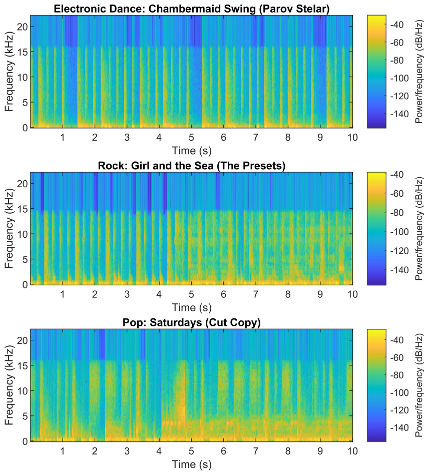

Fig. 1: Spectrograms of song segments from typical popular music sub-genres. The spectrograms from these sub-genres are highly similar.

We select the Bidirectional Encoder Representations from Transformers (BERT) natural language processing model for its lightweight processing and high predictive analysis performance on many information retrieval tasks [25]. BERT is a pre-trained model that is fine-tuned to implement different natural language processing tasks (More details in Sec. 2.1). For our study, we pre-train a modified BERT model to capture the cohesiveness of short popular music sequences and fine-tune the model for a sub-genre classification task. The result section shows that we achieve high performance predictive analysis metrics for this semantic mapping task. We conduct three experiments with different combinations of Short-time Fourier transform (STFT) window size and sequence length: 1) 1,024 window size, 5s sequence; 2) 1,024 window size, 10s sequence; 3) 2,048 window size, 10s sequence. These experiments show that the modified BERT model successfully absorbs and represents the timbral pattern connected to popular music's sub-genres.

The proposed framework does not depend on manually crafted timbral features, instead, it utilizes the automatic feature engineering capacity of the deep natural language processing model to form semantic links directly from spectrographic energy distributions. The strong predictive analysis results as reported in the following result section demonstrate the effectiveness on the semantic mapping and related representational learning tasks. Our proposed framework provides a first

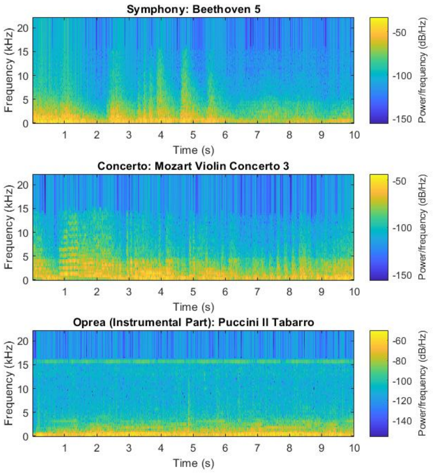

Fig. 2: Spectrograms of audio segments from typical classical music sub-genres. These spectrograms are very different from the spectrograms of popular music.

step towards understanding the sub-genre differences in the popular music. Different from existing timbral descriptors that focus on the global timbre (without considering the temporal evolution of spectrographic features in music phrase time span), our proposed framework is based on sequential modeling and emphasizes the temporal aspects of "dynamic" timbre and thus provides an unique tool for a microscopic investigation of timbral nuances connected to musical concepts and human music cognition.

2 Methods

2.1 Using Deep Natural Language Model for Musical Timbral analyses

The BERT language representation model [25] and other Transformer-based models [26, 27] achieve state-of-the-art results on multiple language tasks. The essential component of these models is the Transformer encoder which aggregates the input sequence based on the "similarities" among elements that constitute the sequence [28]. Unlike the recurrent neural network (RNN) sequential encoder that requires data processed in order, the Transformer encoder combines the input elements based on their linear mappings with the Attention mechanism [28]. With layers of Transformer encoders learning subtle characteristics of natural languages, the model preserves the horizontal concurrent processing that facilitates computations.

As words appear consecutively in natural languages, musical events occur sequentially in music performances to convey information. The Transformer structure has been recently adapted in the cutting-edge music generation models such as Music Transformer [29] and Jukebox [30]. Both models use Transformer with autoregression to generate music. It is reasonable to use the previous information during the generation process since music is delivered in sequence. However, for understanding music, both previous and further information should be considered because the elements in a music sequence are cohesive. The consideration leads to the use of the BERT model which incorporates context from both directions for our study. We hypothesize the music recognition will substantially benefit from the concurrent use of the two directions.

2.2 Audio-Based Representation of Music

An essential choice to make in music studies is the representation of the music. The aforementioned Music Transformer model processes the MIDI data of piano performances [29]. The symbolic MIDI data provides accurate control information and it can be used to represent music with instruments which timbres are well defined. However, it does not provide enough information when representing popular music that involves intricate sounds and effects. The other aforementioned model, the Jukebox model, embeds raw audio data into segments of discrete vectors [30]. The embedding action of the Jukebox model is reminiscent of the STFT process in a way that each dimension of the vector space encodes a certain aspect of the raw audio segment.

Noticing the similarity in the embedding, we question if the STFT coefficients as input can replace the word embedding in the original BERT model. Words in natural languages do not have inherent representations that describe their relationships among each other. The word embedding makes it possible to assign high-dimensional numerical vectors to words in a natural language through some analysis of their co-occurrences [31]. The BERT model utilizes such an embedding to further quantify the association among words in sentences. Short-time segments of music, on the other hand, may not need an embedding process like the word embedding, since they have inherent representations in the frequency domain. There is a large body of research that studies the use of STFT for music classification and similarities [32–35].

2.3 Preprocessing Steps and Configurations

We collected 23,634 songs from five popular music subgenres in MP3 format (alternative: 3,394; electronica: 4,368; pop: 5,318; rap: 5,113; rock: 5,441), resampled them to 8,000Hz and applied STFT. For experiment 1 and 2, we selected window size 1,024 (about 128ms) which generates 513 coefficients in magnitude for each window, and there is no overlap between adjacent windows. We ignored the DC components and used the sets of 512 coefficients (15.6-8,000Hz) as the elements for each input sequence. For experiment 3 with window size 2,048 (about 256ms), we got sets of 1,024 coefficients (7.8-8,000Hz) as sequence elements. The sets of coefficients are also the labels for the masked language modeling (MLM) pre-training task.

2.4 Modifications of the BERT Model

Our model is modified based on the Huggingface Transformers BERT implementation [36]. We eliminated the word embedding and fed STFT coefficients into the Transformer encoders directly with the position embeddings. For experiment 1, each input sequence includes 40 sets of coefficients (about 5s) and one set at the beginning with 512 same values. The value chosen, 1.6, is the average of all coefficients from the dataset. The beginning set corresponds to the classification token ([CLS]) that is added in front of every input example for the original BERT model [25]. The sequence hop size, which is the distance from the start of one sequence to the start of the next sequence, has 20 elements that is about 2.5 seconds in time. Similarity, the inputs of experiment 2 are 81-element sequences (about 10s) with 20-element (about 2.5s) sequence hop size, while the inputs of experiment 3 are 41-element sequences (about 10s) with 10-element (about 2.5s) sequence hop size.

The hidden size, which is the dimension of the Transformer encoder layer, is the number of STFT coefficients (512 for experiment 1 and 2; 1,024 for experiment 3). The intermediate size, which is the dimension of the Feed-Forward layer, is twice the number of STFT coefficients, and it corresponds to the window size (1,024 for experiment 1 and 2; 2,048 for experiment 3). For all three experiments, the number of the attention heads for each attention layer is 4, and the number of hidden layers is 4.

2.5 Pre-training with the MLM Task

For experiment 1, we obtained a dataset with shape 2,257,370 × 41 × 512. For experiment 2, the dataset shape is 2,197,447 × 81 × 512, and for experiment 3, it is 2,197,447 × 41 × 1,024. With each experiment we used 7/8 of the dataset to conduct the pre-training with the MLM task. Following the original MLM paradigm [25], we chose 15% of the element positions at random, and if one element was chosen, we would: mask it at 80% of the time; replace it with a random selected set of STFT coefficients from the pre-training dataset at 10% of the time; keep it unchanged at 10% of the time. The pre-training loss is the average of the squared Euclidean distance between outputs and targets for the masked elements.

We trained the model with 1,024 batch size and 50 epochs, and applied the Adam optimizer ($\beta 1 = 0.9$, $\beta 2 = 0.999$) with learning rate 1×10^{-4} which increases from 0 over the first 3,500 steps and decreases linearly to 0 after that.

2.6 Fine-tuning with the Sub-Genre Classification Task

For each experiment we split the remaining 1/8 data into training and testing datasets with 4:1 ratio, and did a fine-tuning sub-genre classification task. As with the original BERT text classification, the output corresponding to the first element was fed into an extra classification layer with weights $W \in \mathbb{R}^{5 \times hidden\_size}$, where 5 is the number of sub-genres.

The sub-genre classification training was conducted with 128 batch size and 100 epochs. The learning rate is 5×10^{-5}, and the number of warmup steps is 500.

We did the testing using the rest of the sequences and reported the confusion matrices. The metrics of each genre were calculated from the confusion matrices with the following formulas:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$
$$Precision = \frac{TP}{TP+FP}$$
$$Recall = \frac{TP}{TP+FN}$$
$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

TP - True Positive, FP - False Positive, TN - True Negative, FN - False Negative.

3 Results

Figure 3 shows the pre-training loss (average of the squared Euclidean distance for the masked elements) versus the number of epochs for the three experiments. The loss of the 2,048 window size experiment is about twice the loss of each 1,024 window size experiment. Figure 4 shows the fine-tuning classification training loss (cross entropy loss) versus the number of epochs for all three experiments. From the figure we can see the experiment of 10s sequences with window size 2,048 has the fastest convergence while the experiment of 5s sequences with window size 1,024 has the slowest convergence.

Table 1 lists the confusion matrices of the fine-tuning classification testing for the three experiments.The shortened keys alter. and elec. represent Alternative and Electronica respectively. From the metrics tables 2 to 6, we can see that the model has the best performance on Rap among the five genres. The average accuracy reaches to 94.5% with 2,048 window size and 10s sequence length. The Rap minimum metrics show that distinguishing Rap from Electronica is not as good as distinguishing Rap from other genres. The model's second best classified genre is Electronica, which average accuracy reaches to 89.8% with 2,048 window size and 10s sequence length. Both Rock and Alternative reach to more than 80% average accuracy for 5s sequences with 1,024 window size and for 10s sequences with 2,048 window size. The model's ability to separate Rock from Alternative decreases when choosing 1,024 window size and 10s sequence length. The pop genre classification performs the worst among the five genres, and Rock and Alternative are the most confusing genres with Pop.

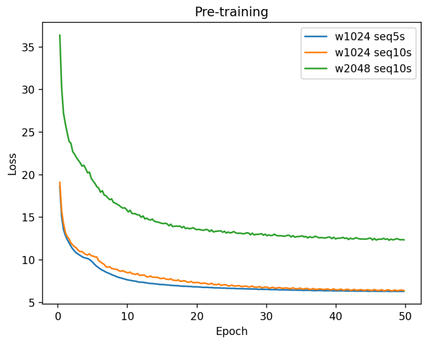

Fig. 3: Pre-training loss versus number of epochs. We observe satisfactory convergence of the training process for all model configurations.

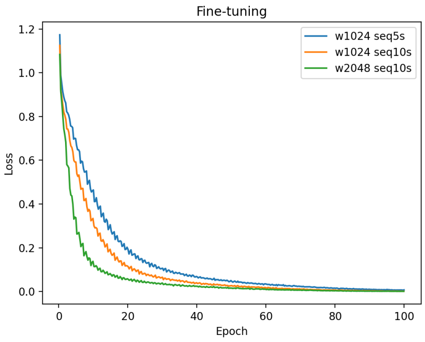

Fig. 4: Fine-tuning classification training loss versus number of epochs. We observe satisfactory convergence of the fine-tuning process for all model configurations.

4 Discussion

The model has the best performance when classifying Rap, which corresponds to the human perception that Rap is more distinguishable than other genres for its intensive vocals and well-structured beats. Rap also has very different instrumental selections/arrangements from the other music genres . For example, the melodic part of Rap is usually the repetition of a simple monophonic tune or parallel tunes without harmonic supporting relationships, and there is no structural voice-leading features between instruments [1, 6]. These factors contribute to the directness of the Rap's stylistic taste and discriminate it from the other genres.

The Rap genre is sometimes confused with the Electronica genre. This might be due to the fact that the musicians in both genres usually borrow from each other in instrumentation and compositional concepts. Actually, many composers and producers are comfortable at both genres. Recently these two genres share a lot of music instruments and production tools (drum machines, sequencers, etc.) in a process of mutual absorption [2, 3]. In Electronica, synthesizers are usually used in a slightly different way to produce futuristic sounds, which can be easily picked up by human listeners for the timbral innovation [5]. In contrast, Rap usually applies established (or familiar) presets (sound generation and modification routines stored as a predefined program in instruments or software) [22, 23]. However, because many of these futuristic or familiar sounds are generated from the same instrument using similar signal paths, the stylistic differences are too nuanced to be automatically discovered by machine learning systems. For these two genre categories, a musical feature based approach or its hybrid implementation with automatic feature engineering approaches might be more appropriate.

The alternative music is typically regarded as more eclectic, original, or challenging than most popular music, and it is often distributed by independent record labels [37]. The difference between the alternative music and the mainstream music such as pop and rock could be subtle, and sometimes cannot be perceived through the auditory features [22, 23]. Alternative is a rather dynamic genre because other genres are consistently absorbing new sonic elements from the experimental dimensions of Alternative [2]. In fact, many innovative aspects from earlier Alternative have become familiar sound features in the other genres. Many Alternative songs acquired the genre label at the specific time of their productions and could be labeled into other genres if produced today [22, 23]. Of course, Rock is a genre that is open for sonic innovations (adapting new instruments, absorbing new musical concepts, etc.) and the experimental elements of Alternative always influence Rock immediately, which causes their myriads of stylistic similarities [4, 8]. The results reflect the difficulty of distinguishing Alternative from Rock, and it is interesting to note that the performance gets worse for 10s sequences than 5s sequences with 1,024 window size.

In future studies, we could examine more combinations of different window sizes and sequence lengths, and search for the optimal temporal settings. We could also test the model with more layers. In our current study we only use 4 hidden layers to obtain a reasonable training time, while the original BERT models have deeper structures ($BERT_{BASE}$ has 12 layers and $BERT_{LARGE}$ has 24 layers). Moreover, the BERT model aggregates the information of an input sequence based on the "similarities" among elements. Although STFT indicates certain relationships in the frequency domain, it does not consider the musical harmony. It is difficult to assign the harmonic distances manually, but we could add an embedding layer to capture the harmonic and some hidden features. Another consideration is how to define elements. In this study, we arbitrarily choose 1,024 and 2,048 window sizes which do not have "semantic" meanings. In the future, we could use notes or beats to separate the musical elements, which would make them more comparable with the words in natural languages.

In addition, we would like to extend the framework to more sub-genre or micro-genre categories. For example, classical genres also present many similar challenges for works composed at adjacent but style-defining time period, such as musical impressionism as “searching for new sound”. More subtle differences between composers or performers present more sophisticated challenges to our proposed framework. Another future research direction could be to explain the insights learned from (or the patterns embedded in) the trained deep learning models. Recent works in deep learning based representations provide “inside-the-black-box” approaches that can summarize the automatically learned features into explicit feature descriptors. A combination and comparison of conventional musical timbral descriptors with learned feature descriptors could also be an interesting research topic for further understanding deep learning based timbral models and the timbral semantic mappings.

5 Summary

The sub-genres inside popular music are highly similar in spectrographic features but strongly perceptible from human cognition. Conventional timbral analyses tools for broad genre categories cannot capture the subtle differences among these sub-genres. In this paper, a sequential timbral pattern modeling framework based on the deep learning natural language processing model is presented. We modified the BERT language representation model and applied it to music. Each musical "sentence" is composed of "tokens" that are STFT coefficients. After pre-training and fine-tuned classification training with different "token" sizes (128ms and 256ms) and "sentence" lengths (5s and 10s), we obtained best average classification accuracies as follows: 85.8% for Alternative, 89.8% for Electronica, 80.8% for Pop, 94.5% for Rap and 82.9% for Rock.

The future works related to this paper are exploring more temporal and structural configurations, extending the framework to more sub-genre categories, studying the insights learned from the trained deep learning models, and integrating the learned feature descriptors with conventional musical timbral descriptors.

Table 1: Genre Classification Confusion Matrices.

true	predicted (w1024 seq5s)				
	alter.	elec.	pop	rap	rock
alter.	**2491**	1104	1616	269	2171
elec.	844	**8416**	1503	959	712
pop	1121	1493	**5449**	800	2713
rap	210	982	1011	**9622**	312
rock	1532	555	2972	236	**6752**
	predicted (w1024 seq10s)				
alter.	**3823**	2341	4094	501	5603
elec.	469	**10340**	1724	891	660
pop	491	813	**3933**	490	1813
rap	108	957	858	**9272**	232
rock	474	262	1268	144	**3375**
	predicted (w2048 seq10s)				
alter.	**675**	226	273	56	567
elec.	937	**9952**	1532	939	735
pop	1615	1689	**7273**	1068	4429
rap	299	1300	906	**14345**	417
rock	862	262	965	86	**3528**

Table 2: Metrics for Alternative. (Acc. - Accuracy, Prec. - Precision, Rec. - Recall)

	w1024 seq5s			w1024 seq10s			w2048 seq10s		
	min	max	avg	min	max	avg	min	max	avg
Acc.	rock: 71.4%	rap: 96.2%	81.7%	rock: 54.2%	rap: 95.6%	74.0%	rock: 74.6%	rap: 97.7%	85.8%
Prec.	rock: 61.9%	rap: 92.2%	74.5%	pop: 88.6%	rap: 97.3%	91.0%	pop: 29.5%	rap: 69.3%	46.1%
Rec.	rock: 53.4%	rap: 90.3%	68.4%	rock: 40.6%	rap: 88.4%	59.8%	rock: 54.3%	rap: 92.3%	73.2%
F1	rock: 57.4%	rap: 91.2%	71.3%	rock: 55.7%	rap: 92.6%	71.0%	pop: 41.7%	rap: 79.2%	55.8%

Table 3: Metrics for Electronica.

	w1024 seq5s			w1024 seq10s			w2048 seq10s		
	min	max	avg	min	max	avg	min	max	avg
Acc.	pop: 82.2%	rock: 92.3%	87.4%	alter.: 83.4%	rock: 93.7%	88.4%	pop: 84.2%	rock: 93.1%	89.8%
Prec.	pop: 84.9%	rock: 93.8%	89.2%	alter.: 81.5%	rock: 97.5%	90.8%	pop: 85.5%	alter.: 97.8%	92.3%
Rec.	pop: 84.8%	rock: 92.2%	89.4%	pop: 85.7%	alter.: 95.7%	91.9%	pop: 86.7%	rock: 93.1%	90.6%
F1	pop: 84.9%	rock: 93.0%	89.3%	alter.: 88.0%	rock: 95.7%	91.2%	pop: 86.1%	rock: 95.2%	91.4%

Table 4: Metrics for Pop.

	w1024 seq5s			w1024 seq10s			w2048 seq10s		
	min	max	avg	min	max	avg	min	max	avg
Acc.	rock: 68.2%	rap: 89.3%	78.5%	alter.: 62.8%	rap: 90.7%	77.2%	rock: 66.7%	rap: 91.6%	80.8%
Prec.	rock: 64.7%	rap: 84.3%	76.1%	alter.: 49.0%	rap: 82.1%	69.1%	elec.: 82.6%	alter.: 96.4%	89.0%
Rec.	rock: 66.8%	rap: 87.2%	78.8%	rock: 68.4%	rap: 88.9%	82.3%	rock: 62.2%	rap: 87.2%	78.1%
F1	rock: 65.7%	rap: 85.8%	77.5%	alter.: 63.2%	rap: 85.4%	74.0%	rock: 72.9%	alter.: 88.5%	82.8%

Table 5: Metrics for Rap.

	w1024 seq5s			w1024 seq10s			w2048 seq10s		
	min	max	avg	min	max	avg	min	max	avg
Acc.	pop: 89.3%	rock: 96.8%	93.1%	pop: 90.7%	rock: 97.1%	93.7%	elec.: 91.6%	alter.: 97.7%	94.5%
Prec.	elec.: 90.9%	rock: 97.6%	94.5%	elec.: 91.2%	rock: 98.5%	94.9%	pop: 93.1%	alter.: 99.6%	96.5%
Rec.	pop: 90.5%	alter.: 97.9%	94.0%	elec.: 90.6%	alter.: 98.8%	94.6%	elec.: 91.7%	alter.: 98.0%	95.2%
F1	elec.: 90.8%	alter.: 97.6%	94.3%	elec.: 90.9%	rock: 98.0%	94.7%	elec.: 92.8%	alter.: 98.8%	95.8%

Table 6: Metrics for Rock.

	w1024 seq5s			w1024 seq10s			w2048 seq10s		
	min	max	avg	min	max	avg	min	max	avg
Acc.	pop: 68.2%	rap: 96.8%	82.2%	alter.: 54.2%	rap: 97.1%	78.8%	pop: 66.7%	rap: 97.3%	82.9%
Prec.	pop: 71.3%	rap: 95.6%	83.3%	alter.: 37.6%	rap: 93.6%	70.0%	pop: 44.3%	rap: 89.4%	75.7%
Rec.	pop: 69.4%	rap: 96.6%	85.0%	pop: 72.7%	rap: 95.9%	87.3%	pop: 78.5%	rap: 97.6%	87.4%
F1	pop: 70.4%	rap: 96.1%	84.1%	alter.: 52.6%	rap: 94.7%	76.0%	pop: 56.7%	rap: 93.3%	80.2%

References

[1] Campbell, M., *Popular music in America: The beat goes on*, Cengage Learning, 2018.

[2] Fink, R., Latour, M., and Wallmark, Z., *The relentless pursuit of tone: Timbre in popular music*, Oxford University Press, 2018.

[3] Scotto, C., Smith, K. M., and Brackett, J., editors, *The Routledge Companion to Popular Music Analysis: Expanding Approaches*, Routledge, 2020.

[4] Clauhs, M., Powell, B., and Clements, A. C., *Popular Music Pedagogies: A Practical Guide for Music Teachers*, Routledge, 2020.

[5] Starr, L. and Waterman, C., *American Popular Music: From Minstrelsy to MP3*, Oxford University Press, 2017.

[6] Shahriari, A., *Popular World Music*, Routledge, 2017.

[7] Brackett, D., *The pop, rock, and soul reader: histories and debates*, Oxford University Press, USA, 4 edition, 2019.

[8] Garofalo, R. and Waksman, S., *Rockin'out: popular music in the USA*, Pearson, 6 edition, 2014.

[9] Sethares, W. A., *Tuning, timbre, spectrum, scale*, Springer Science & Business Media, 2005.

[10] Beauchamp, J. W., *Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music. Modern Acoustics and Signal Processing*, Springer, 2007.

[11] Müller, M., *Fundamentals of Music Processing: Using Python and Jupyter Notebooks*, Springer, 2 edition, 2021.

[12] Müller, M., *Fundamentals of music processing: Audio, analysis, algorithms, applications*, Springer, 2015.

[13] Steiglitz, K., *A digital signal processing primer, with applications to digital audio and computer music*, Dover Publications, 2020.

[14] Campbell, M., Gilbert, J., and Myers, A., *The Science of Brass Instruments*, Springer International Publishing, 2021.

[15] Meyer, J., *Acoustics and the performance of music: Manual for acousticians, audio engineers, musicians, architects and musical instrument makers*, Springer Science & Business Media, 2009.

[16] Giannakopoulos, T. and Pikrakis, A., *Introduction to Audio Analysis: a MATLAB® approach*, Academic Press, 2014.

[17] Deutsch, D., *The Psychology of music*, Academic Press, 3 edition, 2012.

[18] Tan, S.-L., Pfordresher, P., and Harré, R., *Psychology of music: From sound to significance*, Routledge, 2017.

[19] Ashley, R. and Timmers, R., *The Routledge Companion to Music Cognition*, Routledge, 2019.

[20] Hodges, D., *Music in the human experience: An introduction to music psychology*, Routledge, 2019.

[21] Toivonen, I., Csuri, P., and Van Der Zee, E., *Structures in the Mind: Essays on Language, Music, and Cognition in Honor of Ray Jackendoff*, MIT Press, 2015.

[22] Brackett, D., *Categorizing sound: Genre and twentieth-century popular music*, Univ of California Press, 2016.

[23] Borthwick, S. and Moy, R., *Popular music genres: An introduction*, Routledge, 2020.

[24] Williamon, A., Ginsborg, J., Perkins, R., and Waddell, G., *Performing music research: methods in music education, psychology, and performance science*, Oxford University Press, 2021.

[25] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K., "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[26] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V., "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in neural information processing systems*, pp. 5753–5763, 2019.

[27] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V., "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[28] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I., "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[29] Huang, C.-Z. A., Vaswani, A., Uszkoreit, J., Simon, I., Hawthorne, C., Shazeer, N., Dai, A. M., Hoffman, M. D., Dinculescu, M., and Eck, D., "Music transformer: Generating music with long-term structure," in *International Conference on Learning Representations*, 2018.

[30] Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I., "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020.

[31] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L., "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[32] Tzanetakis, G. and Cook, P., "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing*, 10(5), pp. 293–302, 2002.

[33] Li, T. and Ogihara, M., "Toward intelligent music information retrieval," *IEEE Transactions on Multimedia*, 8(3), pp. 564–574, 2006.

[34] Costa, Y. M., Oliveira, L. S., and Silla Jr, C. N., "An evaluation of convolutional neural networks for music classification using spectrograms," *Applied soft computing*, 52, pp. 28–38, 2017.

[35] Yu, Y., Luo, S., Liu, S., Qiao, H., Liu, Y., and Feng, L., "Deep attention based music genre classification," *Neurocomputing*, 372, pp. 84–91, 2020.

[36] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M., "Transformers: State-of-the-Art Natural Language Processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Association for Computational Linguistics, Online, 2020.

[37] Merriam-Webster, "alternative music," in *Merriam-Webster.com dictionary*, n.d., retrieved 12 Mar. 2021.