2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)

2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) | 979-8-3503-0067-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/APSIPAASC58517.2023.10317473

Learning Multifaceted Self-Similarity for Musical Structure Analysis

Tsung-Ping Chen*, Li Su†, and Kazuyoshi Yoshii*
*Graduate School of Informatics, Kyoto University, Japan
{tchen, yoshii}@sap.ist.i.kyoto-u.ac.jp
†Institute of Information Science, Academia Sinica, Taiwan
lisu@iis.sinica.edu.tw

***Abstract*—This paper describes a data-driven music structure analysis (MSA) method that performs segmentation and clustering of musical sections for a music signal. Since the intra-section homogeneity and inter-section difference are important clues for MSA, most studies on MSA have focused on self-similarity matrices (SSMs) computed from various acoustic features of a music signal. The performance of this approach, however, might be limited because the acoustic features used for computing SSMs are designed manually, and multiple SSMs are often integrated in a heuristic manner. To overcome these limitations, we propose a method that learns latent features useful for MSA with a stack of convolution-augmented multi-head self-attention (CAMHSA) layers that compute and fuse multiple self-attention maps representing multifaceted self-similarity. The estimated features are then clustered into an appropriate number of sections with a Gaussian mixture model (GMM). In the segmentation and clustering tasks, the proposed method outperformed baseline methods based on hand-crafted SSMs. In particular, it achieved state-of-the-art performance on the segmentation task. We found that the internal attention maps represent the section boundaries at the fine and course levels.**

I. Introduction

Audio-based music structure analysis (MSA) aims to split a musical recording into musically-meaningful segments based on the homogeneity and heterogeneity of the musical contents, and then assign a letter (e.g., A, B, or C) or functional label (e.g., verse, chorus, or bridge) to each of the segments. The process of MSA can be divided into two sub-tasks: *segmentation* and *clustering*. The former aims to detect the boundaries of musical sections. The latter aims to categorize the segments into multiple groups. In the field of music information retrieval (MIR), the two tasks are often tackled separately.

Given the repetitive nature of music, the self-similarity matrix (SSM) has been a common feature for tackling MSA [1]–[3], as repetitions and musically homogeneous parts typically result in diagonals and block patterns on the SSM. Nevertheless, the choice of audio representation for computing the SSM has a crucial impact on the performance of MSA. For instance, using timbre- or harmony-related features might yield two different SSMs highlighting possibly two distinct temporal structures for the same audio signal, as depicted in Fig. 1. Considering the multiple aspects of similarity, one may compute multiple SSMs from various kinds of acoustic features [4]. Several techniques have also been proposed for fusing different aspects of similarity [5], [6]. Such an approach, however, has a performance limitation due to a finite number of hand-crafted features used for computing SSMs. In addition, segmentation and clustering algorithms should be tailor-made according to the characteristics of each SSM.

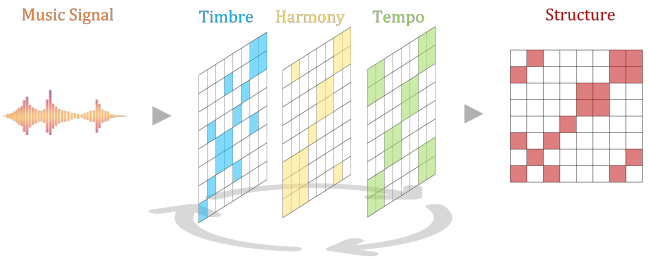


Fig. 1: The self-similarity of a music signal varies with the aspect across musical elements (e.g., timbre, harmony, and tempo). Learning a multifaceted view of self-similarity is thus crucial for music structure analysis.

To overcome these limitations, we propose an MSA method that performs segmentation and clustering on latent features learned by a deep neural network (DNN) with a convolution-augmented multi-head self-attention (CAMHSA) mechanism, an extension of the multi-head self-attention (MHSA) mechanism [7]. The main motivation for our study is that the self-attention maps computed internally through a stack of MHSA layers (i.e., the transformer encoder [7]) are expected to represent multifaceted self-similarity at different abstract levels. Our method is capable of learning to fuse such self-attention maps with additional convolutional operations in each layer. Specifically, the DNN is trained in a supervised manner to output latent features with which section boundaries are detected accurately and from which the derived SSM is made close to the ground-truth SSM. The latent features are then used for an unsupervised clustering method based on the Gaussian mixture model (GMM).

The main contributions of this study are twofold. We theoretically and experimentally show the effectiveness of the CAMHSA in learning a multifaceted self-similarity for MSA. We achieved a state-of-the-art performance of segmentation.

II. Related Work

This section briefly introduces audio-based music structure analysis and the self-attention mechanism.

A. Music Structure Analysis

Previous research in MIR often tackles MSA from either the segmentation or the clustering aspects. For the segmentation task, a bunch of deep learning approaches is proposed to identify musical boundaries within a given piece [8]–[12]. This task is often formulated as a binary classification problem, where spectrogram-based features, such as chromagram and mel-spectrogram, and SSMs derived from these features are commonly used as the inputs. Considering the size of SSMs grows quadratically with the duration of the corresponding audio signal, self-similarity lag matrices (SSLMs) [13] can be used instead to represent the similarities of each audio frame to a limited number of the preceding frames.

In contrast, the clustering task is typically approached as a clustering problem. This involves utilizing frame-wise acoustic features or affinity matrices (either hand-crafted or learned) as inputs for a clustering algorithm or a matrix decomposition method in order to group the audio frames [5], [6], [14]. Alternatively, this task can be formulated as a classification problem, and a model is employed to label each audio frame with a predefined vocabulary [15]. With manually annotated datasets such as the SALAMI dataset [16] and the Beatles dataset (released originally in the Isophonics dataset [17]), previous studies mostly applied supervised learning frameworks to the segmentation and clustering tasks. Recently, contrastive learning approaches are also explored in learning audio representations for MSA in an unsupervised or self-supervised fashion [18], [19]. For a detailed review of the audio-based MSA in MIR, we refer readers to [20].

B. Self-Attention Mechanism

The intra-attention mechanism [21], [22], also known as the *self-attention* (SA) [7], is proposed to encode compositional relationships between a set of elements (e.g., words of a sentence), and has demonstrated its effectiveness in various research fields such as natural language processing (NLP) [23]–[25] and computer vision (CV) [26]–[28]. In the field of MIR, several studies have utilized the SA [29]–[31]. However, the potential of the SA for MSA has yet to be investigated.

Formally, given a sequence of t elements with d-dimensional features, $\mathbf{X} \in \mathbb{R}^{t \times d}$, the SA yields a new representation of the sequence, i.e., $\mathrm{SA}(\mathbf{X}) \in \mathbb{R}^{t \times d}$, by internally computing an affinity matrix or an *attention map*, $\mathbf{A} \in \mathbb{R}^{t \times t}$, indicating the similarity between each pair of the elements in the sequence as follows:

$$\begin{aligned} \mathrm{SA}(\mathbf{X}) &= \mathbf{A} f_v(\mathbf{X}) \\ &= \mathrm{softmax}(f_q(\mathbf{X}) f_k(\mathbf{X})^{\mathrm{T}}) f_v(\mathbf{X}), \end{aligned} \tag{1}$$

where $f_{\{q,k,v\}} \colon \mathbb{R}^{d_{in}} \to \mathbb{R}^{d_{out}}$ are learnable layers projecting the *query*, *key*, and *value* [7] (all denote the same input for self-attention) to a common latent space in which similarity is measured and the input is represented. With this definition, the attention map can be regarded as a generalization of a softmax-normalized SSM computed in the latent space, and it becomes an exact SSM if $f_q(\cdot) = f_k(\cdot)$.

Based on the idea of the SA, the multi-head self-attention (MHSA) mechanism [7] consists of multiple SAs for extracting multifaceted similarity information from various representation subspaces as follows:

$$\mathrm{MHSA}(\mathbf{X}) = [\mathrm{SA}(\mathbf{X}^{(1)}), \cdots, \mathrm{SA}(\mathbf{X}^{(h)})]\mathbf{W}, \tag{2}$$

where $h \in \mathbb{N}^+$ is the number of heads, $\mathbf{X}^{(\mathrm{h}=1:h)} \in \mathbb{R}^{t \times \frac{d}{h}}$ is a division of $\mathbf{X}$ within the feature dimension, $[\ ]$ denotes a matrix concatenation along the feature dimension, and $\mathbf{W} \in \mathbb{R}^{d \times d}$ is a learnable parameter used to fuse information from the attention heads. In practice, the attention heads can be computed in a parallel manner via manipulating the dimensionality of $f_{\{q,k,v\}}(\mathbf{X})$. Let $\hat{f}_{\{q,k,v\}}(\mathbf{X}) \in \mathbb{R}^{h \times t \times \frac{d}{h}}$ be the dimensionality-manipulated tensor of $f_{\{q,k,v\}}(\mathbf{X})$. We can rewrite (2) as follows:

$$\mathrm{MHSA}(\mathbf{X}) = \mathrm{restore}(\hat{\mathbf{A}} \hat{f}_v(\mathbf{X}))\mathbf{W}, \tag{3}$$

where $\hat{\mathbf{A}} = \mathrm{softmax}(\hat{f}_q(\mathbf{X}) \hat{f}_k(\mathbf{X})^{\mathrm{T}})$ and $\mathrm{restore}(\cdot)$ get back the dimensionality from $\mathbb{R}^{h \times t \times \frac{d}{h}}$ to $\mathbb{R}^{t \times d}$.

The MHSA can attend to various relations in different latent spaces by referring to multiple SSMs. Previous studies have pointed out that the attention heads of the MHSA are capable of capturing different types of implicit relationships between the elements of a sequence [32], [33]. The MHSA mechanism is thus favorable for MSA as it can alleviate the feature selection issue with learned features and affinity matrices.

III. Proposed Method

To cope with the feature selection issue when employing SSMs for MSA, we propose the *convolution-augmented multi-head self-attention*, which learns to *construct* and *fuse* multiple affinity matrices. Besides, we tackle the MSA with a two-stage framework. In the first stage, we employ a DNN equipped with the proposed attention mechanism to jointly predict musical boundaries and frame-wise representations for a given input. In the second stage, we obtain structural groups by clustering the frame-wise representations.

A. Data Representation

The input is a standardized audio signal sampled at 32 kHz. Three types of acoustic features are extracted from the signal: *mel-spectrogram*, *chromagram*, and *tempogram* [34]. For mel-spectrogram ($\mathbf{X}_m \in \mathbb{R}^{t \times 80}$), we use a mel-scaled filterbank of 80 triangular filters from 80 Hz to 16 kHz and scale magnitudes logarithmically. For chromagram ($\mathbf{X}_c \in \mathbb{R}^{t \times 12}$), the constant-Q spectrogram of 12 bins per octave is used. For tempogram ($\mathbf{X}_t \in \mathbb{R}^{t \times 384}$), the rhythmic content is encoded via the local auto-correlation of the onset strength envelope. All the features are calculated with *librosa* [35]. To analyze the entire piece once, which is crucial for learning long-term music structure, we compute all three acoustic features with a large hop size of

1600 points and downsample them with a factor of 10 using a median filter, resulting in a frame size of 0.5 sec.

The boundary annotation of each piece is given as a binary sequence ($\mathbf{Y}_b$) where 1 indicates a boundary and 0 otherwise; for the clustering task, the annotations of section names are converted into a finite alphabet ($\mathbf{Y}_s$) and the variation signs (e.g., '′' and '″') are ripped off. For instance, the annotation `[verse,verse′,chorus]` would become `[A,A,B]`.

B. Convolution-Augmented Multi-Head Self-Attention

While the MHSA is capable of capturing various relantionships between a set of elements, the calculations of the attention heads are independent of each other. More precisely, each attention map is computed individually without being related to other attention maps. In fact, it has been demonstrated that the tasks of MSA can benefit from integrating multiple affinity matrices into a united representation [6], [9]. For this reason, we propose to equip the MHSA with convolutional layers to fuse information from multiple attention maps. Concretely, we perform 2-D convolutions on the stack of the attention maps before the softmax function is applied. The convolution-augmented MHSA, denoted by $\mathrm{CAMHSA}(\mathbf{X}) \in \mathbb{R}^{t\times d}$, can thus be formulated as follows (Fig. 2):

$$\mathrm{CAMHSA}(\mathbf{X}) = \mathrm{restore}(\hat{\mathbf{A}}^{*} f_v(\hat{\mathbf{X}}))\mathbf{W}, \tag{4}$$

where $\hat{\mathbf{A}}^{*} = \mathrm{softmax}(\mathrm{conv}(f_q(\hat{\mathbf{X}}) f_k(\hat{\mathbf{X}})^{\mathrm{T}}))$ is an attention map and $\mathrm{conv}(\cdot)$ is a stack of two successive convolutional layers with a layer normalization [36] in between.

In fact, a recent work, i.e., the Conformer [37], has explored the combination of CNNs and the MHSA mechanism for modeling both local and global dependencies of a given sequence. Our approach differs from this work in how convolutions are employed. As shown in Fig. 3, the Conformer employs a 1-D CNN *outside* the MHSA block to capture local dependencies of the output sequence, whereas the CAMHSA performs 2-D convolutions *inside* the MHSA block to merge structural information of different attention maps. By utilizing convolutions, the CAMHSA can learn the correlations among various attention maps, leading to the enhancement of the attention heads.

C. Model Architecture

The model for MSA is composed of three parts, as shown in Fig. 4. The three types of input features are separately encoded by a 2-D CNN. The encoded features, $\mathbf{E}_{\{m,\ c,\ t\}} \in \mathbb{R}^{t\times d}$ for $\mathbf{X}_{\{m,\ c,\ t\}}$ respectively, are fused with another 2-D CNN. The fused feature, $\mathbf{E} \in \mathbb{R}^{t\times d}$, is then fed into a 2-layer transformer encoder with 8 attention heads [7], in which the MHSA mechanism is replaced by the CAMHSA. Finally, the latent representation by the transformer encoder, $\mathbf{H} \in \mathbb{R}^{t\times d}$, is used to predict segmentation, $\mathbf{P}_b \in \mathbb{R}^{t}$, for the segmentation task and frame-wise representations, $\mathbf{P}_s \in \mathbb{R}^{t\times d}$, for the clustering task with two 1-D CNNs. We set $d = 80$ for the model. The details of the computation blocks are depicted in Fig. 5 and elaborated as follows.

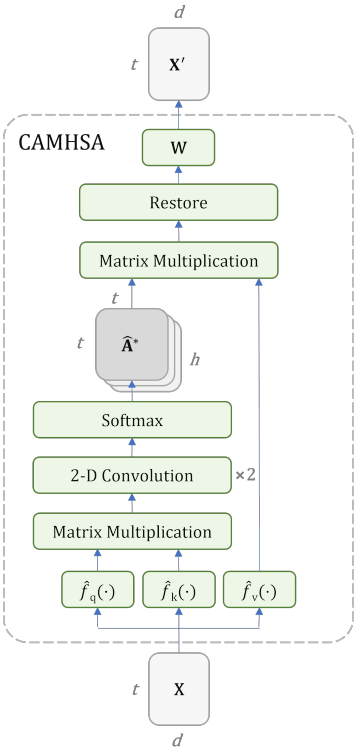


Fig. 2: Schematic diagram of the CAMHSA mechanism. The CAMHSA transforms $\mathbf{X}$ into $\mathbf{X}'$ similarly to the MHSA but with the addition of 2-D convolutions fusing information of the attention maps after the first matrix multiplication.

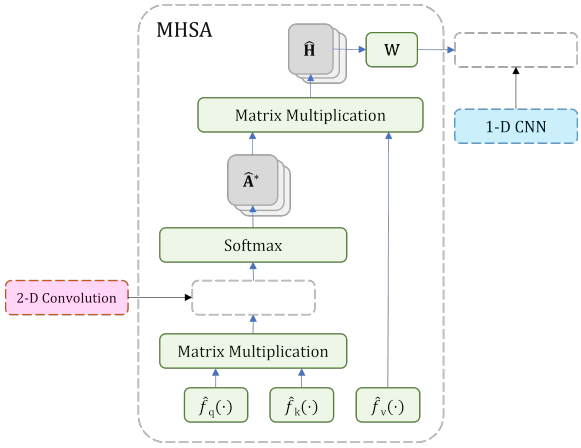


Fig. 3: Comparison between the CAMHSA and the Conformer. The CAMHSA improves the MHSA by performing 2-D convolutions after the matrix multiplication, whereas the Conformer utilizes a 1-D CNN for the MHSA output.

1) 2-D CNN (Fig. 5c): The 2-D CNNs are composed of three successive 2-D convolutions, each of which is followed by AdaNorm [38], the ReLU activation, and the *Squeeze-and-Excitation* (SE) operation [39], two branches of the dimension reduction performing *Max Pool* and *Dense* on the inner dimension, and a fusion of the two branches (*Dense & AdaNorm*

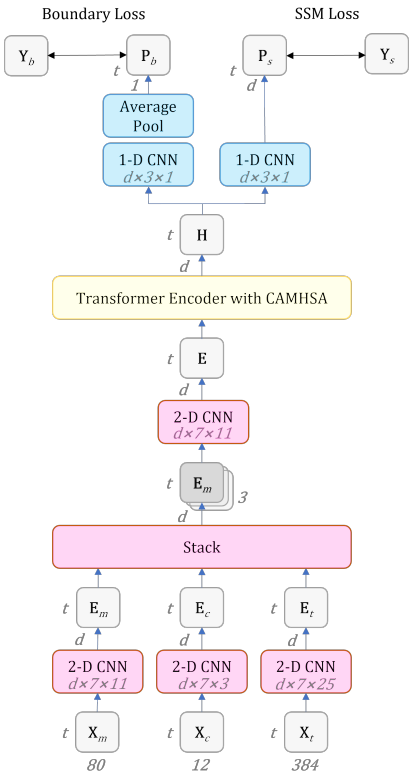
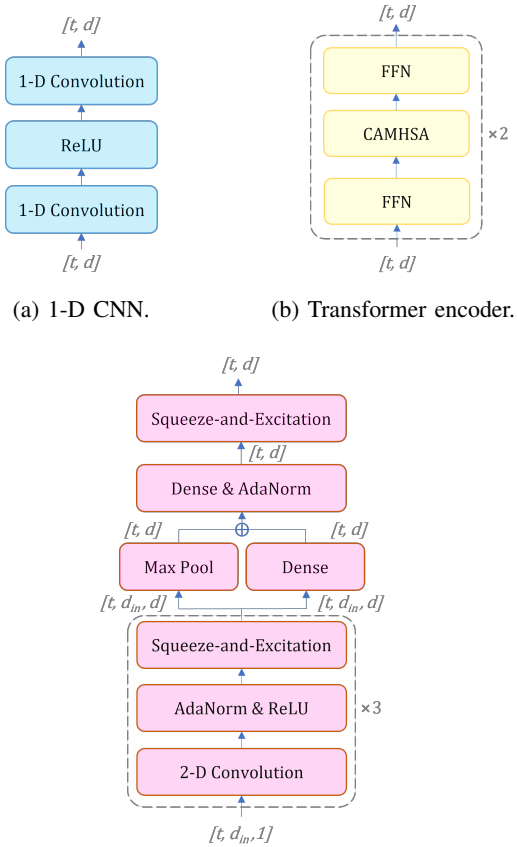


Fig. 4: Proposed model for MSA. The 2-D CNNs are used to encode and fuse the inputs; the Transformer encoder captures structural information via the CAMHSA mechanism; finally, the 1-D CNN functions as an output layer for boundary prediction. For each CNN, the number of convolution filters (c) and the kernel size, represented by height (h) and width (w), are denoted as $c \times h \times w$.



(a) 1-D CNN. (b) Transformer encoder.

(c) 2-D CNN.

Fig. 5: Computation blocks of the segmentation model. The MHSA block of the Transformer encoder is replaced by the CAMHSA. The dimensionality is indicated in the brackets.

followed by *SE*). The two branches of the dimension reduction function as global pooling layers summarizing the information of the *feature dimension* of the input (i.e., d_{in} in the figure).

2) Transformer Encoder with CAMHSA (Fig. 5b): We use a variant of the transformer architecture, i.e., the Macaron Net [37], [40], which has two feed-forward networks (FFNs) sandwiching the CAMHSA. Moreover, we employ relative positional embeddings to represent the distances between elements [41]. Besides, the residual connection and layer normalization techniques are also utilized in both the CAMHSA and the FFN (as used in the transformer blocks) despite not being displayed in the figure.

3) 1-D CNN (Fig. 5a): The 1-D CNNs consist of two 1-D convolutions with a ReLU activation in between.

D. Loss Functions

We employ two losses to jointly optimize the model (Fig. 4). The *boundary loss* is used for the segmentation task, while the *SSM loss* is used for the clustering task. The total loss is thus a summation of the two losses:

$$\text{Total Loss} = \text{Boundary Loss} + \text{SSM Loss}. \tag{5}$$

The boundary loss is a combination of the binary cross entropy loss (BCE) and the dice loss (DL) [42]:

$$\text{Boundary Loss} = \text{BCE}(\mathbf{P}_b, \mathbf{Y}_b) + \text{DL}(\mathbf{P}_b, \mathbf{Y}_b), \tag{6}$$

where $\text{DL}(\mathbf{A}, \mathbf{B}) = 1 - \frac{2 \times \sum \mathbf{A} \odot \mathbf{B}}{\sum \mathbf{A}^2 + \sum \mathbf{B}^2}$ ($\odot$ denotes the Hadamard product). We integrate the DL with the commonly used BCE, for it allows the model to pay attention to the overlapped regions between the predicted segmentation and the ground truth annotation [43], [44], and thus can alleviate the data imbalance issue in the segmentation task (i.e., the imbalance between boundary and non-boundary audio frames).

In contrast, we employ the SSM loss to regulate the predicted features, i.e., $\mathbf{P}_s$. Specifically, we compute two SSMs, $\mathbf{M}^{\mathbf{P}_s} \in \mathbb{R}^{t \times t}$ and $\mathbf{M}^{\mathbf{Y}_s} \in \mathbb{R}^{t \times t}$, with $\mathbf{P}_s$ and $\mathbf{Y}_s$ respectively, and then calculate the mean squared error:

$$\text{SSM Loss} = \frac{1}{t^2} \sum_{i,j=1:t} (\mathbf{M}^{\mathbf{P}_s} - \mathbf{M}^{\mathbf{Y}_s})^2, \tag{7}$$

where $\mathbf{M}_{ij}^{\mathbf{P}_s}$ is a similarity score ($\in [0, 2]$) measuring the normalized Euclidean distance between $\mathbf{P}_{s,i}$ and $\mathbf{P}_{s,j}$, and $\mathbf{M}_{ij}^{\mathbf{Y}_s} = 2$ if $\mathbf{Y}_{s,i} = \mathbf{Y}_{s,j}$ otherwise 0.

E. Clustering

We use the learned features by the proposed model, i.e., $\mathbf{P}_s$, to obtain section groups. We first perform the principal component analysis (PCA) to reduce the feature dimension of $\mathbf{P}_s$ from 80-D to 60-D. The reduced features are then clustered with Gaussian mixture models (GMMs). Specifically, we employ a set of GMMs with the number of mixture components ranging from 1 to k_{max} for each input piece, and then choose the best GMM based on the Bayesian information criterion (BIC) [45]. Finally, given the boundary prediction, we smooth the clustering results by reassigning the most common label in each segment to all frames within the segment.

IV. Evaluation

In this section, we present the comparative experiment conducted to evaluate the performance of the proposed method in terms of segmentation and clustering.

A. Data

To evaluate the proposed method, two datasets have been included. 1) The Beatles-TUT dataset[1] is a refined version of the Beatles dataset [17] by members of the Tampere University of Technology. It contains 174 songs by the Beatles and the corresponding *flat* (one-level, *upper*) annotations of structure. 2) The SALAMI dataset [16] is the largest publicly available set which contains *hierarchical* (two-level, *upper* and *lower*) annotations for 1,359 tracks. We use version 2.0 of the dataset[2] for the evaluation. We acquired 441 audio tracks from the Internet Archive[3] and found 555 matching tracks on YouTube[4].

We train and evaluate the proposed model with the two datasets individually. In the case of the Beatles-TUT dataset, we exclude 14 songs from the first album for testing and use the remaining 160 songs for training purposes. As for the SALAMI dataset, we utilize the 555 tracks from YouTube for training and the 441 tracks from the Internet Archive for testing. For both datasets, we augment the training data by 3 times via shifting the pitch of each track by ± 1 semitone.

B. Evaluation Metrics

For the segmentation task, we report the Hit Rate measure [46] with a time tolerance of ± 0.5 sec and ± 3 sec (which are the most frequently used values in the literature). This metric computes precision (P), recall (R), and F1 score (F1) by checking if a predicted boundary is close enough to an annotated boundary according to the given time tolerance.

For the clustering task, two metrics are employed. The *pairwise agreement* [47] is used to evaluate the frame-clustering segmentation regarding a flat annotation, while the *L-measure* [48] assesses all segmentation levels as a whole concerning a hierarchical annotation. Similarly to the Hit Rate measure, both the pairwise agreement and the L-measure compute the P, R, and F1 scores. Note that the L-measure is applied to the SALAMI dataset only as there are no hierarchical annotations in the Beatles-TUT dataset. All the evaluations are done using the *mir_eval* package [49].

C. Experiment Settings

The proposed model is evaluated separately on the segmentation task and the clustering task. Since the SALAMI dataset contains annotations at two structure levels, we equip the proposed model with an additional branch when training with the SALAMI dataset. Concretely, we create a branch by duplicating all computation blocks following the stack operation (see Fig. 4), while the first three 2-D CNNs are shared by the branches.

To validate our approach regarding the CAMHSA mechanism, we build a baseline model by removing the transformer encoder from the proposed model and including as additional inputs the SSMs derived from $\{\mathbf{X}_m, \mathbf{X}_c, \mathbf{X}_t\}$. Since it is memory-consuming to apply deep learning models to multiple track-level SSMs, we alternatively build three baseline models (denoted as *baseline-*$\{m, c, t\}$), each of which uses one of $\{\mathbf{X}_m, \mathbf{X}_c, \mathbf{X}_t\}$ and its corresponding SSM as inputs.

In comparison to previous work using supervised learning approaches, we report the best results in [9] (segmentation on SALAMI) and [11] (segmentation on Beatles and the upper level of SALAMI). To our knowledge, the former achieves the best segmentation performance on the SALAMI dataset while the latter on the Beatles dataset. We also report the results by [19] (segmentation and clustering on both Beatles and SALAMI) in comparison with the unsupervised learning approach. Note that the comparisons to these works should be taken with a grain of salt for there are many differences between these works and ours, such as dataset version (e.g., Beatles-Isophonics and Beatles-TUT), data usage (e.g., training and testing sets), and data representation.

Finally, we conduct ablation studies on the CAMHSA mechanism and the loss functions. Specifically, we build a model (denoted as *proposed w/o conv*) by replacing the CAMHSA blocks of the proposed model with the MHSA to validate the incorporation of convolutions; we also train the proposed model without including the DL in the boundary loss (denoted as *proposed w/o dl*) to investigate whether the DL provides any benefits to the segmentation task.

D. Results and Discussions

1) The Segmentation Task (Table I): On the Beatles-TUT dataset (Table Ia), our approach consistently outperformed all the baselines. Likewise, on the SALAMI dataset (Table Ib), our method was superior in all cases except for *Hit Rate(3)* at the lower level. When compared with the previous works, our method achieved new state-of-the-art segmentation performances on both datasets. More strikingly, our approach can rival the unsupervised approach [19] where the amount of audio tracks for training is two orders of magnitude larger than ours. Based on these findings, it appears that acquiring knowledge of multifaceted self-similarity is a more effective approach than relying on pre-established affinity matrices.

[1] https://pythonhosted.org/msaf/datasets.html.
[2] https://github.com/DDMAL/salami-data-public.
[3] https://archive.org/.
[4] https://github.com/jblsmith/matching-salami.

TABLE I: Performance on the segmentation task. *Hit Rate (0.5)* and *Hit Rate (3)* denote using the metric with a time threshold of {0.5, 3} sec. *baseline-{m, c, t}* indicates the baseline model with {mel-spectrogram, chromagram, tempogram} and the corresponding SSM as inputs. *proposed w/o dl* is the proposed model without using the dice loss for training. '–' is marked if a measure is not provided in the previous work.

Level	Model	Hit Rate (0.5)			Hit Rate (3)		
		P	R	F1	P	R	F1
upper	proposed	**0.555**	**0.617**	**0.578**	0.675	0.757	0.705
	proposed w/o conv	0.536	0.582	0.553	**0.700**	**0.769**	**0.729**
	proposed w/o dl	0.546	0.606	0.573	0.689	0.762	0.722
	baseline-m	0.436	0.560	0.488	0.559	0.718	0.625
	baseline-c	0.424	0.466	0.441	0.640	0.720	0.672
	baseline-t	0.425	0.511	0.460	0.608	0.733	0.659
	Maezawa [11]	0.507	0.520	0.492	–	–	–
	Buisson et al. [19]	–	–	–	–	–	0.718

(a) Evaluation on the Beatles-TUT dataset.

Level	Model	Hit Rate (0.5)			Hit Rate (3)		
		P	R	F1	P	R	F1
upper	proposed	**0.577**	0.586	0.559	**0.680**	0.688	**0.658**
	proposed w/o conv	0.534	**0.631**	**0.560**	0.629	**0.737**	0.658
	proposed w/o dl	0.533	0.537	0.518	0.665	0.666	0.645
	baseline-m	0.509	0.549	0.512	0.635	0.684	0.638
	baseline-c	0.504	0.375	0.417	0.636	0.472	0.525
	baseline-t	0.423	0.339	0.363	0.613	0.490	0.526
	Grill & Schlüter [9]	–	–	0.508	–	–	–
	Maezawa [11]	0.301	0.347	0.306	–	–	–
	Buisson et al. [19]	–	–	–	–	–	0.627
lower	proposed	**0.650**	**0.657**	**0.621**	0.806	0.820	**0.774**
	proposed w/o conv	0.590	0.611	0.574	0.785	0.821	0.768
	proposed w/o dl	0.582	0.598	0.561	0.790	0.812	0.763
	baseline-m	0.553	0.553	0.528	0.801	0.810	0.771
	baseline-c	0.483	0.581	0.502	0.728	**0.886**	0.763
	baseline-t	0.405	0.349	0.357	**0.827**	0.742	0.746
	Grill & Schlüter [9]	–	–	0.485	–	–	–
	Buisson et al. [19]	–	–	–	–	–	0.643

(b) Evaluation on the SALAMI dataset.

2) The Clustering Task (Table II): Similarly to the segmentation task, our approach surpassed the baselines in almost all cases. In general, the clustering performance on the SALAMI dataset (Table IIb) is worse than that on the Beatles-TUT (Table IIa) since the SALAMI dataset is more diverse in terms of music genres and styles. Upon evaluation, it is apparent that clustering at the lower level is more challenging than at the upper level for the SALAMI dataset. This can be attributed to the greater number of boundaries in the lower level compared to the upper level. Furthermore, the pairwise agreement measure is sensitive to the precise placement of the boundaries between the prediction and the ground truth [50]. Besides, it is worth pointing out that [19] obtained remarkable performances on the clustering task due to its contrastive learning objective aligning with the evaluation metrics.

3) Ablation studies (proposed w/o conv and w/o dl): When the CAMHSA is replaced by the MHSA (*proposed w/o conv*), we noticed a decline in both segmentation and clustering performances. This suggests that the correlations among the attention maps of different heads could potentially play a role in MSA. As for the ablation on the loss functions (*proposed w/o dl*), we found that employing the DL positively impacted the overall performance.

TABLE II: Performance on the clustering task. Note that the L-measure can be applied to the SALAMI dataset only, and it computes a unified score over the two levels although we report the results at the *upper* level.

Level	Model	Pairwise Agreement			L-measure		
		P	R	F1	P	R	F1
upper	proposed	**0.553**	0.803	0.648	–	–	–
	proposed w/o conv	0.536	**0.828**	0.639	–	–	–
	proposed w/o dl	0.512	0.753	0.604	–	–	–
	baseline-m	0.509	0.659	0.565	–	–	–
	baseline-c	0.542	0.752	0.624	–	–	–
	baseline-t	0.553	0.575	0.554	–	–	–
	Buisson et al. [19]	–	–	**0.723**	–	–	–

(a) Evaluation on the Beatles-TUT dataset.

Level	Model	Pairwise Agreement			L-measure		
		P	R	F1	P	R	F1
upper	proposed	0.561	0.695	0.592	0.420	0.559	0.476
	proposed w/o conv	0.550	**0.700**	0.589	0.412	0.552	0.469
	proposed w/o dl	0.581	0.666	0.594	**0.442**	0.570	0.494
	baseline-m	0.590	0.589	0.559	0.417	0.506	0.454
	baseline-c	0.538	0.690	0.577	0.406	0.532	0.457
	baseline-t	**0.596**	0.477	0.501	0.405	0.450	0.423
	Buisson et al. [19]	–	–	**0.714**	0.432	**0.694**	**0.527**
lower	proposed	0.428	0.550	0.457	–	–	–
	proposed w/o conv	0.416	**0.554**	0.450	–	–	–
	proposed w/o dl	**0.490**	0.454	0.445	–	–	–
	baseline-m	0.438	0.408	0.397	–	–	–
	baseline-c	0.415	0.503	0.430	–	–	–
	baseline-t	0.444	0.289	0.328	–	–	–
	Buisson et al. [19]	–	–	**0.580**	–	–	–

(b) Evaluation on the SALAMI dataset.

4) Feature Selection Issue (baseline-{m, c, t}): When comparing the baselines, it becomes clear that there is no prevailing feature for MSA. The mel-spectrogram exhibited superiority in the segmentation task, while the chromagram appeared to be preferable to the clustering task. Besides, the tempogram can also provide structural information to some extent. These results highlight the importance of learning higher-level representations and affinity matrices instead of relying on manually crafted features.

5) Visualization of Attention Maps: To examine the knowledge gained by CAMHSA on MSA, we analyzed the attention maps of the testing data from the SALAMI dataset. There were three primary observations exemplified in Fig. 6: 1) the attention heads display a variety of block patterns; 2) the size of the block patterns tends to be larger at the upper level compared to the lower level; 3) the SSM computed from the learned representation ($\mathbf{M}^{\mathbf{P}_s}$) is similar to the annotated one ($\mathbf{M}^{\mathbf{Y}_s}$). The first observation is in line with previous works suggesting that the attention heads can capture diverse relationships. The second observation reflects the fact that the musical segments tend to have longer duration at the upper

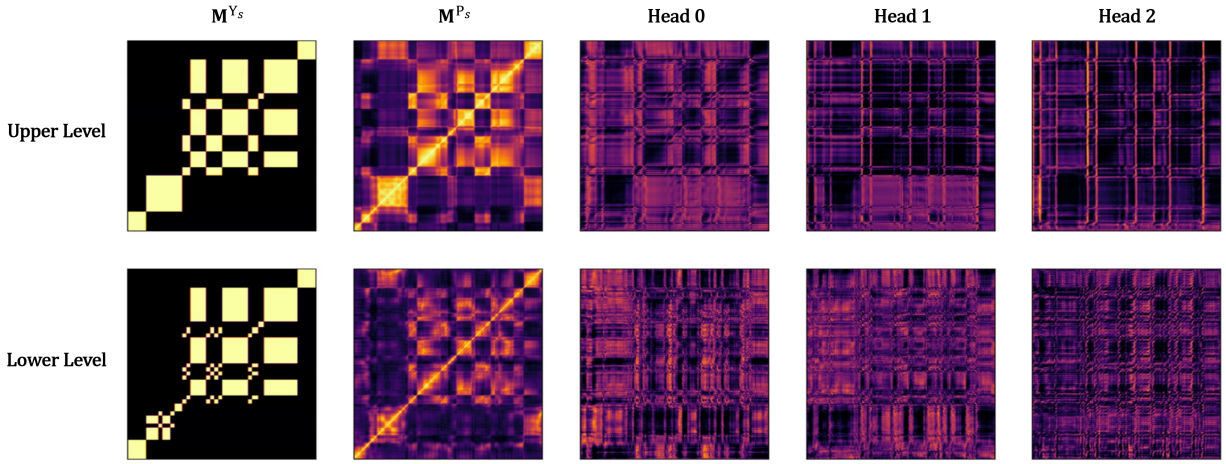


Fig. 6: Visualization of the attention maps for the song "Shim Sham" by Caravan of Thieves (song id 1038 of the SALAMI dataset). $\mathbf{M}^{\{\mathbf{Y}_s,\ \mathbf{P}_s\}}$ is the SSM computed with $\{\mathbf{Y}_s,\ \mathbf{P}_s\}$, respectively. Only the first three heads (index $\{0,\ 1,\ 2\}$) at the second layer of the transformer encoder are shown.

level. The third observation shows that the CAMHSA can effectively combine attention maps from different sources to create a cohesive representation ($\mathbf{P}_s$) that closely resembles the ground truth ($\mathbf{Y}_s$) in terms of SMMs.

V. Conclusion

This research focused on utilizing the multi-head self-attention (MHSA) mechanism to learn a multifaceted view of self-similarity across multiple abstraction levels. To achieve this, we have developed the convolution-augmented MHSA (CAMHSA) mechanism, which fuses various perspectives of self-similarity to extract latent features for audio-based MSA. Through our experiments, we have clearly demonstrated the effectiveness of the CAMHSA by comparing it with other methods that rely on hand-crafted affinity matrices. Especially for the segmentation task, we obtained new state-of-the-art results on the Beatles-TUT and the SALAMI datasets. For the clustering task, we adopted a two-stage framework, where the learning of audio representations is detached from the successive clustering stage. The clustering performance would thus heavily depend on the capability of the employed clustering algorithm. To mitigate this potential incompatibility, we plan to involve a deep clustering method in the current framework. Finally, by fusing different acoustic features, we are able to find a unified structure for each input signal. Examining the differences between the attention maps and SSMs derived from these features can provide musical insights into the relationship between the unified structure and each specific feature.

Acknowledgment

This work was partially supported by JST PRESTO No. JPMJPR20CB and KAKENHI Nos. 20H00602 and 21H03572.

References

[1] H. Grohganz, M. Clausen, N. Jiang, and M. Müller, "Converting path structures into block structures using eigenvalue decompositions of self-similarity matrices," in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 209–214, 2013.

[2] B. McFee and D. P. W. Ellis, "Learning to segment songs with ordinal linear discriminant analysis," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5197–5201, 2014.

[3] J. B. L. Smith and M. Goto, "Multi-part pattern analysis: Combining structure analysis and source separation to discover intra-part repeated sequences," in *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 716–723, 2017.

[4] T. Grill and J. Schlüter, "Music boundary detection using neural networks on spectrograms and self-similarity lag matrices," in *Proceedings of the 23rd European Signal Processing Conference (EUSIPCO)*, pp. 1296–1300, 2015.

[5] T. Cheng, J. B. L. Smith, and M. Goto, "Music structure boundary detection and labelling by a deconvolution of path-enhanced self-similarity matrix," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 106–110, 2018.

[6] C. J. Tralie and B. McFee, "Enhanced hierarchical music structure annotations via feature level similarity fusion," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 201–205, 2019.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017.

[8] K. Ullrich, J. Schlüter, and T. Grill, "Boundary detection in music structure analysis using convolutional neural networks," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 417–422, 2014.

[9] T. Grill and J. Schlüter, "Music boundary detection using neural networks on combined features and two-level annotations," in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 531–537, 2015.

[10] A. Cohen-Hadria and G. Peeters, "Music structure boundaries estimation using multiple self-similarity matrices as input depth of convolutional neural networks," in *Proceedings of the AES International Conference Semantic Audio*, 2017.

[11] A. Maezawa, "Music boundary detection based on a hybrid deep model of novelty, homogeneity, repetition and duration," in *Proceedings of*

the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 206–210, 2019.

[12] C. Hernandez-Olivan, J. R. Beltrán, and D. Diaz-Guerra, "Music boundary detection using convolutional neural networks: A comparative analysis of combined input features," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 7, no. 2, p. 78, 2021.

[13] M. Goto, "A chorus-section detecting method for musical audio signals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 437–440, 2003.

[14] B. McFee and D. Ellis, "Analyzing song structure with spectral clustering," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 405–410, 2014.

[15] G. Shibata, R. Nishikimi, and K. Yoshii, "Music structure analysis based on an LSTM-HSMM hybrid model," in *Proceedings of the 21th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 23–29, 2020.

[16] J. B. L. Smith, J. A. Burgoyne, I. Fujinaga, D. D. Roure, and J. S. Downie, "Design and creation of a large-scale database of structural annotations," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 555–560, 2011.

[17] M. Mauch, C. Cannam, M. Davies, S. Dixon, C. Harte, S. Kolozali, D. Tidhar, and M. Sandler, "OMRAS2 metadata project 2009," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR) - Late-Breaking Session*, 2009.

[18] M. C. McCallum, "Unsupervised learning of deep features for music segmentation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 346–350, IEEE, 2019.

[19] M. Buisson, B. McFee, S. Essid, and H. C. Crayencour, "Learning multi-level representations for hierarchical music structure analysis," in *Proceedings of the 23rd International Society for Music Information Retrieval Conference (ISMIR)*, pp. 591–597, 2022.

[20] O. Nieto, G. J. Mysore, C. Wang, J. B. L. Smith, J. Schlüter, T. Grill, and B. McFee, "Audio-based music structure analysis: Current trends, open challenges, and applications," *Transactions of the International Society for Music Information Retrieval (TISMIR)*, vol. 3, no. 1, pp. 246–263, 2020.

[21] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 551–561, 2016.

[22] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A decomposable attention model for natural language inference," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2249–2255, 2016.

[23] P. Cao, Y. Chen, K. Liu, J. Zhao, and S. Liu, "Adversarial transfer learning for chinese named entity recognition with self-attention mechanism," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 182–192, 2018.

[24] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (NAACL-HLT)*, pp. 4171–4186, 2019.

[25] R. Al-Rfou, D. Choe, N. Constant, M. Guo, and L. Jones, "Character-level language modeling with deeper self-attention," in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pp. 3159–3166, 2019.

[26] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7794–7803, 2018.

[27] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, A. Ku, and D. Tran, "Image transformer," in *Proceedings of the 35th International Conference on Machine Learning (ICML)*, vol. 80, pp. 4052–4061, 2018.

[28] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, pp. 7354–7363, 2019.

[29] J. Park, K. Choi, S. Jeon, D. Kim, and J. Park, "A bi-directional Transformer for musical chord recognition," in *Proceedings of the 20th International Society for Music Information Retrieval Conference, (ISMIR)*, pp. 620–627, 2019.

[30] T. Chen and L. Su, "Harmony Transformer: Incorporating chord segmentation into harmony recognition," in *Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 259–267, 2019.

[31] Y. Huang and Y. Yang, "Pop music Transformer: Beat-based modeling and generation of expressive pop piano compositions," in *Proceedings of the 28th ACM International Conference on Multimedia (ACM Multimedia)*, pp. 1180–1188, 2020.

[32] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, "Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned," in *Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL)*, pp. 5797–5808, 2019.

[33] J. Cordonnier, A. Loukas, and M. Jaggi, "On the relationship between self-attention and convolutional layers," in *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

[34] P. Grosche, M. Müller, and F. Kurth, "Cyclic tempogram - A mid-level tempo representation for musicsignals," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5522–5525, 2010.

[35] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in Python," in *Proceedings of the 14th python in science conference (SCIPY)*, vol. 8, 2015.

[36] L. J. Ba, R. Kiros, and G. E. Hinton, "Layer normalization," in *arXiv preprint arXiv: 1607.06450*, 2016.

[37] A. Gulati, J. Qin, C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for speech recognition," in *Proceedings of the 21st Annual Conference of the International Speech Communication Association (Interspeech)*, pp. 5036–5040, 2020.

[38] J. Xu, X. Sun, Z. Zhang, G. Zhao, and J. Lin, "Understanding and improving layer normalization," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 4383–4393, 2019.

[39] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7132–7141, 2018.

[40] Y. Lu, Z. Li, D. He, Z. Sun, B. Dong, T. Qin, L. Wang, and T. Liu, "Understanding and improving Transformer from a multi-particle dynamic system point of view," *CoRR abs/1906.02762*, 2019.

[41] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (NAACL-HLT)*, pp. 464–468, 2018.

[42] F. Milletari, N. Navab, and S. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proceedings of the 4th International Conference on 3D Vision (3DV)*, pp. 565–571, 2016.

[43] S. Jadon, "A survey of loss functions for semantic segmentation," in *Proceedings of the IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–7, 2020.

[44] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, and J. Li, "Dice loss for data-imbalanced NLP tasks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 465–476, 2020.

[45] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461 – 464, 1978.

[46] D. Turnbull, G. R. G. Lanckriet, E. Pampalk, and M. Goto, "A supervised approach for detecting boundaries in music using difference features and boosting," in *Proceedings of the 8th International Conference on Music Information (ISMIR)*, 2007.

[47] M. Levy and M. B. Sandler, "Structural segmentation of musical audio by constrained clustering," *IEEE Transactions on Speech and Audio Processing*, vol. 16, no. 2, pp. 318–326, 2008.

[48] B. McFee, O. Nieto, M. M. Farbood, and J. P. Bello, "Evaluating hierarchical structure in music annotations," *Frontiers in Psychology*, vol. 8, 2017.

[49] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis, "mir_eval: A transparent implementation of common MIR metrics," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 367–372, 2014.

[50] O. Nieto and J. P. Bello, "Music segment similarity using 2d-fourier magnitude coefficients," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 664–668, 2014.