# A Comparative Study of Statistical Features and Deep Learning for Orchestral Texture Classification

Zih-Syuan Lin*, Jun-You Wang†, and Li Su*
* Institute of Information Science, Academia Sinica, Taiwan
† Department of Computer Science and Information Engineering, National Taiwan Normal University, Taiwan
E-mail: clarelyn@iis.sinica.edu.tw, jywang@csie.ntnu.edu.tw, lisu@iis.sinica.edu.tw

***Abstract*—Orchestral texture, which reflects the interaction of instruments in ensemble music, is central to understanding multi-track music; however, symbolic texture classification has not been widely studied. This study addresses the task of assigning textural roles to individual track-bar units (i.e., each bar of each track) in symphonic movements. We evaluate a range of features and models, including random forest, convolutional neural networks, hybrid approaches, and pre-trained models, on the Orchestration and S3 datasets. The results show that duration-related statistical features are the most informative, while identifying melodic roles remains challenging. Overall, our findings suggest the potential of symbolic texture classification, and highlight key challenges in melody extraction for multi-track music and ensuring labeling consistency across datasets.**

## I. Introduction

Orchestral music represents one of the largest and most complex forms of ensemble in Western music, characterized by its large scale, rich instrumentation, and diverse textural changes. Among these, texture plays a key analytical role in Classical and Romantic repertoire [1]. As defined by Giraud *et al.* [2], "Texture refers to the sound aspects of a musical structure." It generally describes the interaction among voices and their roles, typically melody and accompaniment, and how individual layers collectively form texture such as monophony, polyphony, homophony, and beyond.

Symbolic (orchestral) texture classification, a task that explores textural role changes across an entire piece for each track or voice, remains relatively rare and challenging in the field of Music Information Retrieval (MIR). Most symbolic music recognition tasks in MIR have focused on identifying boundaries or change points along the temporal axis, often treating all tracks as a unified whole, including musical structure analysis [3], [4] and functional harmony recognition [5], [6]. Other tasks, for example, motif discovery [7], instead emphasize note-level patterns, regardless of the tracks from which these notes originate. In contrast, analyzing textural roles, such as melody or harmonic/rhythmic accompaniment, provides a different perspective that emphasizes functional relationships between tracks. Although monophonic melody identification has been investigated in the context of pop and piano music [8], [9], few studies have directly addressed the concept of texture as it applies to all tracks [10], [11].

One of the major challenges in computational research on texture in ensemble music is the limited availability of annotated datasets. Although many multi-track symbolic music datasets exist, few provide texture annotations. This scarcity is largely due to the laborious and conceptually challenging nature of annotating textural roles, as such roles are context-dependent and require perceptual interpretation of the richness and complexity of possible instrument combinations in ensemble settings [12]. Giraud *et al.* [2] note that the roles of melody and accompaniment can also be perceived in monophonic or single instrument contexts, illustrating this ambiguity. Fig. 1 shows an example annotation from Dvořák No. 9 (I). In addition, there is still a lack of systematic comparisons across models, features, and cross-dataset generalization.

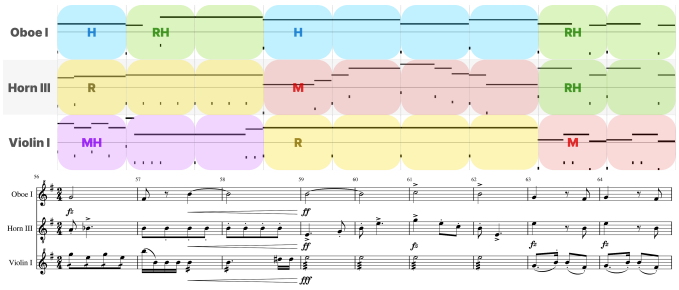


Fig. 1. A texture annotation excerpt from Dvořák No. 9 (I) in the S3 dataset, featuring the theme played by the third horn. Each instrument is shown with a piano roll (top) and a replay matrix (bottom), where the replay matrix marks only the onset times of each note [15], with the corresponding score below.

To address these gaps, this study aims to model the symbolic orchestral texture classification task, which assigns a textural label to each track at the bar level, and to analyze the impact of symbolic and statistical features, all within the context of orchestral classical music, where melodic lines are often less straightforward. Meanwhile, inspired by the success of large language models in natural language processing, recent work has begun applying similar frameworks to symbolic music by treating it as structured language. For instance, BERT-based models [13], [14] have been trained to learn musical syntax and semantics, achieving convincing results in melody extraction for pop music. These promising outcomes raise the question of whether such models can generalize to other musical attributes, particularly orchestral texture.

The contributions of this paper are summarized below. First, we frame symbolic orchestral texture classification as a systematic research task and provide the benchmark for two annotated orchestral datasets. Second, our comparative analysis

979-8-3315-7206-8/25/$31.00 

shows that note duration is the most informative symbolic cue, with random forest (RF) leveraging it most consistently, while the high variance of hybrid and pre-trained models suggests both their potential and their sensitivity to limited training data. Third, our experiments reveal the inherent challenges in texture classification, particularly the difficulty of identifying "melody" roles in complex musical contexts.

## II. Related Work

The term used to describe texture and its related concepts varies in the literature. Commonly used terms also include "layers," "streams," as well as functional descriptions such as "melody" or "rhythm." The survey is based on these terms.

### A. Texture in Music Analysis

Texture is mentioned sporadically in the literature of computational music analysis. Huron discussed textural changes relying on the concept of synchrony, which is an approach based on texture that measures and the degrees of sonic simultaneity [16]. Duane discussed linking note synchrony with human perception of auditory streams in string quartets [17], and then adjudicated between two competing interpretations of Schubert's three-key expositions—the six-zone structure and tonal-functional nonalignment—by applying statistical models of texture, using linear discriminant analysis to approximate thematic function [18]. Later, De Souza *et al.* [19] used the corpus proposed by [17] to investigate the relationship between onset synchrony and section changes in sonata form. Moreover, texture also has been used as an entry point for analyzing modern music, such as EDM [20] and rock music [21], as modern music usually emphasizes the organization, tension, and development of material through timbre and texture [22].

### B. Symbolic datasets for texture classification

Although several symbolic datasets have attempted to provide texture annotations across different musical settings, from solo piano to full orchestral works, their number remains limited. Couturier *et al.* [23] focus on note, chord, and voice organization in classical piano scores. For ensemble music, the string quartet dataset by Giraud *et al.* [2] includes two roles: melody (typically monophonic) and accompaniment (providing rhythmic and harmonic support). Le *et al.* [12] offer a finer classification for orchestral music, labeling each bar of each track as melody, rhythmic accompaniment, harmonic accompaniment, or mixed. Building on this, Lin *et al.* [24] proposed the S3 dataset, including annotations of textural roles, musical form, and functional harmony in symphonic works.

### C. Texture-related Recognition and Generation Tasks

Recent work on multi-track symbolic music has investigated a range of strategies for texture-related analysis. In the absence of textural labels, stream segmentation methods [25] resemble voice separation, but with a different focus: rather than separating individual voices, they aim to segment musical streams into smaller groups that reflect functional consistency, offering a different viewpoint on how texture can be represented. Hsiao *et al.* [8] employed a convolutional neural network (CNN)-based model to group notes into streams, selecting the stream with the highest average pitch as melody.

When textural labels are available, supervised texture classification is performed on temporal segments of individual instruments, using models including RFs or CNNs. Giraud *et al.* [2] applied dynamic programming to segment musical layers based on rhythmic similarity. Soum-Fontez *et al.* [10] adopted statistical features from [26] and trained RF classifiers to predict textural roles. Chu *et al.* [11] explored CNNs for orchestral texture classification, examining how input content (e.g., window sizes) and the number of input tracks influence model performance.

On the generation side, Wang *et al.* [27] pre-defined note patterns for different textures to generate music with more regular rhythm and varying accompaniment density. Maccarini *et al.* [28] leveraged textural labels to explore a human-machine co-creative scenario for composition. Le and Yang [29] proposed METEOR for orchestration, adjusting note patterns for homophonic music using statistical features.

TABLE I
Mapping of textural role to class IDs used for evaluation

| Textural role (`abbreviation`) | Class ID | Multi-label `[M, R, H]` | Portion in dataset | |
|---|---|---|---|---|
| | | | Orch | S3 |
| None (`N` or `None`) | 0 | `[0, 0, 0]` | 39.14% | 41.86% |
| Melody (`M`) | 1 | `[1, 0, 0]` | 15.17% | 14.58% |
| Rhythm (`R`) | 2 | `[0, 1, 0]` | 20.24% | 16.88% |
| Melody+Rhythm (`MR`) | 3 | `[1, 1, 0]` | 0.63% | 5.81% |
| Harmony (`H`) | 4 | `[0, 0, 1]` | 14.90% | 10.32% |
| Melody+Harmony (`MH`) | 5 | `[1, 0, 1]` | 0.00% | 2.34% |
| Rhythm+Harmony (`RH`) | 6 | `[0, 1, 1]` | 9.91% | 7.98% |
| All (`All`) | 7 | `[1, 1, 1]` | 0.00% | 0.23% |

## III. Methodology

We follow [11] to formulate orchestral texture classification as a multi-label classification task based on the textural role definitions provided in [12]. Each input unit, referred to as a *track-bar*, corresponds to one bar from a track within a symphonic movement. Each track-bar is annotated with a combination of three textural role classes: melody, rhythm, and harmony. For evaluation, we treat each label combination as a distinct class, mapping them to eight mutually exclusive class IDs, as shown in Table I.

The Orchestration dataset [12] and the S3 dataset [24] are used. The former consists of 18 first movements in sonata form, selected from symphonies by Mozart (Nos. 32–41), Haydn (Hob. 99–104), and Beethoven (Nos. 1–9), following the subset used in [11] due to missing MusicXML files in the original corpus. The latter comprises 16 movements from four complete symphonies, namely Mozart No. 41, Beethoven No. 9, Dvořák No. 9, and Tchaikovsky No. 6, covering diverse formal types such as sonata, ternary, rondo, and theme and variations. In total, the Orchestration and S3 datasets contain 91,053 and 172,171 track-bar units, respectively, of which

63,659 and 94,773 contain at least one note. The distribution of texture labels across the two datasets is summarized in Table I.

Our experiments include four types of approaches: a rule-based method (skyline), simple deep learning models, pre-trained models, and tree-based or hybrid methods using symbolic features.

**Rule-based algorithm (skyline).** This method is adopted from [10], assuming the melody corresponds to the highest pitch line. In our adaptation, each track-bar is labeled as "Melody" if more than 50% of its notes in a given bar hold the highest pitch among all sounding notes within that bar; otherwise, it is labeled as "Rhythm," representing the dominant class after merging the other seven roles. Harmony and others are not considered in this heuristic.

**Deep learning (DL) models.** First, we evaluate four common neural architectures, each of which consists of a feature encoder (CNN layers, BiLSTM layers, Transformer blocks, or a CNN–BiLSTM stack for the CRNN) followed by linear layers for classification. The input to all models is a piano-roll segment of $k$ tracks (i.e., the target plus four randomly sampled tracks) over three consecutive bars (i.e., the target bar and its adjacent bars), resulting in an input size of $(k, 288, 128)$, where $k$ is set at 5, $288 = 96$ (time steps per bar) $\times$ 3 (bars) is the temporal duration, and 128 denotes the MIDI pitch dimension. The output is a three-dimensional multi-label prediction corresponding to the texture roles. Training schedules were determined empirically from preliminary convergence tests: 150 epochs for CNN, 500 for BiLSTM, and 200 for both CRNN and Transformer, with the best checkpoints selected by validation accuracy. Next, we evaluate two BERT-based models, MidiBERT and M2BERT, both fine-tuned for the orchestral texture classification task. MidiBERT [14] includes two variants: MidiBERT-f is pre-trained on piano datasets and kept frozen during training, with only a linear layer trained for texture classification. In contrast, MidiBERT-n is fully fine-tuned for the task. M2BERT [30] adopts a different architecture while using the same pre-training data as MidiBERT-n, and is also trained with all parameters updated. Following the original fine-tuning procedure, MidiBERT-n and M2BERT are trained for 10 epochs. By contrast, we train MidiBERT-f for 100 epochs, as we empirically found that convergence is slower when only the final linear layer is updated.

**Symbolic-feature-based models.** This part includes a RF with handcrafted features, CNN with added symbolic cues, and a CNN–RF hybrid using symbolic features. To begin with, the RF classifier was first employed in [10] for string quartets, using statistical features extracted at the *track-bar* level as input. These include: *onset synchrony* (the ratio between the number of new notes and the total number of notes within a bar); *polyphony rate* (the proportion of ticks within a track where two or more notes start simultaneously, relative to the total duration of the track in ticks); *syncopation rate* (the number of notes whose onsets do not fall on beat positions, normalized by the total number of notes in the movement); *occupation rate* (the ratio of the total duration of all notes in a given track within a bar to the total duration of all notes across tracks in that bar); maximum, minimum, mean, and standard deviation for durations, pitches, and pitch intervals, as well as the total duration and note count. All features are normalized to account for differences in bar length and pitch range. The same statistical features are also computed at *all-bar* level (i.e., over notes within each bar regardless of track). Instrument names are encoded using one-hot vectors with 31 categories. We then augment CNN models with additional symbolic cues by incorporating three features into the piano roll input: a binary matrix indicating bar lines, replay matrices, and instrument-name scalars, where the latter two are computed for each channel. Instrument names are encoded as scalar values in $[0, 1]$ by normalizing the instrument index over the total number of categories and broadcasting to match the input dimensions. All features are stacked with the original piano roll, resulting in an input shape of $(k \times 3 + 1, 288, 128)$. Lastly, for a hybrid design, we extract intermediate CNN embeddings (before the final linear layers), flatten them into a one-dimensional vector, and concatenate them with the symbolic features described above as input to an RF classifier.

For evaluation, model performance is assessed using accuracy and weighted macro-averaged precision, recall, and F1-score. Weighted macro-averaging is employed to address the substantial class imbalance, as some categories (e.g., `All` and `MH`) occur very infrequently. The `None` label is excluded from evaluation, as it typically corresponds to bars with little or no sound, accounting for 39.14% (Orchestration) and 41.86% (S3) of the track-bar units. Including this category may inflate evaluation scores by encouraging models to overpredict `None`.

We adopt two evaluation protocols. The first focuses on the Orchestration dataset with fixed test set (1st movements form Mozart No.38, Haydn Hob.99, and Beethoven No.1), and include two experiments: (i) a comparison across deep learning models, and (ii) evaluations of symbolic-feature-based models. Each trained model is also evaluated on the S3 dataset. The second setting applies k-fold cross-validation (CV) to assess robustness and to examine whether models trained on one dataset can generalize to another. For the Orchestration dataset, which contains all movements from different pieces, we adopt 18-fold CV (often referred to as leave-one-piece-out), where each fold is tested not only on the held-out movement but also on the entire S3 dataset. For the S3 dataset, we use 4-fold CV, since it consists of four symphonies; here each fold holds out one symphony to avoid data leakage, and the trained model is evaluated on both the held-out symphony and the full Orchestration dataset. This protocol allows us to assess whether textural label classification is a feasible task and how well models transfer across datasets.

## IV. Discussion

We summarize and discuss the results from three perspectives: overall model performance, to compare supervised and hybrid approaches; feature importance, to identify key statistical cues; and annotation consistency, to assess the impact of labeling differences on model generalization.

TABLE II
MODEL COMPARISON ON THE ORCHESTRATION AND S3 DATASETS. SCORES ARE REPORTED AS WEIGHTED MACRO PRECISION, RECALL, AND F1 (ABBREVIATED AS ACC., PREC., REC., AND F1).

| Model | Input / Features | Orchestration dataset | | | | S3 dataset | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Acc. | Prec. | Rec. | F1 | Acc. | Prec. | Rec. | F1 |
| Skyline | - | .357 | .207 | .357 | .223 | .202 | .187 | .306 | .195 |
| *Fix Test Set: (i) comparing simple deep learning models and pre-trained models* | | | | | | | | | |
| CNN | Piano roll input | .638 | **.675** | .644 | .652 | .289 | **.315** | .296 | **.305** |
| BiLSTM | Piano roll input | .649 | .669 | .640 | **.654** | .198 | .197 | .166 | .181 |
| CRNN | Piano roll input | .635 | .669 | .624 | .643 | **.321** | .312 | **.321** | .278 |
| Transformer | Piano roll input | .600 | .626 | .600 | .608 | .204 | .236 | .204 | .131 |
| MidiBERT-f | Tokenized symbolic sequences | .592 | .588 | **.651** | .618 | .179 | .203 | .238 | .185 |
| MidiBERT-n[1] | Tokenized symbolic sequences | .694 | - | - | - | - | - | - | - |
| M2BERT[1] | Tokenized symbolic sequences | **.725** | - | - | - | - | - | - | - |
| *Fix Test Set: (ii) evaluating symbolic-feature-based models* | | | | | | | | | |
| CNN | Piano roll input (w/ replay matrix & inst. names & bar lines) | .667 | .673 | .663 | .668 | .319 | .346 | .335 | .340 |
| RF | Stat. features (track-bar & all-bar) & inst. names | **.745** | **.731** | **.748** | **.740** | **.401** | **.414** | **.455** | **.434** |
| CNN+RF | Combined input and features from CNN and RF rows above | .723 | .705 | .726 | .715 | .358 | .369 | .393 | .381 |
| *18-fold CV on the Orchestration dataset* | | | | | | | | | |
| CNN | Piano roll input (w/ replay matrix & inst. names & bar lines) | .664 | .686 | .663 | .664 | .332 | .332 | .332 | .309 |
| RF | Stat. features (track-bar & all-bar) & inst. names | .704 | .709 | .704 | .697 | **.382** | **.341** | **.382** | **.352** |
| CNN+RF | Combined input and features from CNN and RF rows above | **.721** | **.728** | **.721** | **.712** | .359 | .310 | .359 | .329 |
| *4-Fold CV on the S3 dataset* | | | | | | | | | |
| MidiBERT-n | Tokenized symbolic sequences | - | - | - | - | **.440** | - | - | - |
| RF | Stat. features (track-bar & all-bar) & inst. names | **.536** | **.562** | **.536** | **.533** | .400 | **.392** | **.400** | **.367** |

[1] Results copied from [30].

### A. Overall Model Performance

Table II presents the results, with boldface indicating the highest value in each column for each evaluation protocol.

**Fixed Test Set.** Among DL models trained from scratch using piano roll input, BiLSTM achieves the highest accuracy on the Orchestration test set but requires nearly 500 epochs to converge, making it the most computationally expensive. In contrast, CNN reaches the highest F1 score on both datasets within just 150 epochs, offering the best trade-off between performance and training efficiency. CRNN and Transformer attain their best performance around 200 epochs, further highlighting CNN's advantage as the most lightweight and effective option within this group.

In comparison, pre-trained models achieve even higher performance. M2BERT obtains the highest overall accuracy across all settings, surpassing all simple models. Between the two MidiBERT variants, MidiBERT-n outperforms MidiBERT-f, which is expected: the former is fully fine-tuned in our setting, while the latter is kept frozen.

Regarding evaluation for symbolic-feature-based models, enhancing the CNN input with replay matrices, instrument names, and bar lines yields an improvement of 2.9% in accuracy. Among these settings, RF with statistical features and instrument names performed the best. Notably, this model does not rely on any note-level piano roll input.

**k-fold CV.** To assess generalizability, we apply 18-fold CV on Orchestration dataset, and results show that CNN+RF consistently achieves the most robust performance across pieces, surpassing RF in this setup. However, none of the models exceed 40% accuracy when evaluated on the full S3 dataset. To examine whether this limitation is solely due to the models not being trained on S3, we conduct 4-fold CV on the S3 dataset. Although MidiBERT-n achieves the highest F1 score (0.44), closely followed by RF (0.40), the results suggest that factors beyond the training–testing mismatch contribute to the poor performance. Further discussion is provided in Part C.

These findings highlight both the promise of pre-trained models and the robustness of RF. RF outperforms simple deep learning models as statistical features can directly capture texture-related patterns, whereas deep models either lack sufficient data or are constrained by their inductive biases. More detailed experiment results and case studies can be found at `https://github.com/clarelinzx/orchestral-texture-classification`.

### B. Feature Importance from RF Models

Given the strong and consistent performance of RF models, we further examine which features contribute most to classification by computing the *cumulative reciprocal rank* (CRR) for each feature $f$, defined as $\mathrm{CRR}(f) = \sum_{i=1}^{18} 1/\mathrm{rank}_i(f)$, where $\mathrm{rank}_i(f)$ denotes the position of feature $f$ in the descending order of feature importance in the $i$-th fold (i.e., more important features receive smaller rank values).

As table III shows, duration-related statistical features consistently dominate the top ranks. Track-bar statistics, including shortest, mean, normalized, and longest duration, are particularly influential. While these features do not directly encode rhythmic patterns, their distributions shape rhythmic density, on which annotators often rely to distinguish rhythmic from harmonic roles [12]. Notably, these features outrank interval- and pitch-based features, suggesting that duration-based cues offer greater discriminative power in this context.

TABLE III
THE CUMULATIVE RECIPROCAL RANK (CRR) OF THE TOP 15 FEATURES ACROSS 18 FOLDS OF THE RF MODEL

| Feature | CRR |
|---|---|
| shortest duration (track-bar) | 17.000 |
| mean duration (track-bar) | 9.667 |
| normalized duration (track-bar) | 6.333 |
| longest duration (track-bar) | 4.500 |
| std duration (all-bar) | 3.400 |
| occupation rate (track-bar) | 3.176 |
| mean duration (all-bar) | 2.482 |
| number of syncopated note (all-bar) | 2.363 |
| normalized duration (all-bar) | 1.926 |
| std interval (track-bar) | 1.790 |
| number of different intervals (all-bar) | 1.673 |
| number of different intervals (track-bar) | 1.446 |
| longest duration (all-bar) | 1.372 |
| std pitch (track-bar) | 1.345 |
| std pitch (all-bar) | 1.163 |

Meanwhile, *number of syncopated notes* ranks eighth in CRR, with its all-bar version proving more informative than the track-bar counterpart. This suggests that rhythmic displacement is perceived in relation to the broader ensemble context rather than within a single part, aligning with the intuition that texture are shaped by temporal interplay across instruments.

The strong performance of duration-related features points to the opportunities for integrating rhythmic information with statistical features in related tasks such as generation, where pitch and harmony have previously received more attention.

### C. Annotation Consistency Across Datasets

To investigate how the model performs on different datasets, Fig. 2 presents the confusion matrices of two datasets with RF model under the evaluation protocol of the fixed test set. Both matrices reveal consistent challenges in classifying mixed roles. In particular, `M` is frequently misclassified as `R`, likely due to the rhythmic variability inherent in melodic lines, which can blur the distinction between roles. This ambiguity aligns with our feature importance analysis (Table III), where the highest ranking features primarily capture duration-based information rather than pitch-related cues.

Using the Orchestration dataset as ground truth and the S3 dataset as predictions, we report the accuracy and weighted macro-averaged precision, recall, and F1 scores in Table IV, where M and B denote Mozart and Beethoven, respectively. Per-label F1 scores for `M`, `R`, and `H` are also shown. This table highlights annotation inconsistencies across the datasets. Moreover, the S3 dataset tends to assign mixed roles involving melodic functions (e.g., `MR`, `MH`) to track-bar units, whereas the Orchestration dataset more strictly avoids assigning melody in conjunction with rhythmic or harmonic accompaniment (see Table I). Such different annotation conventions not only lead to discrepancies in model performance but also pose challenges for aligning texture classification results across corpora.

## V. CONCLUSIONS

We performed a comparative study on the orchestral texture classification task, with multiple models evaluated across the only two available symbolic datasets with texture annotations (Orchestration and S3). Constructing such annotations is time-consuming, and even for overlapping pieces, the consistency between datasets is limited, underscoring both the need for automatic labeling methods and the inherent difficulty of the task. Results suggest that statistical features, especially those related to note durations, play a critical role in texture analysis.

Our initial motivation was to train supervised models to generate pseudo-labels for large unlabeled multi-track datasets. However, results reveal that even simple classifiers, such as RFs, when equipped with well-designed symbolic features, can be more reliable than complex deep learning models under limited data. This suggests that statistical features may already capture textural roles such as rhythm and harmony, while melody remains challenging to classify due to its variability and context dependence. Overall, rather than pursuing state-of-the-art performance, this study serves as a first step toward clarifying the feasibility of symbolic orchestral texture classification and identifying the most promising modeling strategies for future research.

Future work may treat melody extraction in orchestral music as a distinct task, supported by clearer definitions of melodic function and more consistent annotation standards. Additionally, since this study focused exclusively on role classification, future research may incorporate stream segmentation to address the multi-layered nature of texture more holistically. Combining both perspectives could lead to a more comprehensive understanding of how texture is organized and perceived across time and instrumentation.

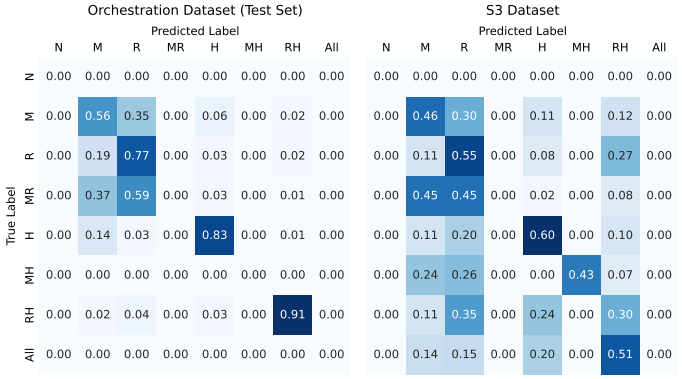


Fig. 2. Confusion matrices on two datasets.

TABLE IV
CROSS-DATASET EVALUATION RESULTS

| Movement | Acc. | Prec. | Rec. | F1 | F1(`M`) | F1(`R`) | F1(`H`) |
|---|---|---|---|---|---|---|---|
| M No.41 (I) | .300 | .498 | .327 | .395 | .525 | .296 | .196 |
| B No.9 (I) | .282 | .711 | .307 | .429 | .344 | .481 | .124 |

## ACKNOWLEDGMENT

This project is supported in part by the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2022-1004). The authors would like to thank Dinh-Viet-Toan

Le from Université de Lille for providing feedback on earlier versions of the paper.

## REFERENCES


[1] J. M. Levy, "Texture as a sign in classic and early romantic music," *Journal of the American Musicological Society*, vol. 35, no. 3, pp. 482–531, 1982.

[2] M. Giraud, F. Levé, F. Mercier, M. Rigaudière, and D. Thorez, "Towards modeling texture in symbolic data," in *Proc. of ISMIR*, 2014, pp. 59–64.

[3] T.-P. Chen, L. Su, and K. Yoshii, "Learning multifaceted self-similarity for musical structure analysis," in *APSIPA*, 2023, pp. 165–172.

[4] M. Buisson, B. Mcfee, and S. Essid, "Using pairwise link prediction and graph attention networks for music structure analysis," in *Proc. of the Int. Society for Music Information Retrieval Conf. (ISMIR)*, 2024.

[5] T.-P. Chen and L. Su, "Functional harmony recognition of symbolic music data with multi-task recurrent neural networks," in *Proc. of ISMIR*, 2018, pp. 90–97.

[6] T.-P. Chen and L. Su, "Harmony Transformer: Incorporating chord segmentation into harmony recognition," in *Proc. of ISMIR*, 2019, pp. 259–267.

[7] Y.-W. Hsiao, T.-Y. Hung, T.-P. Chen, and L. Su, "BPS-Motif: A dataset for repeated pattern discovery of polyphonic symbolic music," in *Proc. of ISMIR*, 2023, pp. 281–288.

[8] Y.-W. Hsiao and L. Su, "Learning note-to-note affinity for voice segregation and melody line identification of symbolic music data," in *Proc. of ISMIR*, 2021, pp. 285–292.

[9] X. Ma, X. Liu, B. Zhang, and Y. Wang, "Robust melody track identification in symbolic music," in *Proc. of ISMIR*, 2022, pp. 842–849.

[10] L. Soum-Fontez, M. Giraud, N. Guiomard-Kagan, and F. Levé, "Symbolic textural features and melody/accompaniment detection in string quartets," in *Int. Symposium on Computer Music Multidisciplinary Research*, 2021, pp. 175–184.

[11] Y.-H. Chu and L. Su, "Orchestral texture classification with convolution," in *Extended Abstracts for the Late-Breaking Demo Session (LBD) of the 24th ISMIR*, 2023.

[12] D.-V.-T. Le, M. Giraud, F. Levé, and F. Maccarini, "A corpus describing orchestral texture in first movements of classical and early-romantic symphonies," in *Proc. of the Int. Conf. on Digital Libraries for Musicology*, 2022, pp. 27–35.

[13] Z. Zhao, "Let Network Decide What to Learn: Symbolic music understanding model based on large-scale adversarial pre-training," *arXiv preprint*, 2024.

[14] Y.-H. Chou, I.-C. Chen, C.-J. Chang, J. Ching, and Y.-H. Yang, "BERT-like pre-training for symbolic piano music classification tasks," *Journal of Creative Music Systems*, vol. 8, no. 1, pp. 1–19, 2024.

[15] H. H. Mao, T. Shin, and G. Cottrell, "DeepJ: Style-specific music generation," in *IEEE 12th international conference on semantic computing*, 2018, pp. 377–382.

[16] D. Huron, "Note-onset asynchrony in J. S. Bach's two-part inventions," *Music Perception*, vol. 10, no. 4, pp. 435–443, 1993.

[17] B. Duane, "Auditory streaming cues in eighteenth-and early nineteenth-century string quartets:a corpus-based study," *Music Perception*, vol. 31, no. 1, pp. 46–58, 2013.

[18] B. Duane, "Thematic and non-thematic textures in Schubert's three-key expositions," *Music Theory Spectrum*, vol. 39, no. 1, pp. 36–65, 2017.

[19] J. D. Souza, C. Dvorsky, and O. Oyon, "Texture and sonata form in classical string quartets: A corpus study," *Empirical Musicology Review*, vol. 19, no. 2, pp. 102–110, 2024.

[20] M. L. Lavengood, "Timbre, rhythm, and texture within music theory's white racial frame," in *The Oxford Handbook of Electronic Dance Music*, 2021.

[21] J. Covach, *Analyzing Texture in Rock Music: Stratification, Coordination, Position, and Perspective*. Universitätsbibliothek, 2020.

[22] P. A. Kokoras, "Towards a holophonic musical texture," in *Proc. of the Int. Computer Music Conference*, 2005.

[23] L. Couturier, L. Bigo, and F. Levé, "A dataset of symbolic texture annotations in mozart piano sonatas," in *Proc. of ISMIR*, 2022, pp. 509–516.

[24] Z.-S. Lin*, Y.-C. Kuo*, T.-Y. Hung, *et al.*, "S3: A symbolic music dataset for computational music analysis of symphonies," in *Proc. of ISMIR LBD session*, 2024.

[25] D. Rafailidis, A. Nanopoulos, Y. Manolopoulos, and E. Cambouropoulos, "Detection of stream segments in symbolic musical data," in *Proc. of ISMIR*, 2008, pp. 83–88.

[26] D. Rizo, P. J. P. De León, A. Pertusa, C. Pérez-Sancho, and J. M. I. Quereda, "Melody track identification in music symbolic files," in *Proc. of the Int. Florida Artificial Intelligence Research Society Conf.*, 2006, pp. 254–259.

[27] Z. Wang, K. Zhang, Y. Wang, *et al.*, "Songdriver: Real-time music accompaniment generation without logical latency nor exposure bias," in *Proc. of the ACM Int. Conf. on Multimedia*, 2022, pp. 1057–1067.

[28] F. Maccarini, M. Oudin, M. Giraud, and F. Levé, "Co-creative orchestration of angeles with layer scores and orchestration plans," in *Int. Conf. on Artificial Intelligence in Music, Sound, Art and Design*, Springer, 2024, pp. 228–245.

[29] D.-V.-T. Le and Y.-H. Yang, "METEOR: Melody-aware texture-controllable symbolic orchestral music generation," *arXiv preprint*, 2024.

[30] J.-Y. Wang and L. Su, "Improving BERT for symbolic music understanding using token denoising and pianoroll prediction," in *Proc. of ISMIR*, 2025.