

Peransformer: Improving Low-informed Expressive Performance Rendering with Score-aware Discriminator

Xian He*, Wei Zeng*[†], and Ye Wang*^{†‡}

* School of Computing, National University of Singapore

[†] Integrative Sciences and Engineering Programme, NUS Graduate School

E-mail: {xian.he, w.zeng}@u.nus.edu.sg

[‡]Corresponding author. Email: wangye@comp.nus.edu.sg

Abstract—Highly-informed Expressive Performance Rendering (EPR) systems transform music scores with rich musical annotations into human-like expressive performance MIDI files. While these systems have achieved promising results, the availability of detailed music scores is limited compared to MIDI files and are less flexible to work with using a digital audio workstation (DAW). Recent advancements in low-informed EPR systems offer a more accessible alternative by directly utilizing score-derived MIDI as input, but these systems often exhibit suboptimal performance. Meanwhile, existing works are evaluated with diverse automatic metrics and data formats, hindering direct objective comparisons between EPR systems. In this study, we introduce Peransformer, a transformer-based low-informed EPR system designed to bridge the gap between low-informed and highly-informed EPR systems. Our approach incorporates a score-aware discriminator that leverages the underlying score-derived MIDI files and is trained on a score-to-performance paired, note-to-note aligned MIDI dataset. Experimental results demonstrate that Peransformer achieves state-of-the-art performance among low-informed systems, as validated by subjective evaluations. Furthermore, we extend existing automatic evaluation metrics for EPR systems and introduce generalized EPR metrics (GEM), enabling more direct, accurate, and reliable comparisons across EPR systems. The repository is available at <https://github.com/Bigstool/peransformer>.

I. INTRODUCTION

Musical expressivity arises not only from composition but also from the performer’s interpretation and delivery. Nuances in tempo, dynamics, articulation, and other performance parameters critically influence how a piece resonates with listeners [1]. Expressive Performance Rendering (EPR) refers to the computational task of rendering performances that mimic human expressiveness from existing music compositions. EPR systems can serve as virtual reference performers in music education, function as plugins to streamline music production, and operate as surrogates for human performance in tasks such as Audio-to-Score Transcription (A2S) [2], particularly in scenarios where human-recorded data is limited.

EPR systems are generally categorized by their input modalities: highly-informed systems leverage score-level information such as tempo, time signature, key signature, and note value, whereas low-informed systems operate on more accessible note-level information like onset, offset, and velocity. While

highly-informed systems have achieved convincing results [3]–[6], their flexibility and accessibility are inherently constrained — score notations (e.g., MusicXML) are not only harder to obtain but also more cumbersome to manipulate compared to MIDI-based representations, especially when working with a digital audio workstation (DAW). Recent advances in low-informed EPR systems [7], [8] address this issue by directly using score-derived MIDI as input, offering greater practicality. However, subjective evaluations have shown that their expressive quality still falls short compared to highly-informed models. Additionally, many of these systems are trained on unpaired [7] or pseudo-paired [8] data, limiting their ability to render performances coherent with the underlying composition.

Another challenge is the evaluation of EPR models using automatic metrics. Various metrics, such as mean squared error (MSE) [4], [5], [9], Pearson correlation coefficient [4], [5], [9], and mean absolute error (MAE) [8], have been employed for this purpose. However, current evaluation procedures apply these metrics directly to raw model outputs. Converting model outputs into the standard MIDI format often disrupts input-target alignment due to expressive variations in note ordering, making automatic evaluation non-trivial. Furthermore, differences in data representations and evaluation workflows across studies make direct comparisons between results difficult.

To address these limitations, we propose Peransformer, a low-informed expressive performance rendering (EPR) model based on the Transformer encoder [10]. In order to encourage the renditions to remain faithful to the composition, we utilize a score-aware discriminator, which conditions on MIDI files derived from the original score. We construct ASAP-MIDI, a note-to-note aligned, score-to-performance paired dataset with the ASAP dataset [11] and an alignment tool [12] for the training of Peransformer. Subjective listening tests demonstrate that our model significantly outperforms existing low-informed EPR approaches and greatly narrows the performance quality gap between low-informed and high-informed models.

Furthermore, we introduce Generalized EPR Metrics (GEM) — a set of evaluation tools that standardize both data formatting and metric computation. GEM facilitates consistent evaluation

Split	Composers	Compo- sitions	Score Notes	Performance Duration
Train	14	168	441k	56.6h
Val	9	22	62.5k	11.2h
Test	7	20	53.0k	6.16h
Total	15	210	556k	74.0h

TABLE I
STATISTICS OF THE ALIGNED AND SPLIT DATASET.

by using alignment tools [12], enabling comparisons across any EPR model that outputs MIDI data.

In summary, our contributions are threefold:

- 1) Performer – a novel low-informed EPR model with a score-aware discriminator conditioned on the original composition.
- 2) ASAP-MIDI - an open access, score-to-performance paired, note-to-note aligned dataset for training low-informed EPR models.
- 3) Generalized EPR Metrics (GEM) – a standardized evaluation workflow that enables direct, accurate, and reliable comparisons between EPR models.

II. METHODOLOGY

A. Dataset

We source the score and performance MIDI files of ASAP-MIDI from the Aligned Scores and Performances (ASAP) dataset [11], a classical piano music dataset. However, ASAP only provides beat-level alignment. We obtain score-to-performance, note-to-note alignment using the alignment tool proposed by Nakamura et al. [12]. Each composition has at least one human performance.

As a quality control measure, a performance MIDI is discarded if more than 6% of the notes in both the score and the performance MIDI failed to match, or if more than 3% of the pairs of matched notes have different pitches. The thresholds are empirically chosen so that no obvious difference between performances before and after alignment is perceived, and approximately 90% of the human performances in the dataset are retained.

For every performance, we further scale the score MIDI to match its length. Finally, the dataset is split with a rough ratio of 8:1:1. The statistics of the processed dataset are shown in Table I.

B. Data Representation

We adopted the encoding method for model input in [7] and applied it to both the score and performance MIDI. A performance $\mathbf{p} = (n_1, n_2, n_3, \dots)$ is defined as a list of notes, where the j -th note $n_j = (i_j, d_j, p_j, v_j)$ is a tuple of the Inter-Onset Interval (IOI) in seconds $i \in \mathbb{R}_{\geq 0}$, duration in seconds $d \in \mathbb{R}_{\geq 0}$, MIDI note number $p \in \{m \in \mathbb{Z} \mid 0 \leq m \leq 127\}$, and MIDI velocity $v \in \{m \in \mathbb{Z} \mid 0 \leq m \leq 127\}$. Specifically, i_j is calculated as:

$$i_j = \begin{cases} 0 & \text{if } j = 1 \\ o_j - o_{j-1} & \text{if } j = 2, 3, \dots, m \end{cases}, \quad (1)$$

and d_i is calculated as:

$$d_j = f_j - o_j \text{ for } j = 1, 2, \dots, m, \quad (2)$$

where o is the absolute onset time and f is the absolute offset time, in seconds. We further apply z-score scaling to the four features as a standardization measure.

C. Model Architecture

1) *Performance Model*: The performance model is comprised of a score embedding layer, a feature extractor layer, and a regression head, as shown in Figure 1. The score embedding layer is a fully connected layer that transforms the dimensions of the input \mathbf{p}^s to the dimensions of the feature extractor. A sinusoidal positional encoding from [10] is added to the score embedding before being passed into the feature extractor, which is a stack of transformer encoder blocks [10]. The regression head is another fully connected layer with an output dimension of 3, corresponding to the predicted IOI, duration, and velocity. Since the pitches of the notes are unchanged during this process, they are copied from the input \mathbf{p}^s and assembled with the predicted values as the rendition \mathbf{p}^e .

2) *Discriminator*: The score-aware discriminator takes either the rendition \mathbf{p}^e or the human performance \mathbf{p}^h as input through the performance embedding layer, while conditioning on the score MIDI \mathbf{p}^s passed through the score embedding. Also shown in Figure 1, all layers share the same architecture as the performance model, with the difference that the two embeddings are added together with the positional embedding, and the output dimension of the classification head is 1, producing the classification logit.

3) *Loss Functions*: The loss function of the performance model Ψ^P given the discriminator Ψ^D is formulated as:

$$L_{\Psi^P} = \frac{\lambda_0 \text{BCE}(\Psi^D(\mathbf{p}^s, \mathbf{p}^e))}{c} + \frac{\lambda_1 \text{MSE}(\mathbf{i}^h, \mathbf{i}^e)}{c} + \frac{\lambda_2 \text{MSE}(\mathbf{d}^h, \mathbf{d}^e)}{c} + \frac{\lambda_3 \text{MSE}(\mathbf{v}^h, \mathbf{v}^e)}{c}, \quad (3)$$

where $\text{BCE}()$ is the binary cross-entropy loss. $\text{MSE}()$ is the mean squared error loss, which is added to stabilize training [13]. \mathbf{i} , \mathbf{d} , \mathbf{v} , are the lists of IOI, duration, and velocity from the corresponding performance. λ are the weight balancing terms. c is the count of human performances of the composition in the dataset, which balances the highly skewed dataset with various numbers of performances for each composition.

The loss of the discriminator is formulated as:

$$L_{\Psi^D} = \frac{\lambda_4 \text{BCE}(\Psi^D(\mathbf{p}^s, \mathbf{p}^h))}{c} + \frac{\lambda_5 \text{BCE}(1 - \Psi^D(\mathbf{p}^s, \mathbf{p}^e))}{c}. \quad (4)$$

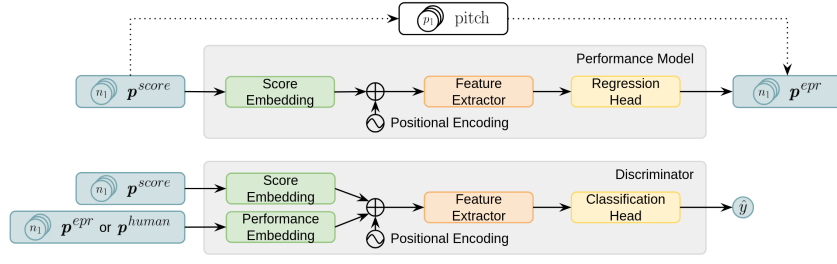


Fig. 1. The performance model and the discriminator.

D. The Generalized EPR Metrics (GEM)

Following commonly used automatic metrics for the evaluation of EPR models, we calculate the mean squared error (MSE) and Pearson correlation coefficient between the rendition and the human performance over the IOI, duration, and velocity features. To allow for MIDI-based evaluation, hence enabling the application of GEM on any EPR model with MIDI output, we apply the aforementioned alignment tool [12] on the rendition and the human performance to recover the note-to-note correspondence which may be perturbed during the conversion from the raw model output to MIDI. Since each composition in the dataset can have multiple human performances, drawing inspiration from ROUGE [14], an established metric for text summarization, we treat multiple performances of the same composition as multiple references. One result is calculated between the rendition and each reference, and the best result is kept to represent the performance of the model on the given composition.

To enhance the reproducibility of GEM, we proceed to define a standardized evaluation process and a unified data format, starting from the following preprocessing steps:

- Align the renditions to each of the human performances of the corresponding composition using [12].
- Convert the aligned MIDI files to the data representation defined in II-B, but without z-score scaling to preserve all data in the original units.

Then, without loss of generality, the application of GEM on the duration feature is given in Algorithm 1. The same can also be applied to the IOI and velocity predictions.

III. EXPERIMENTS

A. Experimental Setup

For both the performance model and the discriminator, a learning rate of 1×10^{-5} and batch size of 4 is used. The values used for the weight-balancing factors λ_0 to λ_5 are set to $[1, 3, 0.1, 0.1, 1, 1]$. Furthermore, the discriminator is trained every 5 epochs. For the feature extractor, the number of encoder blocks in the stack is 6. Each encoder block with the dimensionality of input and output $d_{\text{model}} = 256$, inner-layer the dimensionality $d_{ff} = 1024$, and the number of heads $h = 4$. The hyperparameters are chosen through coarse parameter search.

We employ early stopping to elicit three models with the best validation performance in terms of IOI, duration, and

Algorithm 1: Duration Metric Calculation

Input: comps_e , comps_h : Lists of compositions by EPR model or human, each composition is a list of performances

Output: l : MSE, p : Pearson correlation

```

1 Initialize  $l\_sum$ ,  $l\_n$ ,  $p\_sum \leftarrow 0$ ;
2 foreach ( $\text{comp\_epr}$ ,  $\text{comp\_human}$ ) in ( $\text{comps\_e}$ ,
    $\text{comps\_h}$ ) do
3   Initialize  $l\_best \leftarrow \infty$ ,  $n\_best \leftarrow 0$ ,  $p\_best \leftarrow -1$ ;
4   foreach ( $\text{perf\_e}$ ,  $\text{perf\_h}$ ) in ( $\text{comp\_epr}$ ,
    $\text{comp\_human}$ ) do
5      $loss \leftarrow \text{MSE}(\text{perf\_e}['dur'], \text{perf\_h}['dur'])$ ;
6     if  $loss < l\_best$  then
7        $l\_best \leftarrow loss$ ;
8        $n\_best \leftarrow \text{len}(\text{perf\_h}['dur'])$ ;
9      $pearson \leftarrow \text{PearsonR}(\text{perf\_e}['dur'], \text{perf\_h}['dur'])$ ;
10    if  $pearson > p\_best$  then
11       $p\_best \leftarrow pearson$ ;
12     $l\_sum \leftarrow l\_sum + (l\_best \times n\_best)$ ;
13     $n\_notes \leftarrow n\_notes + n\_best$ ;
14     $p\_sum \leftarrow p\_sum + p\_best$ ;
15  $l \leftarrow l\_sum / n\_notes$ ;
16  $p \leftarrow p\_sum / \text{len}(\text{comps\_e})$ ;
17 return  $l$ ,  $p$ ;

```

velocity features separately. By merging the predictions for the three features, we combine the models into Peransformer, an ensemble model.

Various Python libraries are used to process MIDI data [15], to build and train the model [16], and to calculate the Pearson correlation coefficient and perform hypothesis tests [17].

B. Ablation Study

To understand the role of various modules of the proposed model, and to facilitate later evaluation of the Generalized EPR Metrics (GEM), we perform an ablation study by applying Algorithm 1 on the raw model output with z-score scaling removed, reproducing existing objective evaluation procedures for EPR models. The ablation models are: No c : set c to 1 to remove loss balancing between compositions. No D pause: train the discriminator every epoch. No λ : set λ to 1 to remove loss weighting. No D score: remove the score embedding from the discriminator to make it unconditional. No MSE: train with

Model	$l_{ioi} \downarrow$	$l_{dur} \downarrow$	$l_{vel} \downarrow$	$p_{ioi} \uparrow$	$p_{dur} \uparrow$	$p_{vel} \uparrow$
Ensemble	0.0114	0.0896	220.47	0.9016	<u>0.7589</u>	<u>0.5122</u>
No c	0.0129	0.1770	<u>214.89</u>	0.8949	0.6230	0.4377
No D pause	<u>0.0116</u>	0.0994	342.72	0.8938	0.6336	0.3593
No λ	0.0127	<u>0.0860</u>	253.06	0.8955	0.7568	0.4876
No D score	0.0125	0.1033	247.40	<u>0.8993</u>	0.7355	0.4841
No MSE	0.0154	0.2012	352.42	<u>0.8772</u>	0.7074	0.3981
No D	0.0119	0.0788	204.54	0.8972	0.7674	0.5361

TABLE II
RESULTS OF THE ABLATION STUDY. THE BEST RESULTS ARE SHOWN IN BOLD FACE AND THE SECOND-BEST RESULTS ARE MARKED WITH UNDERLINE.

	$l_{ioi} \uparrow$	$l_{dur} \uparrow$	$l_{vel} \uparrow$	$p_{ioi} \uparrow$	$p_{dur} \uparrow$	$p_{vel} \uparrow$
36 @ 99%	0.9296	0.8354	0.9971	0.7798	0.9695	0.9952
All 180	0.9534	0.8702	0.9800	0.7401	0.8462	0.9918

TABLE III
PEARSON CORRELATION COEFFICIENTS BETWEEN THE RESULTS OF III-B AND GEM.

GAN loss only. No D: train with MSE loss only. The results are given in Table II.

Notably, the no D score model resembles the one proposed in [7], which also leverages unconditional GAN. However, a major difference remains between the two models: while the MSE loss in [7] is calculated between the rendition and the score MIDI, our work calculates the loss between the rendition and the human performance.

Despite obtaining conspicuous results under objective evaluation, the no D model is arguably abusing the metrics by directly optimizing towards them, resulting in outputs that are "in the middle" of the distribution of human performances from the perspective of the metrics. While these outputs may have low losses, they are not necessarily inside of the distribution. Further inspections of the rendition samples also suggest that they are very similar to the input score MIDI file than to the target human performance, with less expressively varied local tempo, articulations, and dynamics compared to the ensemble model, corroborating the hypothesis.

Other ablation models exhibit various performance decays, evidenced by both automatic metrics results and inspections on rendition samples. More noticeable issues include unstable tempo (no c , no λ , no MSE), and sudden changes in dynamics (no D pause, no MSE). Overall, the ensemble model is the most well-rounded model among the ablation models.

C. Generalized EPR Metrics (GEM)

To examine the reliability of GEM as a replacement for existing objective evaluation procedures, we reconduct the ablation study using GEM, and compare the results with the ones obtained in III-B. With the ensemble model being disassembled into the underlying IOI, duration, and velocity models, 180 result pairs are acquired from the 20 compositions of the test split using the 9 models being evaluated. The

Pearson correlation coefficients between the results are shown in Table III. "36 @ 99%" shows the results calculated from the 36 samples with more than 99% of notes matched in the alignment, representing the performance of GEM at the highest alignment accuracy. "All 180" is calculated from all of the 180 samples, indicating the performance of GEM under typical conditions.

The results suggest that GEM make accurate evaluations. A very strong correlation with current metrics can be seen in the IOI loss, duration loss, velocity loss, duration Pearson, and velocity Pearson metrics, while a strong correlation is observed in the IOI Pearson metric.

GEM also show high reliability and remain robust against varying alignment accuracies, shown by the very similar correlation between GEM and current metrics on compositions where the alignment accuracy is high (36 @ 99%) and typical (all 180).

D. Quantitative Evaluation

Taking advantage of the ability of GEM to evaluate any EPR model with output in the MIDI format, we conduct a quantitative evaluation on Peransformer and existing EPR models. The models being compared include two recent low-informed models [7], [8], and one high-informed model [5] which achieved state-of-the-art performance. On top of making direct comparisons among EPR models, we take one step further and apply GEM to the score MIDI files and human performances for additional context.

We adopt the procedure proposed in [18] to approximate human performance. The first human performance of each composition is taken as the surrogate of human expressiveness, and we evaluate it against the remaining human performances. As this procedure would require each composition to have at least 2 human performances, the quantitative evaluation was conducted on the subset of 13 compositions of the test dataset that meets this requirement. The results are shown in Table IV. "Match %" shows the percentage of rendition notes successfully matched with the human performance.

In comparison, [7] appears to struggle with the velocity predictions, possibly due to its tendency to produce high-velocity predictions. Among the renditions for the 20 compositions of the test split, 8 of which have the velocity clipped to the maximum value of 127 for every note in the composition. [8] often greatly accelerates the composition. For example, the input score MIDI of the second movement of Piano Sonata No. 11 by Beethoven is 459 seconds long. However, the rendition was shrunk to only 76 seconds by [8], potentially harming its IOI metric results.

The results suggest that Peransformer performs significantly better in the velocity metrics while archiving better or similar results in the IOI and duration metrics against existing low-informed EPR models. Meanwhile, our model greatly narrows the gap to the highly-informed model from [5]. The approximated human performance remains better than the EPR models.

Model Type	Model	$l_{ioi} \downarrow$	$l_{dur} \downarrow$	$l_{vel} \downarrow$	$p_{ioi} \uparrow$	$p_{dur} \uparrow$	$p_{vel} \uparrow$	Match %
Low-informed	Ours	0.0210	0.1528	233.91	0.8956	0.7802	0.5378	97.12
	Renault <i>et al.</i> [7]	0.0205	0.5291	2491.09	0.9006	0.5805	0.2668	96.61
	Tang <i>et al.</i> [8]	0.0550	0.1305	1022.75	0.5035	0.6124	0.0996	91.88
Highly-informed	Jeong <i>et al.</i> [5]	0.0102	0.0710	136.51	0.9277	0.8057	0.6397	97.31
	Score	0.0197	0.1768	798.82	0.9042	0.7897	0.3157	95.51
	Human	0.0071	0.0618	168.75	0.9419	0.8704	0.6761	96.34

TABLE IV

QUANTITATIVE ANALYSIS WITH GEM. THE BEST RESULTS UP TO THE NEXT HORIZONTAL RULE ARE HIGHLIGHTED IN **BOLD FACE**.

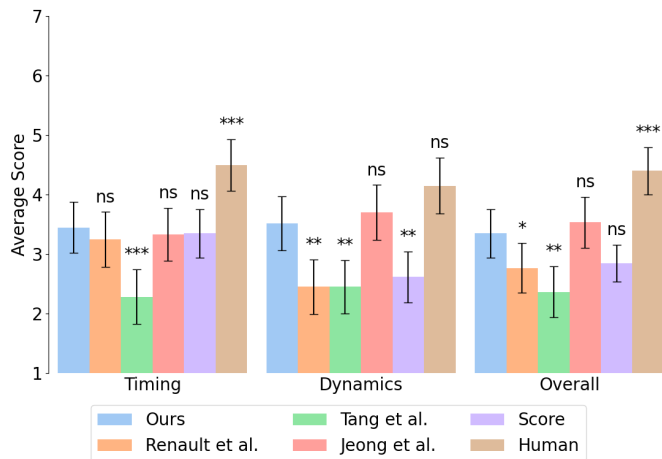


Fig. 2. Results of the subjective evaluation. The bar charts show the Mean Opinion Score (MOS) of the models. The error bars show the 95% Confidence Interval (CI). The statistical significance of Welch’s t-tests between Peransformer and the corresponding model is indicated above the error bars. “ns”, “*”, “**”, and “***” denote $p \geq 0.05$, $p < 0.05$, $p < 0.01$, and $p < 0.001$ respectively.

E. Subjective Evaluation

We conduct a survey to evaluate Peransformer and existing models qualitatively. 5 compositions with different compositional contents were selected from the test split, namely Beethoven’s Piano Sonata No. 1 in F Minor, Op. 2 No. 1, Chopin’s Etude in C major Op. 10 No. 7, Chopin’s Ballade No. 2 in F major, Op. 38, Schubert’s The Fantasie in C major, Op. 15, and Bach’s Fugue in E-flat Major, BWV 876. The renditions are synthesized into audio clips using the same soundfont without pedaling. The audio clips are then trimmed to the first minute.

In this study, 60 participants aged over 21 were recruited. Among the participants, 51 (85%) identified themselves as regular music listeners who listen to the music for more than 1 hour every week. 46 (76%) claim to have at least 1 year of experience with some musical instruments.

The participants were asked to score each audio clip on a 7-point Likert scale regarding timing expressiveness, dynamics expressiveness, and overall expressiveness. Timing expressiveness evaluates the expressiveness of local tempo and articulation, relating to the IOI and duration predictions. Similarly, dynamics expressiveness is related to the velocity predictions

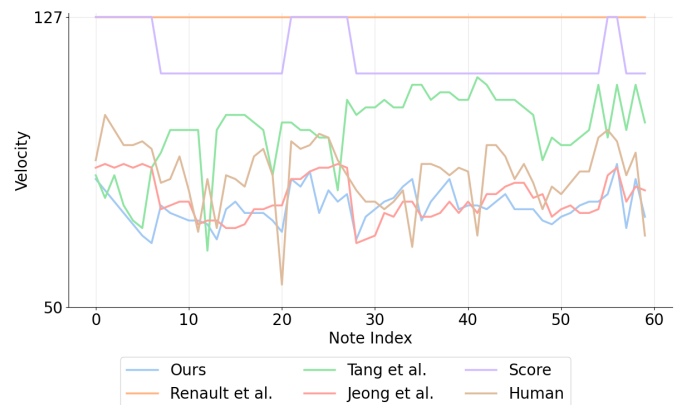


Fig. 3. Velocity changes of the first 60 notes of Schubert’s The Fantasie in C major, Op. 15.

and evaluates the expressiveness of dynamics. Finally, overall expressiveness is a holistic evaluation of the performance with all factors combined.

We discard responses that assign a higher overall expressiveness point to the score MIDI instead of the human performance as a quality control measure. The results are then aggregated across the compositions and shown in Figure 2.

The qualitative results from the survey are fairly consistent with the quantitative results calculated using GEM. For timing, all models, unfortunately, show a similar level of expressiveness to the score MIDI files except for [8], which showed a lower performance. The human performances are still significantly more expressive in timing compared to the others.

In terms of dynamics, both Peransformer and the highly-informed [5] showed high performances, achieving results similar to human performance, surpassing current low-informed models and the score MIDI files.

Overall, Peransformer significantly outperforms existing low-informed EPR models [7], [8] and has achieved results similar to the highly-informed model [5]. The human performance remains the best across all metrics.

F. Case Study: Comparison in Velocity

To better understand the velocity predictions of the Peransformer model, which corresponds to the dynamics expressiveness in the subjective evaluation, we compare the velocity

changes predicted by the EPR models along with the score MIDI and the human performance, using the first 60 notes of Schubert's The Fantasie in C major, Op. 15. The results are shown in Figure 3.

The input score MIDI file arguably does not provide much meaningful information to assist the velocity prediction for the EPR models, as it only alternates between two values. The velocity predictions by [7] are clipped to the maximum value of 127, showing a peculiar example of the observation discussed in III-D. [8] partially followed the human performance, while partially deviating from it.

Among the EPR models, the predictions of Peransformer and [5] closely follow the human performance in general. Corroborating the observations from the quantitative and subjective analysis. Meanwhile, the human performance appears to have a more pronounced phrasing, indicated by the more obvious peaks and dips line chart, possibly adding extra expressiveness to the performance.

IV. CONCLUSION

We proposed Peransformer, a low-informed EPR model trained with a score-aware discriminator and ASAP-MIDI - a score-to-performance paired, note-to-note aligned MIDI dataset. Quantitative and subjective evaluations show that Peransformer has achieved state-of-the-art results among current low-informed EPR models, while greatly narrowing the gap between low-informed and highly-informed EPR models. We also proposed the Generalized EPR Metrics (GEM), a standardized evaluation workflow that enables direct, accurate, and reliable evaluation of any EPR model with output in the MIDI format.

Future work can focus on further improving the performance of EPR models, especially on timing-related aspects, to match human expressiveness. The relation between the IOI, duration, and velocity metrics and the overall subjective expressiveness of performances can be further investigated to pave the road for the development of more comprehensive automatic metrics.

REFERENCES

- [1] C. E. Cancino-Chacón, M. Grachten, W. Goebel, and G. Widmer, "Computational models of expressive music performance: A comprehensive and critical review," *Frontiers in Digital Humanities*, vol. 5, p. 25, 2018.
- [2] W. Zeng, X. He, and Y. Wang, "End-to-end real-world polyphonic piano audio-to-score transcription with hierarchical decoding," *arXiv preprint arXiv:2405.13527*, 2024.
- [3] C. E. C. Chacón and M. Grachten, "The basis mixer: A computational romantic pianist," in *Late-Breaking Demos of the 17th International Society for Music Information Retrieval Conf.(ISMIR)*, 2016.
- [4] D. Jeong, T. Kwon, Y. Kim, K. Lee, and J. Nam, "Virtuosonet: A hierarchical rnn-based system for modeling expressive piano performance.," in *ISMIR*, 2019, pp. 908–915.
- [5] D. Jeong, T. Kwon, Y. Kim, and J. Nam, "Graph neural network for music score data and modeling expressive piano performance," in *International conference on machine learning*, PMLR, 2019, pp. 3060–3070.
- [6] I. Borovik and V. Viro, "Scoreperformer: Expressive piano performance rendering with fine-grained control.," in *ISMIR*, 2023, pp. 588–596.
- [7] L. Renault, R. Mignot, and A. Roebel, "Expressive piano performance rendering from unpaired data," in *International Conference on Digital Audio Effects (DAFx23)*, 2023.
- [8] J. Tang, G. Wiggins, and G. Fazekas, "Reconstructing human expressiveness in piano performances with a transformer network," *arXiv preprint arXiv:2306.06040*, 2023.
- [9] S. Rhyu, S. Kim, and K. Lee, "Sketching the expression: Flexible rendering of expressive piano performance with self-supervised learning," *arXiv preprint arXiv:2208.14867*, 2022.
- [10] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.
- [11] F. Foscarin, A. Mcleod, P. Rigaux, F. Jacquemard, and M. Sakai, "Asap: A dataset of aligned scores and performances for piano transcription," in *International Society for Music Information Retrieval Conference*, 2020, pp. 534–541.
- [12] E. Nakamura, K. Yoshii, and H. Katayose, "Performance error detection and post-processing for fast and accurate symbolic music alignment.," in *ISMIR*, Suzhou, 2017, pp. 347–353.
- [13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [14] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.
- [15] C. Raffel and D. P. Ellis, "Intuitive analysis, creation and manipulation of midi data with pretty_midi," in *15th international society for music information retrieval conference late breaking and demo papers*, 2014, pp. 84–93.
- [16] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.
- [17] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "Scipy 1.0: Fundamental algorithms for scientific computing in python," *Nature methods*, vol. 17, no. 3, pp. 261–272, 2020.
- [18] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.