Comparative evaluation in the wild: Systems for the expressive rendering of music

Kyle Worrall, Zongyu Yin, and Tom Collins

***Abstract*—There have been many attempts to model the ability of human musicians to take a score and perform or render it expressively, by adding tempo, timing, loudness and articulation changes to non-expressive music data. While expressive rendering models exist in academic research, most of these are not open source or accessible, meaning they are difficult to evaluate empirically and have not been widely adopted in professional music software. Systematic comparative evaluation of such algorithms stopped after the last Performance Rendering Contest (RENCON) in 2013, making it difficult to compare newer models to existing work in a fair and valid way. In this paper, we introduce the first transformer-based model for expressive rendering, Cue-Free Express + Pedal (CFE+P), which predicts expressive attributes such as note-wise dynamics and micro-timing adjustments, and beat-wise tempo and sustain pedal use based only on the start and end times and pitches of notes (e.g., inexpressive MIDI input). We perform two comparative evaluations on our model against a non-machine learning baseline taken from professional music software and two open-source algorithms – a feedforward neural network (FFNN) and hierarchical recurrent neural network (HRNN). The results of two listening studies indicate that our model renders passages that outperform what can be done in professional music software such as Logic Pro and Ableton Live.[1]**

***Impact Statement*—While artificial intelligence has seen rapid growth and development across many fields in recent years, the adoption rate of artificially intelligent music technology remains low. Deep learning has been utilized to generate, mix and master music, but there has been a lack of application of expressive rendering algorithms in the industry. In this paper we present the first transformer-based expressive rendering model, which uses minimal input features for the widest possible breadth of application in music software. Our evaluations demonstrate that our algorithm outperforms the industry standard non-machine learning technology, which we propose should be considered the new baseline in a field lacking systematic comparative evaluation.**

***Index Terms*—Artificial Intelligence in art and music, Computer Generated Music, Music Information Retrieval, Neural Networks, Deep Learning.**

Submitted for review on ??/??/??. This work was supported by the EPSRC Centre for Doctoral Training in Intelligent Games / Games Intelligence (IGGI) [EP/S022325/1].

Kyle Worrall is with the Centre for Doctoral Training in Intelligent Games and Game Intelligence, University of York, York, YO10 5DD, UK (e-mail: kyle.worrall@york.ac.uk).

Zongyu Yin was with the University of York, York, YO10 5DD, UK. He is now a Research Scientist with the Speech, Audio and Music Intelligence team at TikTok, London, EC1A 9HP, UK (e-mail: zongyu.yin@outlook.com).

Tom Collins is an Associate Professor in the Frost School of Music, University of Miami, Florida, FL 33146, USA (e-mail: tomethecollins@gmail.com).

This paragraph will include the Associate Editor who handled your paper.

[1]All source code and pre-experiment hypotheses can be accessed via the Open Science Foundation: https://osf.io/6uwjk/.

I. Introduction

THE art of musical performance resides in being able to take a score – which, in addition to the notes themselves, often includes **expressive instructions** or **score cues** regarding the tempo, articulation, and dynamics of notes – and play it for an audience in a manner that is deemed novel yet within cultural and stylistic norms "creative" [66]. One could also say that composers use score cues to indicate to performers how to bring out the character of a piece.

Beyond score cues as directives for performers, there are further annotations that can be added to scores in order to describe the musical information on a deeper level, thus improving the way they can be understood/computed. These include but are not limited to annotations for meter, phrase information, or musicological analyses [48], [43], [34].

Research spanning several decades has explored computational analysis and generation of expressive performances (e.g., [64], [27], [25], [39], [6]), with more recent efforts leveraging neural networks and deep learning [37], [38], [67]. For instance, the VirtuosoNet [37] algorithm takes basic score data (the notes) and additional score cues, and creates deviations in tempo, note timing, articulation (sometimes including pedalling), and dynamics (note velocity) that imitate a human performance. By comparison, the Basis Mixer [6] encodes high-level aspects of the score as a type of score annotation (e.g., tonal tension calculations [34]) to supplement encoded score cues (e.g., *dynamic markings*) and low-level note information when rendering expressive performances.

While architectures and input features grow more complex, however, there is a lack of standardized evaluation of expressive rendering algorithms following the discontinuation of the Performance Rendering Contest (RENCON) in 2014 [39].

Furthermore, while symbolic music generation algorithms are adopted in professional Digital Audio Workstations (DAWs) such as Ableton [30], [29] and Cubase [61], algorithms for the expressive rendering of inexpressive music data have yet to be integrated in DAWs. Rather, DAW users can choose to add uniformly distributed randomized deviations to musical parameters such as loudness and timing, to provide variation (Logic Pro X's "Humanize") or create their own rule-sets to improve upon randomness (Cubase's logical editor). These industry standard are generally overlooked in comparative evaluations [37]–[39].

Below we introduce two transformer-based expressive rendering algorithms, Cue-Free Express (CFE) and Cue-Free Express + Pedal (CFE+P), which require only the start and end

times and pitches of notes as input.[2] While the application of the transformer architecture to music tasks is not novel in and of itself [36], this is the first use of an ensemble of transformers to tackle expressive rendering.

We compare our systems to the aforementioned "Humanize" function, which we propose should be a baseline for future research in this area, as it is an industry standard, and does not require any additional input beyond that found in MIDI. Additionally, we compare our systems to two other deep learning models – the Basis Mixer [6] and VirtuosoNet [37], as well as real human performances [17]. While other existing models could be compared in studies to follow, we must consider the time that listening studies take to complete, and we choose to be representative of the existing open-source systems by choosing the Basis Mixer and VirtuosoNet. We choose these models as both have been shown to perform well in this domain, and because they utilise fundamentally different architectures (LSTM+FFNN and HRNN respectively). Use of the phrase "in the wild" in our title indicates that Basis Mixer and VirtuosoNet's source code was not usable for retraining the models on the same data [17], but they could be used in their pretrained form, and so we did not want to omit them from consideration. Due to this difference in training data, however, the description "in the wild" is a better characterisation than "systematic" of the comparative evaluation we provide.

The remaining sections of this paper are structured as follows. First, we provide a review of the literature surrounding music performance science (MPS), expressive rendering, music generation, and evaluation. Second, we discuss the aims of our machine learning approach and how the dataset is prepared. Third, we introduce Cue-free Express (CFE) and Cue-free Express + Pedal (CFE+P) . Fourth, we report the results of two comparative listening studies using a relatively new technique for non-parametric statistical analysis called Bayes factor analysis (BFA) [62]. Last, we discuss our findings and their implications, the limitations of our work, and possible future directions for research in this area.

II. Related Work

A. Music Performance Science

A plethora of research considers how musicians make decisions during performances [56], [21], [50], [53], [54]. While many factors can affect the perceived expression of a musical performance, it is well established that musicians make their performances expressive through the "subtle continuous shaping of musical parameters such as tempo, timing, dynamics and articulation" [8, p.1], and so research into expressive performance focuses on these parameters [41], [65], [7].

In the area of music performance science (MPS), there are efforts to mathematically evaluate musical expression [12] – in particular, how performers use dynamics [42] and timing [52] – often in order to better-model timing/micro-timing deviations [58] or dynamics [60] via machine learning. Consideration of this literature led us to the initial decision to model the general tempo of playing, micro-timing of note start and end times, and dynamics (changes in loudness or velocity) in piano playing, as these are the main factors affecting the expressive qualities in a musical performance [8], [12].

[2]Henceforth, we refer to this information as OPD, short for ontimes, pitches (represented using MIDI note numbers – MNN), and durations.

B. Expressive Rendering

Expressive rendering entails the study or modelling of human-like performance parameters, based on but going beyond information contained in a musical score. Computational approaches go back several decades, with rule-based modelling systems emerging in the late 80's/early 90's [59], [18]. Table I contains a reverse-chronological, non-comprehensive but representative summary of systems used for expressive rendering in the last 20 years: we can see that rules-based models are among the earliest examples of research into expressive rendering [64], [4], [71] with artificial neural network models soon following [3], [19]. This research leverages MPS to better-inform computational modelling of human musical expression.

Table I indicates that early computer systems for expressive music performance (CSEMPs) are not reliant on the use of score cues as input, focusing mainly on data that can be extracted from notes such as ontime, pitch, and duration (OPD). Additionally, some models supplement note-level data with extra conditional information using *score annotations*. These computational annotations are often derived from musicological rulesets, and are typically created automatically at runtime [16], [28], [1], [6]. Alternatively, models may incorporate a human-in-the-loop design to achieve similar goals [32], [31].

The focus on OPD (absent score cues, sometimes including score annotations) for use in expressive performance modelling applies to most early systems such as YQX [15] and Director Musices/KTH [4], both of which are examples of CSEMPs that performed well at RENCON [39]. As the field has evolved, and we move up Table I, additional input features have been included with the aim of improving performance. For example, human-in-the-loop CSEMPs (e.g., Pop-E [31] and Mixtract [32]), allow users to annotate music for phrasal information and edit articulation curves. Later machine learning models incorporate a variety of input features for the same purpose, including: score cues (i.e., dynamic and tempo markings) [37], [15], [5]; encoded mid- and high-level features such as metrical stress patterns [6]; or automatically added score annotations based on musicological analyses [48], [43], [34].

As Table I indicates, the continued development of deep learning architectures and an increased number of input features have continued to push the boundaries. CSEMPs have seen continued improvement using, but not limited to; graph neural networks (GNNs) [38], hierarchical attention recurrent neural networks (HRNNs) with conditional variational autoencoders (CVAEs) [37], and convolution variable recurrent neural networks (CVRNNs) [20].

While the evaluation of CSEMPs has become non-standardized in recent years (see Sec. II-D), many papers include a listening study to evaluate the quality of a model's performance, and some evaluations are even comparative. In

System	Description	Inference input	Code	Evaluation
CFE and CFE+P (this paper)	NN Model with 4 or 5 transformers	OPD	Yes	Listening study, metric evaluation of distributions, self-attention mapping
Compose + Embellish [67]	Generation and performance modelling with transformer	OPD + chord progression + beat-level info	No	Listening study & metrical evaluation of distinct n-grams, beat-level info
3-layer bi-directional LSTM [57]	Three LSTM models with 128 units	beat-wise OPD + score cues	No	Pearson co-efficient vs human, schemata discussion
VirtuosoNet [37]	RNN with hierarchical attention and CVAE	OPD + score cues	∼	Listening study, case study comparison, reconstruction loss
Graph Neural Network [38]	Graph gated hierarchical attention model	OPD + score cues	Yes	Listening study, reconstruction loss
Seq2seq + VIB [23]	Seq2Seq model + recurrent variational info bottleneck	OPD + aligned audio	Yes	Listening study, mean square absolute error, KL divergence
Basis Mixer [6]	Neural network + linear models	OPD[3]+ score cues	∼	Cross validation comparison
CVRNN [20]	CVRNN with position dependent conditional inputs	OPD + piano roll around position	No	Listening study
Maximum Entropy [47]	Maximum Entropy model	Note metrical onset	No	Listening study, convergence/swing, perceptual validation
CaRo 2.0 [5]	Interactive rule set	OPD + score cues + ANN	No	RENCON
Linear basis mixer [25]	Linear regression	OPD + score cues	No	Goodness-of-fit, analysis of dynamics in recording, and predictions
Kalman Filter [28]	Switching Kalman Filter/Gaussian	Ontime + metrical onset	No	Listening study (pairwise), smoothing, improvement
YQX [16]	DBN + rules for articulation	OPD[4]	No	RENCON
ESP [27]	Hierarchical HMMs	OPD + ANN	Yes	Listening study
KTH Perf. Rules [4]	ANN	OPD + KTH rules	Yes	RENCON
SaxEx System [1]	Case-based reasoning	OPD[5]+ sound + user input[6]	No	Listening study
KTH Perf. Rules [59]	Rule-Based	OPD + KTH rules	Yes	Listening study

TABLE I
SUMMARY OF EXPRESSIVE RENDERING SYSTEMS. OPD STANDS FOR ONTIMES, PITCHES, AND DURATIONS. ANN STANDS FOR SCORE ANNOTATIONS.

particular, the graph gated HRNN architecture has outperformed the Basis Mixer and the vanilla HRNN, showing the promise of graph gated hierarchical architectures [37]. In this comparison, all models use similar input features, such as; grace notes [14], dynamics markings [25], and articulation markings [6], [26] commonly found in MusicXML files. In addition to listening studies, some research includes quantitative analysis of the proposed model and human output, to indicate the extent to which model output emulates characteristics of human output [38], [37], [23], [47].

The CVRNN model [20] is an exception to the trend of the increased utilization of score cues. Unlike the models compared in [37], the CVRNN emphasizes OPD, supplementing it by providing note-level information for all of the notes in a seven-bar window around each note. This approach offers enhanced phrasal context to inform predictions. This aligns closely with the input features of the models introduced in this paper, with our use of positional embeddings and self-attention mirroring this phrasal-contextual information.

C. Music Generation

Beyond the scope of research focused entirely on rendering expressivity onto non-expressive music data, music generation tasks have seen similar application of machine learning techniques, including but not limited to neural networks: seq2seq [45], recurrent neural networks (RNNs) [13], and Markov chains [11] in the generation of non-expressive symbolic musical output. In three non-peer-reviewed papers, the transformer neural network architecture has been used to output symbolic notes **and** expressive renderings of those notes, all in one go [36], [51], [67].[7] As such, these algorithms also represent research into expressive rendering, in scenarios where score cues are not available per se. However, the training data itself is expressive, containing expressive tempo, timing, and dynamic information. Large datasets have been used to train two of these models (e.g., hundreds of hours of piano performances transcribed from YouTube, among other data sets), and the authors acknowledge that there may be shortcomings with regards to the extent that the algorithms' outputs are original.

Theoretical and empirical work attempting to address this originality issue has found evidence to suggest that users of these algorithms may be listening to or using output consisting of large chunks of training data, often without recognizing the problem. This is because the training sets are too large for instances of copying to be identified manually (by ear) [68], [69]. As such, without checking properly for originality, we suggest that it might be overly ambitious to use a transformer neural network to generate notes and expressive

[1]This model encodes mid-high level features of the score (i.e., tonal tension) for improved inference [34].

[2]This model uses Narmour's IR model, these are basic principles of melodic perception, derived empirically by [48] to enhance their input.

[3]This model uses the generative theory of tonal music [43], Narmour's IR [48], and a proposed analysis (Jazz Theory) to enhance their input.

[4]This model supplements input with user provided qualitative values across 3 dimensions of preference; tender-aggressive, sad-joyful, calm-restless.

[7]This has also been termed "direct performance generation" in a peer-reviewed contribution focusing on LSTMs [49].

renderings of those notes all in one go.[8] We see potential, however, in using transformer neural networks to learn and predict just those aspects of expressivity, given quantized start and end times and pitches of notes as a reduced input, similar to that seen in early CSEMP research [15], [4]. The benefits of this reduced set of input features is the increased potential for use within the domains such as DAWs and game engines, where these algorithms have innovative potential. Furthermore, the transformer architecture's self-attention mechanism may be powerful enough to compensate for the smaller number of input features, by providing positional context to the models understanding of notes, and potentially surpass the capabilities of non-attention based architectures, such as RNN or LSTM.

The recent Compose & Embellish model [67] utilises two transformers based on the Compound Word model [35]. The first "Compose" model generates a lead melody based on a prompt, while the second "Embellishment" model generates an accompaniment (i.e., chords/left hand on a piano), as well as generating deviations in timing and dynamics on the note level for the generated content. While this utilisation of the transformer does tackle elements of performance modelling, it does not tackle deviations in human tempo change, nor the use of a sustain pedal, leaving it lacking in some elements of performance modelling that are addressed below.[9]

D. Evaluation

Since the discontinuation of RENCON, the evaluation of CSEMPs has become non-standardized. Of the papers listed in Table I, 10 including this paper use listening studies, and 10 use a variety of metric analyses (goodness-of-fit, loss, divergence, etc.) to assess their models. While only 3 of the models included in Table I were evaluated at RENCON [39], the contest/platform enabled models using a variety of architectures, datasets and input features to participate in widespread comparative evaluation. Notable models that performed well at RENCON include YQX [15], Director Musices [4], Pop-E [31], Mixtract [32] and CaRo 2.0 [5].

In recent years, listening studies have been widely used in this field, but they demonstrate a variety of methodologies [23], [20], [38], [67]. Each of these papers includes a listening study with between 11-40 participants, however, of these four papers, two use pairwise comparisons per piece, where listeners pick which is better [23], [38], while two compare all systems per piece, where metrics of composition and performance are rated using Likert scales [20], [67]. In addition, three of these papers use ≈ 30 sec excerpts (which [38] finds is not long enough to judge a performance). Furthermore, none of the 4 papers uses a standardized baseline for their comparison; instead opting for bespoke or adapted existing algorithms. This variety of evaluation approaches, combined with a historic absence of open-source code, makes "systematic" comparison of models difficult. We propose the adoption of the industry-standard baseline from DAWs, to establish consistency across future listening experiments.

[8]In models that rely on tokenisation, expressive data (e.g., MIDI events where timing and velocity values are still expressive) lead to a larger dictionary than inexpressive data (where the note ons/offs can be represented by fewer tokens, and velocities would not be encoded at all). If two training sets contain the same pieces, but one has expressive versions and the other inexpressive versions, then a model trained on the expressive set is more likely to have issues with copying chunks of training data than one trained on the inexpressive set, since the variety in dependencies between local tokens is lower for the scenario with the larger dictionary.

[9]This system's code is not open source at the time of conducting these experiments.

III. Dataset and Learning Aims

The dataset used for this paper is the Aligned Scores and Performances (ASAP) dataset [17], which consists of 236 musical scores and 1,067 performances of these scores. All of the pieces are Western classical music from 15 different composers and all are written for and played on piano. Each performance in the ASAP dataset has an associated annotation file, which we use to calculate an **performance-symbolic beat map** (PSBM) between the beat `ontime` in the symbolic representation of the piece and the beat `onset` in seconds in the performance. The original beat annotations for the ASAP dataset were created automatically [explained in 17].

Our overall data processing approach and terminology is similar to other work in the field [6], [37], [64]. From the MIDI performance of a piece, we derive the `onset` in seconds that each note begins, the duration in seconds for which it is held (that we call `durSec`), and the `velocity` (dynamic level) of each note. As indicated in Figure 1, for any score note (shaded note in top panel), we can use the PSBM to *estimate* where that note will begin and end in the performance (empty note in bottom panel), and compare these to the same properties *observed* in the performance (shaded note in bottom panel), deriving the quantities d' and d''. Using the PSBM once more, we can define the start- and end-time differences in units of quarter-note beats (empty note in top panel), labeled x' and x'', generally, referred to as `ontimeAdj` and `offtimeAdj`.

Based on existing MPS and observations of our first listening study, we developed models aiming to predict dynamics, tempo, timing, and sustain pedal:

- **Dynamics.** Given OPD note data, predict the `velocity` value of each note;
- **Tempo.** Given OPD note data and one existing tempo in BPM, predict the beat-wise tempo values up to and including the end of the note data;
- **Timing.** Given OPD note data, predicted velocity data and associated beat-wise tempo values, predict the `ontimeAdj` and `offtimeAdj` values of each note;
- **Sustain pedal.** Given OPD note data, predict beat-wise pedal values (our CFE+P model only).

IV. Method

A. Data preparation

Before training any models, we first determine for each performed note in ASAP whether there was a score note of the same pitch within 0.1 sec of where we would expect it to occur, given the PSBM. We calculate an F_1 synchronization score for each MIDI performance and only admit

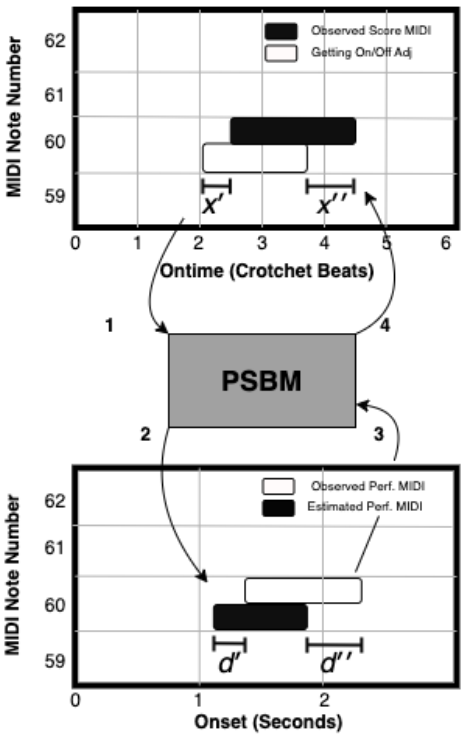

Fig. 1. 1. A note's `ontime` is extracted from the score; 2. This is combined with the PSBM to estimate the `onset` of the corresponding note in a performance; 3. This estimated `onset` is compared to the actual time at which the note is performed; 4. The difference is combined with the PSBM again to give `ontimeAdj`/`offtimeAdj`. The values indicated in the figure are exaggerated for sake of readability.

a performance into our dataset if its F_1-score exceeds .9.[10] Additionally, we applied a further filtering of notes if either the `ontimeAdj` or `offtimeAdj` values were outside of the range $(-0.7, 0.7)$ sec. This gives us 247,247 notes for training our models, and we adhere to a train:test:validate split of 80:10:10.[11]

B. Architecture

The architecture for our initial CFE model consists of an ensemble of four transformer models [63], [36] (one for velocity, one for tempo fluctuations, one for note start-time adjustment, and one for note end-time adjustment). This initial model is then augmented for our second listening study, to include a fifth transformer for predicting sustain pedal usage (also indicated in Figure 2). We refer to this second model as CFE+P.

In this architecture, we use the transformer encoder only, as the predictive task is not auto-regressive, and we directly input the conditioning features and predict the target feature, meaning there is no application of cross-attention or causal masking. The right-hand side of Figure 2 depicts our architecture in more detail: we have an initial embedding layer for pitch, which converts the discretized pitches into learnable embeddings. Subsequently, we use the remaining input features (e.g., ontime and duration) to provide positional information to the model, where the value or position is not discrete. For the ontimeAdj and offtimeAdj models, we also provide the predicted velocity values as further positional information. These embeddings are then converted into a vector of the same size of the initial pitch embedding, and the summed information is passed through the transformer encoder, which captures the dependencies and relationships between elements in the sequence. The model's output y_t is a certain predicted attribute (e.g., velocity) of the note at time t, based on the given attributes x_t (e.g., OPD).

The transformer architecture is chosen because the self-attention mechanism, in combination with positional context, gives the model the potential to understand the relationship between tokens, within the max sequence length parameter of the model. Our premise is that with our architecture's focus on using only OPD as input, our premise is that self-attention may help it to overcome the lack of additional input features, such as score cues (e.g., crescendo marks).

First in the inference pipeline,[12] `OPD` are used as input for the *Dynamics* model. Using these features, the model predicts the velocity of each note in the sequence, where the predicted attribute is an integer value of 0-31, with each value representing four values in the range of 0-127 typically used for velocity.

Second, the *Timing* model is comprized of two components: one that predicts adjustments to the note's quantized ontime (`ontimeAdj`); and another that predicts adjustments to note's quantized offtime (`offtimeAdj`). This model takes the `OPD` and predicted `velocity` values as input, and predicts an ontimeAdj and offtimeAdj for each note in the sequence. In these cases, the predicted attribute is a float representing a micro-adjustment to quantized-start or end-time of a note, respectively, in units of quarter notes.

Thirdly, the *Tempo* model analyses the `OPD` information for all the notes in the piece, and calculates a set of features per beat: `note density`, `min ontime`, `max ontime`, `mean ontime`, `min MNN`, `max MNN`, `mean MNN`, `min duration`, `max duration`, `mean duration`. Using these input features, it predicts the proportional change in tempo from one beat to the next. We label this $\alpha\Delta t$, where $\alpha = 0.25$ is a weighting affecting the amount we allow tempo to change proportionally from one beat to the next.

With the information provided, we then calculate the new

[10]For each note in the score MIDI, we use the annotated beat locations to derive an estimate of where that note should occur in a performance of the piece. On searching the performance, if we find a note of the same pitch within 0.1 sec of our estimate, then this counts as a true-positive (TP) result (when there are multiple candidates, the one with the smallest absolute time difference between estimated and performed time is selected); otherwise (no note found in the performance) it counts as a false-negative (FN); if, once the score MIDI has been iterated over, a note from the performance MIDI remains unmatched, this counts as a false-positive (FP). The F_1 synchronisation score is then calculated in the ordinary way: $F_1 = 2PR/(P+R)$, where $P = \mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$ and $R = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$.

[11]We note that the number of successfully synchronized notes is fewer than the total number of notes in the ASAP dataset. We suspect that some MIDI encodings of performances having non-standard tempos may be responsible for synchronisation failures, but we are satisfied with the amount of data available, and leave further investigation of this issue to future work.

[12]All of the models in this architecture make probabilistic predictions, using categorical sampling at a temperature of 0.4.

expressive start time x according to Eq. 1 and expressive duration z according to Eq. 2, where O is the expected start time and D is the expected duration according to the tempo fluctuations, x' and x'' are `ontimeAdj` and `offtimeAdj`, respectively, and $\beta = 0.25$ is a weighting affecting the amount we allow timing variations to shift notes either side of the underlying beat. The weights α, β were chosen via varying these parameters and listening to the resulting outputs.

$$x = O + \beta x' \tag{1}$$

$$z = O + D + \beta x'' - x \tag{2}$$

For CFE+P, the last step involves using the same beat-level input features employed in tempo modelling to predict the given position of the sustain pedal at the time of each beat. For this model, the predicted attribute is an integer within the range of 0-3, where 0 denotes complete depression of the pedal, and 1-3 represent varying degrees of the pedal being raised. We do not model sustain pedal use in this project initially, for the CFE architecture, because we do not want our modelling efforts and results to be too confined to one specific instrumental category (e.g., keyboard instruments with a sustain pedal). The downside of this decision, however, is that the software we use to turn expressive MIDI data into audio accesses a qualitatively different sample bank when the sustain pedal is down, and this may lead to slightly more realistic or favorable-sounding excerpts for the VirtuosoNet and human categories. As such, we introduce the modelling of the sustain pedal in Listening Study 2, which comes with a reduction in instrument agnosticism.

C. Training

We train all of the models in a supervised fashion, aiming to minimize the loss of the predictions [38], [57]. The three note-level transformers have a model dimension = 256, number of heads = 4, number of layers = 6, batch size = 32, sequence length = 255, while the beat-level transformers have a model dimension = 64, number of heads = 4, number of layers = 3, batch size = 12, sequence length = 255. All are trained with the Adam optimiser [40] with a learning rate = 0.0001. Loss scores are tracked during training and the final scores are shown in Table II. We evaluate the note-wise models using L^2 loss (mean squared error, MSE), because it is appropriate for the loss ratings to be scaled depending on the distance between the prediction and target, rather than reporting the prediction as equally incorrect for all values except the target. We evaluate the beat-wise models using L^1 (mean absolute error, MAE) and Cross Entropy (CE) loss, respectively. We do this as we feel that deviations in tempo do not need to be penalized as strongly as those in note-wise models, and pedal prediction is a classification task, meaning CE loss is appropriate.

D. Model Evaluation

We employ Bayes factor analysis (BFA) to analyse our quantitative listening study results. BFA is superior to the frequentist hypothesis testing framework (e.g., the F-test behind an ANOVA, and t-tests) because it allows for the possibility of finding a meaningful lack of difference between two systems being compared; in frequentist hypothesis testing, one can only reject the null hypothesis (abbreviated H_0) in favor of the alternative hypothesis (H_1), or fail to reject H_0 in favor of H_1 – the latter being subtly but importantly different from finding evidence in favor of H_0 [62]. The Bayesian hypothesis testing framework is being adopted gradually in the comparative evaluation of systems [70], [2], with this manuscript representing a relatively novel example of its use.

Model	Loss Type	Loss/Error
Velocity	MSE	0.650
OntimeAdj	MSE	1.200
OfftimeAdj	MSE	0.098
Tempo	MAE	0.029
Pedal	CE	0.441

TABLE II
FINAL LOSS RATINGS FOR EACH MODEL DURING TRAINING.

In conducting a non-parametric BFA, 20 simulations are run for each system comparison, each providing a BF_{10}-value (a counterpart to the traditional p-value). The mean of these 20 results is reported for each hypothesis for Listening Study 1 (Sec. V-A3) and 2 (Sec. V-B3).

Table III shows how these values are typically interpreted in terms of strength of evidence for H_0 or H_1. These coefficient thresholds are somewhat arbitrary, similar to the α-value in frequentist statistics, but are based on those used in previous work [70]. The BF_{10} score is a likelihood ratio of the marginal likelihood of H_0 and H_1, given the following equation, where θ_0 denotes the parameter of interest:

$$\mathrm{BF}_{10} = \Pr(\theta_0|H_0) \,/\, \Pr(\theta_0|\mathrm{data}, H_1) \tag{3}$$

BF_{10}	Interpreting
> 100	Extreme evidence for H_1
30 – 100	Very strong evidence for H_1
10 – 30	Strong evidence for H_1
3 – 10	Moderate evidence for H_1
1 – 3	Anecdotal evidence for H_1
1	No evidence
1 – 0.333	Anecdotal evidence for H_0
0.333 – 0.1	Moderate evidence for H_0
0.1 – 0.033	Strong evidence for H_0
0.033 – 0.01	Very strong evidence for H_0
< 0.01	Extreme evidence for H_0

TABLE III
BAYES FACTOR INTERPRETATION [70]

V. EXPERIMENTS

We assess our model's (CFE and CFE+P) through a comparative evaluation of expressive renderings of music data, taking place over two listening experiments. The first study includes data for nine participants, and the second expanded version of the first study includes data for 13 participants. The experimental design, hypotheses, and results of each study are detailed below.

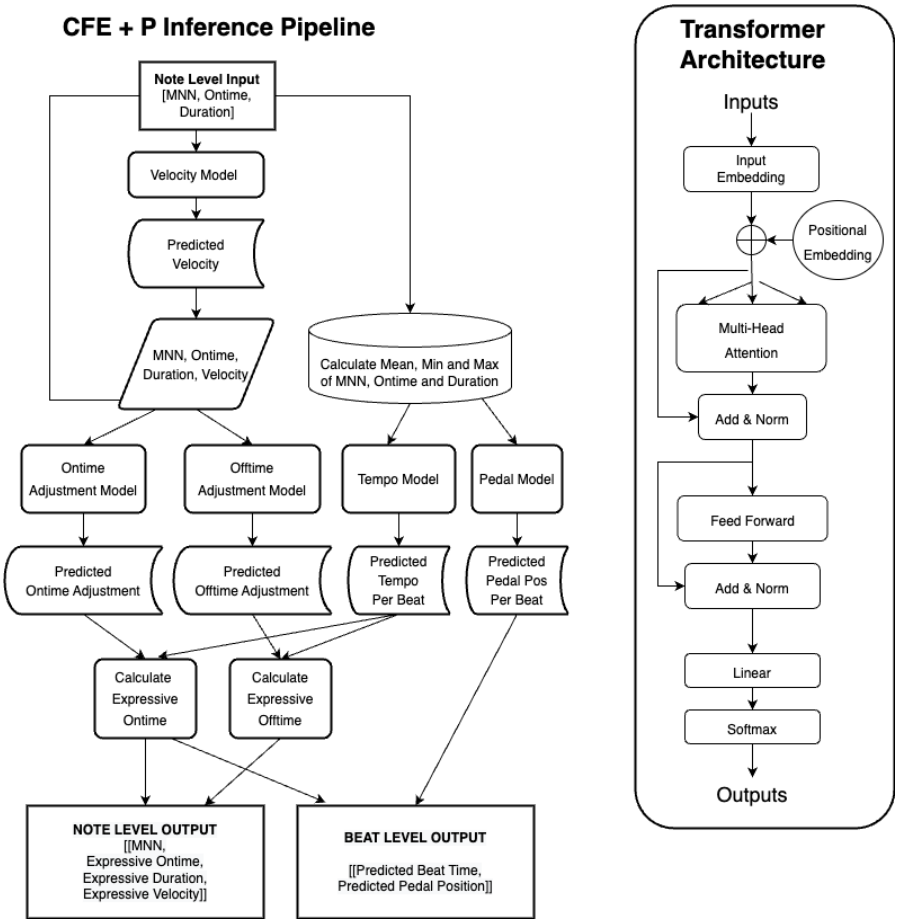

Fig. 2. The inference pipeline of Cue-free Express (CFE) and Cue-free Express + Pedal (CFE+P) models (left); and a diagram after indicating the transformer architecture used (right).

A. Listening Study 1

1) Experimental Design: Two computational expressive rendering algorithms (VirtuosoNet [37], our CFE model) and two comparison points (Human Performer and Inexpressive MIDI) are evaluated using a within-participants design involving nine pieces of music, giving rise to 36 stimuli.

Participants are undergraduate or postgraduate music students at the University of York and the study received ethical approval from from York Computer Science Ethics Board. Nineteen participants completed the study and received £15 in compensation for one hour of their time. Four participants' data are removed for completing the questions in less time than it took to listen to all the music excerpts, and six for rating *completely inexpressive* output as highly expressive (rating of four or higher on a Likert scale of 1-7) two or more times, leaving nine participants' data for analysis.

The participants rate excerpts along the following dimensions: overall expressiveness of the performance; use of dynamics; use of timing and articulation; use of tempo change/rubato. A free-text box is also provided per piece for participants to record their thoughts and reactions. Participants are not aware of the provenance of any excerpt with respect to underlying system. Furthermore, it is explained to participants that the lowest ratings should be reserved for excerpts that sound inexpressive or robotic, and the highest ratings reserved for excerpts that sound expressive or human-like.

The nine excerpts used as stimuli for this study consist of three Bach Fugues, two Bach Preludes, one Haydn Keyboard Sonata, one Chopin Ballade, and two excerpts from a Schubert Impromptu.[13] These pieces represent a range of musical periods (Baroque, Classical, Romantic) and time signatures (9-8, 4-4, 3-4, 6-8). This piece selection and the test:train:validate split happens pseudo-randomly to keep the tests fair, using a seed for replicability. In keeping with previous comparative evaluations of expressive rendering systems (e.g., [67], [20]), we take matching sections from the human performance data and the MIDI scores, and give VirtuosoNet and CFE the necessary input (see Table I) to render the same selections.

All of the MIDI is processed using the same Steinway Piano preset in GarageBand, and is faded in and out using the same amounts of time within Audacity to ensure consistency. Based on the existing literature, we opt for all excerpts to be 20-60 sec long [due to feedback that 10-15 sec is too short a time to assess music, 46].

2) Hypotheses: As we use a BFA for our data analysis, we are able to state hypotheses about finding evidence in favor

[13]The opus numbers of the test data are as follows: Bach Fugues BVW862, 885 & 893; Bach Preludes BVW865 & 892; Chopin Ballade D47 no.3 mvt.1; Haydn Keyboard Sonata Hob XVI nr.32 mvt.1; Schubert Impromptu D899 no.2. mvt.2

of the null hypothesis (i.e., no difference between systems), unlike in frequentist hypothesis testing. Our hypotheses are as follows:

1) CFE will receive significantly higher ratings on expressivity that the Inexpressive baseline;
2) Human Performer will receive significantly higher ratings on tempo/rubato than VirtuosoNet;
3) Human Performer will receive significantly higher ratings on tempo/rubato than CFE;
4) Human Performer will receive significantly higher ratings on expressivity than VirtuosoNet;
5) Human Performer will receive significantly higher ratings on expressivity than CFE;

3) Results: Our results can be seen in Figure 3, which indicate that our model, CFE, was rated higher than the inexpressive baseline MIDI system. This finding is confirmed by the statistical analyses reported below. It is also worth noting that across our approximately 30-sec excerpts, VirtuosoNet is evaluated on par with the human performers, being almost indistinguishable across all four metrics of expressivity, dynamics, timing and tempo. The outcomes of the BFA are as follows:

1) Extreme evidence ($\mathrm{BF}_{10} > 100$) that CFE receives significantly higher ratings on expressivity than the expressionless baseline;
2) Strong evidence ($\mathrm{BF}_{10} = 0.0565$) that Human Performer did not receive significantly higher ratings on expressive tempo than VirtuosoNet;
3) Extreme evidence ($\mathrm{BF}_{10} > 100$) that Human Performer receives significantly higher ratings on expressive tempo than CFE;
4) Strong evidence ($\mathrm{BF}_{10} = 0.0624$) that Human Performer did not receive significantly higher ratings on expressivity than VirtuosoNet;
5) Extreme evidence ($\mathrm{BF}_{10} > 100$) that Human Performer receives significantly higher ratings on expressivity than CFE;

4) Listening Study 1 Discussion: The main findings of Listening Study 1 are that: our model (CFE) performs better than the *inexpressive* baseline; and VirtuosoNet performs well compared to Human Performer and CFE. Interesting qualitative responses highlight some thoughts on: how participants engage with listening studies based on their own performance experience; the nature (and emphasis) of fugue subject during performances; the use of pedal in performances being a potential detriment (even though ratings indicate otherwise in this case).

- "my opinion [of each audio stimuli] would have differed had I had experience playing [the pieces the excerpts came from]...I judged the ones I have played much more critically – probably because I knew what the score demanded";
- "all [system's renditions of a single piece] seem to think the fugue's subject has to [be] emphasized and be dynamic";
- "[The final performance of Schubert's D899 no.2] is rather muffled with too much pedal".

B. Listening Study 2

Based on feedback received concerning Listening Study 1, we decide to perform a second listening study which includes two new systems. These systems are: the Basis Mixer [BM, 6], another score cue informed algorithm; and a Randomized MIDI (RM) system (described more fully below). RM is intended to offer a stronger baseline than *Inexpressive*, while BM offers another computational model for comparison.

Furthermore, based on observations of VirtuosoNet's performance in Listening Study 1, we train a fifth transformer model to generate sustain pedal predictions. We do this as, the use of sustain pedal automation in GarageBand appears to access qualitatively different samples on the Steinway Piano preset, and VirtuosoNet's incorporation of sustain pedal predictions could be driving a larger difference in ratings between CFE and VirtuosoNet in Listening Study 1 than would otherwise be the case. As such, we include CFE+P in Listening Study 2, which enables us to shed some light on this matter.

For Listening Study 2, the same musical excerpts are used as in Listening Study 1, so the examples for CFE, VirtuosoNet, Human Performer and *Inexpressive* remain the same. We prepare additional audio stimuli for the three new systems: CFE+P, RM, and BM. The BM system uses the publicly available implementation that can be found "in the wild",[14] which is trained on the Vienna 4x22 data set. This data set is smaller than the full ASAP data set, but of a comparable size to the amount of data that we use for our models based on synchronisation F_1 scores. The RM system is based on code we wrote to simulate Logic Pro X's "humanise function" (referred to technically as MIDI Transform presets). This is the industry standard, non-machine learning solution that composers use when working in Logic Pro X, whereby random, uniformly distributed numbers in a specifiable range are added to or subtracted from the velocity, ontime, and duration of inexpressive note data, to replicate what Logic Pro X users would expect from "humanising" their MIDI.[15] In this case, the natural, scientific approach would be to parameterise the RM system using the ranges observed in CFE's output, but an RM system parameterized in this way sounds too chaotic, so we reduce the ranges RM used until the output sounds less chaotic but not too metronomic or inexpressive. While Cubase's logical editor provides an alternative, potentially

[14] https://github.com/CPJKU/basismixer

[15] We read the documentation at https://support.apple.com/guide/logicpro/midi-transform-window-presets-lgcp215831be/mac and studied distributions derived from applying the MIDI Transform presets to toy examples, in order to arrive at our findings and simulation.

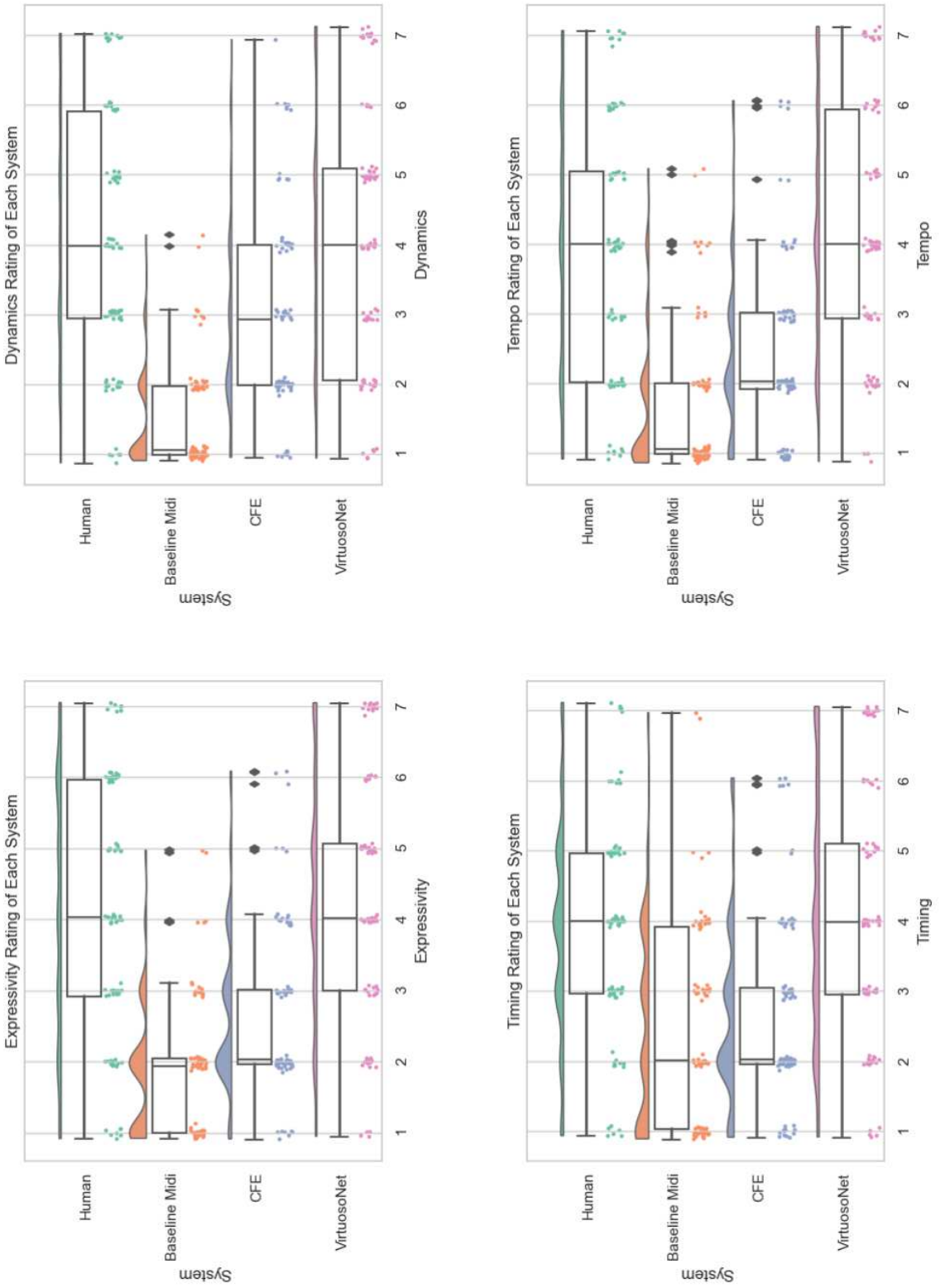

Fig. 3. Raincloud plots of our results for Listening Study 1; expressivity, dynamics, timing and tempo ratings.Jittering of data means that there may be slight discrepencies between scores of the same value, i.e. if they sit around the same number but on either side.

superior baseline for potential comparison, we choose to use "humanise" as the second baseline because: our model purposely requires only inputs seen in MIDI data and is designed to replace this function in particular; the use of the logical editor requires direct input from users which for the experiment would be different depending on who is conducting the experiment; the use of the logical editor is analogous to how score cues and markings are potential impediments to creative workflow when using a DAW, even if they may increase the expressive quality of MIDI playback.

1) Experimental Design: We use the same experimental design and recruitment methods as the previous study, and the same exclusion criteria, but scaled up to allow for participants listening to almost twice as many excerpts. For example, in Listening Study 2, a participant's data are only excluded if they rated **four or more** *Inexpressive* stimuli higher than 4 for expressivity. This exclusion criteria meant that we excluded data for seven of our 20 participants, leaving us with the data of 13 participants for analysis.

2) Hypotheses: For Listening Study 2, our hypotheses are as follows:

1) CFE will receive significantly higher ratings in expressivity than RM;
2) CFE+P will receive significantly higher ratings in expressivity than CFE;
3) CFE will receive significantly higher ratings in dynamics than RM;

3) Results: Our results can be seen in Figure 4, with the BFA results as follows:

1) Anecdotal evidence ($\mathrm{BF}_{10} = 0.267$) that CFE did not receive significantly higher ratings in expressivity than RM;
2) Moderate evidence ($\mathrm{BF}_{10} = 4.160$) that CFE+P receives significantly higher ratings in expressivity than CFE;
3) Strong evidence ($\mathrm{BF}_{10} = 0.0956$) that CFE did not receive significantly higher ratings in dynamics than RM;

Based on the results above, we decide to conduct a post-hoc analysis to determine if our CFE+P model rates significantly higher than RM (since CFE did not). The results of this post-hoc analysis show strong evidence ($\mathrm{BF}_{10} = 30.0$) that *CFE+P receives significantly higher ratings for expressivity than RM*.

C. Listening Study 2 Discussion

The results of this study indicate that our model that includes sustain pedal prediction (CFE+P) rates higher than BM [6], CFE (our non-pedal using model), and both the RM and *Inexpressive* baselines for expressivity, dynamics and tempo. RM is evaluated better than BM and *Inexpressive* for expressivity and dynamics, but performs equally to BM and *Inexpressive* for tempo. Furthermore, RM performs better than *Inexpressive*, and equally to BM, CFE and CFE+P for timing. The timing result is unexpected, considering the random distribution of expressive values used by this system. Additionally, as with Listening Study 1, VirtuosoNet [37] is evaluated almost on par with Human Performer across our 30-sec excerpts, the difference being slightly more distinguishable than found previously (see Figure 3).

Qualitative data from free-text boxes in Listening Study 2 reveal: a participant senses that RM's distribution of velocity values is random and arbitrary compared to a human; longer excerpt lengths can affect the ratings positively; pedal use could be masking perception, making it harder to discern expressive intent from timing mistakes.

- "Some of the [RM] performances have wide dynamic ranges, but are not necessarily expressive as a result of this because the louder/softer notes are clearly arbitrary";
- "I felt as though some of these extracts were too short to get a true measure of their expression – had these gone on for longer, I would probably have considered some of them to be more expressive because there would have been more room for contrast";
- "The midi [*Inexpressive* system] is far too perfect with it's timing, I hate it";
- "It is quite hard to tell which is good because a lot of them cover their mistakes with the pedal".

VI. General Discussion

How expert human performers imbue music with expression, and how computational models of expressive performance emulate this process, are topics of interest to the academic fields of music performance science (MPS), music-cognitive science, and music informatics research, as well as in multiple application domains, such as musical co-creation with AI, and game audio. While research into the creation of expressive rendering algorithms is ongoing, unlike music generation research, there has been no commercial application of these algorithms in professional music making software. Furthermore, while these algorithms continue to yield improving results, the industry standard, non-machine learning solution that is available in DAWs has not been adopted in research to serve as a basis of comparison. This lack of a standardized baseline makes comparative evaluation between experiments complicated, and this issue is compounded by a lack of systematic evaluation within the field (since the discontinuation of RENCON). Further complicating the problem is a lack of available open-source models "in the wild." Understandably, these factors make it difficult to conduct fair comparison between models, in experimental settings with high internal validity. This is especially true when considering that many models utilize wildly different data sets, architectures, and require varying input features.

Not only is there an academic-scientific imperative to establish a more ecologically valid methodology for comparison between expressive rendering algorithms in future work, but

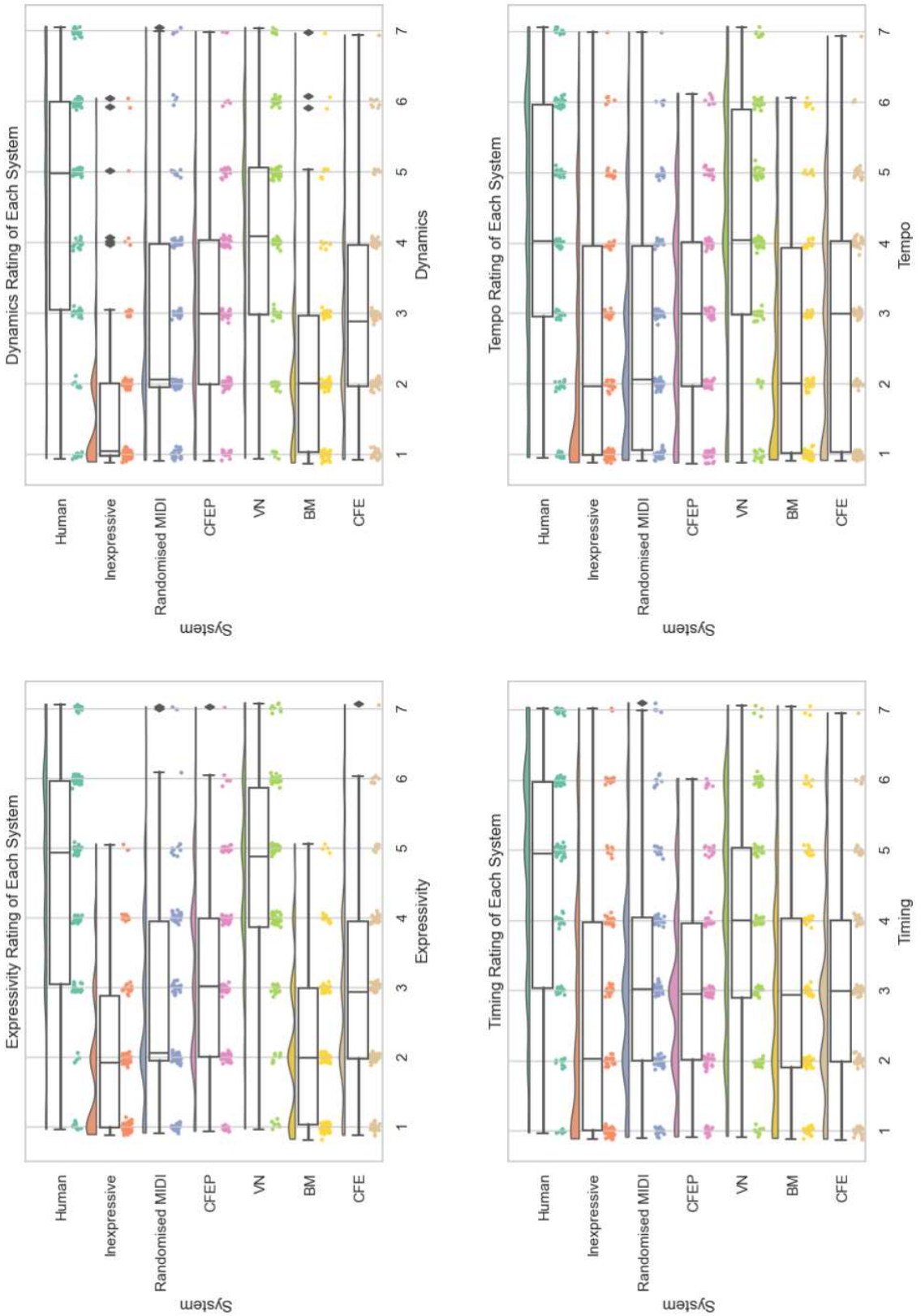

Fig. 4. Raincloud plots of our results for Listening Study 2; expressivity, dynamics, timing and tempo ratings. The scores are integers between 1 and 7. Jittering of data means that there may be slight discrepencies between scores of the same value, i.e. if they sit around the same number but on either side.

to consider the many potential applications for expressive rendering algorithms, which could innovate and support creative workflows. OPD-focused models by requiring minimal input features, can be applied in situations where a composer is working with piano roll in a DAW, and intends the DAW export to represent some or all of the final expressive musical artifact. The application of these algorithms can support creative workflows, as it can be cumbersome and an impediment to have to draw in tempo, timing, and velocity information for each bar/note [55], or where a tool like Cubase's logical editor requires the devising of rulesets by the user.

With such motivation in mind, the first algorithm we present in this paper is called Cue-Free Express (CFE). This model consists of an ensemble of transformer neural networks focused on tempo, timing, and dynamics, and requires only minimal input that can be found in MIDI data for the maximum breadth of application. We pose the question of whether this cue-free model can outperform an inexpressive baseline (see Listening Study 1). We find that CFE does receive significantly higher ratings for expressivity compared to the inexpressive baseline, but it is short of the cue-based system VirtusosoNet [37] and also the professional human performers. We use non-parametric Bayes factor analyses to test our hypotheses [62], which is an improvement upon frequentist hypothesis testing (allowing for testing of meaningful non-differences between systems), and could be of benefit in future research. We then augment our initial architecture with a fifth model that predicts sustain pedal use (CFE+P), and pose the questions: can either of our models outperform the industry standard, non-machine learning baseline; and how do these OPD-focused models compare to existing neural network models [6], [37]? In Listening Study 2, we find that while our original CFE model only outperforms the inexpressive baseline, that CFE+P outperforms the industry standard, the inexpressive baseline, our initial CFE model, and performs on par with the cue-informed Basis Mixer as it is found "in the wild". This improvement is not enough to close the gap between our efforts and VirtuosoNet/human performers, but provides valuable insight into the advantage of including pedal predictions in expressive rendering pipelines. We also provide evidence to demonstrate the potential advantages of architectures leveraging self-attention for modelling expressive performance tasks.

Thus we re-initiate the interesting and application-relevant challenge of generating expressive renderings *using only the pitch and quantized start and end-times of notes as input for prediction* and compare our systems to those models that are available for comparison "in the wild," and the industry standard baseline, and finish by outlining some limitations and ideas that for future work in this domain.

A. Limitations

A limitation of this research is that there are three reasons that the Basis Mixer may not have scored higher than CFE+P: a lack of pedal predictions in the Basis Mixer (as it scored similalry to CFE); the MusicXML files used for these studies may contain fewer cues that are necessary for better prediction of basis functions; the Vienna 4x22 dataset may not be large enough to compete with our selection from the ASAP dataset (even though they are approximately equal in size once we filter for highly synchronized data).

Another limitation is that we do not manage to synchronise the entire quantized-expressive representations present in the ASAP dataset. The dataset contains separate quantized (e.g., MusicXML and MIDI from score) and expressive (e.g., MIDI from human performer) representations, and then (semi-automatically-calculated) beat annotation files that should make it possible to bridge or synchronise the two. We find, however, that some of these annotation files lead to predictions of performed notes given the score notes that do not exist within a reasonable time threshold, and so we work only with the subset of pieces for which synchronization F_1 values are above .9. Addressing this limitation may lead to future modelling improvements, because we would be able to work with a larger training dataset. This also means that while a broad range of periods, keys, and time signatures are used for musical excerpts in our study, all the pieces do have quite similar tempi (allegro). This means we cannot guarantee that our findings will generalize well to more extreme tempi.

B. Future work

In conducting this research, we made initial explorations in using Gaussian-uniform transformations during inference, to increase the frequency with which our transformer models predict tokens from the tails of distributions – to "take more risks" or utilise stronger contrasts with regards to expressive rendering decisions. This was motivated by how human performers use (and listeners perceive) contrast, and take risks such as extreme changes in dynamics or tempo change in their performances, as opposed to making changes that can be quantified as being drawn from near to a distribution's mean.

As an example of "risk taking" and a final, more high-level thought, we return to the Bach fugue (BVW 885) excerpt from our listening studies. The fugue subject is present in the top or soprano part at the beginning of the excerpt, and again in the tenor or lower part towards the end of the excerpt. The human performer emphasizes its appearance with dynamic values that are greater than those of contemporaneous, higher-pitches notes – which could be said to go against a general observation of expressive performance that the highest notes in a texture should be played louder, typically because this is where the "melody" resides.[16] This is a relatively mild risk for a human performer to take – to go against the general observation or rule of expressive performance and instead adhere to a stylistically appropriate rule that the appearance of a thematically important element or repetition occurrence ought to be highlighted, even if it occurs in a lower voice. Considering the dynamic values of the VirtuosoNet output, however, we see that the occurrence is quieter than the contemporaneous higher notes.[17] VirtuosoNet appears to have learnt the general rule (top-of-texture notes should be played loudly) but presumably not the rule that would be more stylistically

[16]This is Human 01_02 in the Supplementary Audio.
[17]This is VirtuosoNet 05_02 in the Supplementary Audio.

appropriate in this scenario (emphasise thematically important note collections). Our models also adhere to this same generalized rule. Existing work on pattern discovery is far enough advanced that repetitive elements such as fugue subjects can be extracted automatically from a musical score [9], [10], [24]. As such, future work may consider how such extracted information could be integrated with a neural net approach, in order to further advance work on expressive rendering.

C. Conclusion

In conclusion, in this paper we introduce the first transformer-based CSEMP models, and we evidence that an ensemble of transformers utilising self-attention and positional embeddings for tokens in a sequence can be used to expressively render musical performance onto inexpressive MIDI data, *using only the pitch and quantized start and end times of notes as input for prediction*. Our results demonstrate that this model can outperform both the industry standard and inexpressive baselines, while performing on par with an existing FFNN model. Additionally, we demonstrate that with small edits to the pedal predictions, an existing model (VirtuosoNet) can perform on par with human performances over 30-sec excerpts. Furthermore, we propose, that future research use a model based off the "Humanize" function from Logic Pro X as the industry standard to set a precedent for fair and valid comparison across experiments.

ACKNOWLEDGMENT

This work was supported by the EPSRC Centre for Doctoral Training in Intelligent Games & Games Intelligence (IGGI) [EP/S022325/1]. The authors would also like to thank Dasaem Jeong for expressively rendering the MIDI from their system VirtuosoNet for use in this research.

VII. ETHICS AND CONSENT

All research conducted for this project was approved by the University of York Computer Science Ethics Committee. All participants gave informed consent.

REFERENCES

[1] Arcos, J. & De Mántaras, R. "An Interactive Case-Based Reasoning Approach for Generating Expressive Music," *Applied Intelligence*., vol. 14, pp.115–129, 2001.

[2] Barahona-Ríos, A. & Collins, T, "SpecSinGAN: Sound effect variation synthesis using single-image GANs," in *Proc. Sound & Mus. Comput. Conf.*, 2022, pp. 302–309.

[3] Bresin, R, "Virtual virtuosity: studies in automatic music performance," Ph.D. dissertation, Dept. Speech, Music & Hearing. KTH, Stockholm, Sweden, 2000.

[4] Bresin, R., Friberg, A. & Sundberg, J, "Director Musices: The KTH Performance Rules System," *Mus. Inf. Proc. Soc. of Jap.,* vol. 46, pp.43–48, 2002.

[5] Canazza, S., De Poli, G. & Rodà, A, "Caro 2.0: an interactive system for expressive music rendering," *Advances In Human-Computer Interaction*., vol. 2, no. 2, Feb. 2015.

[6] Cancino-Chacón, C, "Computational Modeling of Expressive Music Performance with Linear and Non-linear Basis Function Models," Ph.D dissertation, Dept. Computational Perception. JKU, Linz, 2018.

[7] Cancino-Chacón, C., Peter, S., Chowdhury, S., Aljanaki, A. & Widmer, G, "On the Characterization of Expressive Performance in Classical Music: First Results of the Con Espressione Game," in *Proc. 21st Conf. for the Int. Soc. for Mus. Inf. Retr.*, 2020, pp. 613–620.

[8] Chowdhury, S. & Widmer, G, "Towards Explaining Expressive Qualities in Piano Recordings: Transfer of Explanatory Features Via Acoustic Domain Adaptation," *IEEE Int. Conf. On Acous., Speech And Sig. Proc. (ICASSP)*. pp. 561-565, 2021.

[9] Collins, T., Arzt, A., Flossmann, S. & Widmer, G, "SIARCT-CFP: Improving Precision and the Discovery of Inexact Musical Patterns in Point-Set Representations," in *Proc. 14th Conf. for the Int. Soc. for Mus. Inf. Retr.*, pp. 549–554, 2013.

[10] Collins, T., Arzt, A., Frostel, H. & Widmer, G, "Using geometric symbolic fingerprinting to discover distinctive patterns in polyphonic music corpora," in *Computational Music Analysis*. Cham, Switzerland.: Springer, Cham, 2016, ch. 7, pp. 445–474.

[11] Collins, T. & Laney, R, "Computer-Generated Stylistic Compositions with Long-Term Repetitive and Phrasal Structure," *Journal Of Creative Music Systems*., vol. 1, no. 2, 2017.

[12] Doherty, E, "Measuring Expressive Music Performances: a Performance Science Model using Symbolic Approximation," Ph.D dissertation, Cons. of Music and Drama. Tech. Uni. Dublin, Ireland, 2019.

[13] Dua, M., Yadav, R., Mamgai, D. & Brodiya, S, "An Improved RNN-LSTM based Novel Approach for Sheet Music Generation," *Procedia Computer Science*., vol. 171, pp. 465–474, 2020.

[14] Flossmann, S., Grachten, M. & Widmer, G, "Experimentally investigating the use of score features for computational models of expressive timing," in *Proc 10th Int. Conf. On Music Perc. & Cogn.*, 2008, pp. 1–6.

[15] Flossmann, S., Grachten, M. & Widmer, G, "Expressive Performance Rendering: Introducing Performance Context," in *Proc. 6th Sound & Music Comp. Conf.*, 2009, pp. 155–160.

[16] Flossman, S., Grachten, M. & Widmer, G, "Expressive Performance Rendering with Probabilistic Models," in *Guide To Computing For Expressive Music Performance,* A.Kirke, E.R. Miranda, Ed., London, UK: Springer, 2012, ch. 3, pp. 75-98.

[17] Foscarin, F., Mcleod, A., Rigaux, P., Jacquemard, F. & Sakai, M, "ASAP: a dataset of aligned scores and performances for piano transcription," in *Proc. 21st Int. Conf. For Music Inf. Ret.*, 2020, pp. 534–541.

[18] Friberg, A, "A Quantitative Rule System for Musical Expression," Ph.D dissertation, Dept Speech, Music & Hearing. KTH, Stockholm, Sweden, 1995.

[19] Friberg, A., Bresin, R. & Sundberg, J, "Overview of the KTH rule system for musical performance," *Advances*

In Cognitive Psychology., vol. 2, pp. 145-161, Jun. 2006.

[20] Maezawa, A., Yamamoto, K. & Fujishima, T, "Rendering music performance with interpretation variations using conditional variational RNN," in *Proc. 19th Conf. for the Int. Soc. for Mus. Inf. Retr.*. pp. 855-861, 2018.

[21] Gabrielsson, A, "Once again: The theme from Mozart's piano sonata in A major (K.331)," *Action And Perception In Rhythm And Music,* pp. 81-104, 1987.

[22] Gabrielsson, A, "Music Performance Research at the Millennium," *Psychology Of Music*., vol. 31, pp. 221–272, Jun. 2003.

[23] Gillick, J., Roberts, A., Engel, J., Eck, D. & Bamman, D, "Learning to Groove with Inverse Sequence Transformations," in *Proc. Of The Inter. Conf. On ML.*, pp. 2269-2279, 2019.

[24] Giraud, M., Groult, R., Leguy, E. & Levé, F, "Computational fugue analysis," *Computer Music Journal*. vol. 39, no. 2, pp.77–96, Jun. 2015.

[25] Grachten, M. & Widmer, G, "Linear Basis Models for Prediction and Analysis of Musical Expression," *Journal Of New Music Research*., vol. 4, no. 4, pp.311–322, Dec. 2012.

[26] Grachten, M. & Krebs, F, "An assessment of learned score features for modeling expressive dynamics in music," *IEEE Trans. On Multimedia.*, vol. 16, no. 5, pp.1211–1218, Mar. 2014.

[27] Grindlay, G. & Helmbold, D, "Modeling, analyzing, and synthesizing expressive piano performance with graphical models," *Machine Learning*., vol. 65, pp.361–387, Jun. 2006.

[28] Gu, Y. & Raphael, C, "Modeling piano interpretation using switching kalman filter," in *Proc. for the 13th Conf. Int. Soc. For Mus. Inf. Retr.*, pp.145–151, 2012.

[29] Guo, R., Simpson, I., Kiefer, C., Magnusson, T. & Herremans, D, "MusIAC: An Extensible Generative Framework for Music Infilling Applications with Multi-level Control," in *Artificial Intelligence in Music, Sound, Art and Design.*, T, Martins., N, Rodríguez-Fernández., S, M, Rebelo, Eds. vol 13221, pp. 341-35, 2022.

[30] Hadjeres, G. and Crestel, L, "The Piano Inpainting Application," 2021, *arXiv:2107.05944*.

[31] Hashida, M., Nagata, N. & Katayose, H, "Pop-E: a performance rendering system for the ensemble music that considered group expression," in *Proc. 9th Int. Conf. Music Perc. & Cogn*, 2006, pp.526–534.

[32] Hashida, M., Tanaka, S., Baba, T. & Katayose, H, "Mixtract: An Environment for Designing Musical Phrase Expression" in *Proc. Of Sound And Mus. Comp. Conf.*, 2010, pp.21–24.

[33] Hawthorne, C., Stasyuk, A., Roberts, A., Simon, I., Huang, C., Dieleman, S., Elsen, E., Engel, J. & Eck, D, "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset", presented at the International Conf. on Learning Representations, Morial Convention Center, New Orleans, USA, May 6–9, 2019.

[34] Herremans, D. & Chew, E, "Tension ribbons: Quantifying and visualising tonal tension," in *Proc. 2nd Inter. Conf. on Tech. for Music Notation. & Repres,* 2016.

[35] Hsiao, W., Liu, J., Yeh, Y. & Yang, Y, "Compound Word Transformer: Learning to Compose Full-Song Music over Dynamic Directed Hypergraphs," in *Proceedings Of The AAAI Conf. On AI*, vol, 35, pp.178–186, 2021.

[36] Huang, C., Vaswani, A., Uszkoreit, J., Shazeer, N., Simon, I., Hawthorne, C., Dai, A., Hoffman, M., Dinculescu, M. & Eck, D, "Music transformer," 2018, *arXiv:1809.04281*.

[37] Jeong, D., Kwon, T., Kim, Y., Lee, K. & Nam, J, "VirtuosoNet: A Hierarchical RNN-based system for modelling expressive piano performance," in *Proc. of the 20th Conf. for Int. Soc. For Mus. Inf. Retr.*, 2019, pp. 908-915.

[38] Jeong, D., Kwon, T., Kim, Y. & Nam, J, "Graph neural network for music score data and modeling expressive piano performance," in *Proceedings Of The Int. Conf. On ML Res.*, 2019, pp. 3060-3070.

[39] Katayose, H., Hashida, M., De Poli, G. & Hirata, K, "On Evaluating Systems for Generating Expressive Music Performance: the Rencon Experience," *Journal Of New Music Research*, vol. 41, pp.299–310, Dec. 2012.

[40] Kingma, D. & Ba, J, "Adam: A method for stochastic optimization," presented at the International Conf. on Learning Representations, San Diego, CA, USA, May 7–9, 2015.

[41] Kirke, A. & Miranda, E, "A survey of computer systems for expressive music performance," *ACM Computing Surveys.*, vol. 42, pp.1–41 Dec. 2009.

[42] Kosta, K., Ramírez, R., Bandtlow, O. & Chew, E, "Mapping between dynamic markings and performed loudness: a machine learning approach," *Journal Of Math. And Music*., vol. 10, pp.149–172, Aug. 2016.

[43] Lerdahl, F. & Jackendoff, R, "A Generative Theory of Tonal Music, reissue, with a new preface," Cambridge, MA, USA: MIT Press, 1996.

[44] Louie, R., Coenen, A., Huang, C., Terry, M. & Cai, C, "Novice-AI Music Co-Creation via AI-Steering Tools for Deep Generative Models," in *Proc. of the CHI Conf. On Human Factors In Comp. Systems*, pp. 1–13, 2020.

[45] Madaghiele, V., Lisena, P. & Troncy, R, "Mingus: melodic improvisation neural generator using seq2seq," in *Proc. of the 22nd Conf. for Inter. Soc. Of Mus. Info. Retr.*, pp. 412–419, 2021.

[46] Malik, I. & Ek, C, "Neural Translation of Musical Style," presented at the 31st Conf. on Neural Info. Proc. Syst., Long Beach, CA, USA, Dec. 4–9, 2017.

[47] Moulieras, S. & Pachet, F, "Maximum entropy models for generation of expressive music," *arXiv,:1610.03606*, 2016.

[48] Narmour, E, "The analysis and cognition of basic melodic structures: The implication-realization model," Chicago, IL, USA:University of Chicago Press, 1990.

[49] Oore, S., Simon, I., Dieleman, S., Eck, D. & Simonyan, K, "This time with feeling: Learning expressive musical performance," *Neural Computing And Applications*.,

vol. 32, pp.955–967, Nov. 2020.
[50] Palmer, C, "Timing in Skilled Piano Performance," Ithica, New York, USA: Cornell University Press, 1988.
[51] Payne, C. "MuseNet." OpenAI.com. https://openai.com/blog/musenet/ (accessed Sept. 4, 2023).
[52] Rector, M, "Historical Trends in Expressive Timing Strategies: Chopin's Etude, Op. 25 no. 1," *Empirical Musicology Review*., vol. 15, pp.176–201, 2021.
[53] Repp, B, "Patterns of expressive timing in performances of a Beethoven minuet by nineteen famous pianists," *The Journal Of The Acoustical Society Of America*., vol. 88, pp.622–641, 1990.
[54] Repp, B, "On Determining the Basic Tempo of an Expressive Music Performance," *Psychology Of Music*., vol. 22, pp.157–167, 1994.
[55] Roberts, A., Engel, J., Mann, Y., Gillick, J., Kayacik, C., Nørly, S., Dinculescu, M., Radebaugh, C., Hawthorne, C. & Eck, D, "Magenta studio: Augmenting creativity with deep learning in ableton live," in *Proc. of the Int. Workshop On Musical Meta.*, 2019, pp. 1–7.
[56] Seashore, C, "Objective Analysis of Music Performance," Iowa City, Iowa, USA:University of Iowa Press, 1936.
[57] Shi, Z, "Computational analysis and modelling of expressive timing in Chopin mazurkas," in *Proc. of the 22nd Conf. Int. Soc. Mus. Inf. Retr.*, 2021, pp.650– 656.
[58] Stables, R., Endo, S. & Wing, A, "Multi-Player microtiming humanisation using a multivariate Markov model," in *Proc. Of The 17th Int. Conf. On Dig. Audio Effects*, 2014, pp.1–6.
[59] Sundberg, J., Frydén, L., Bodin, L. & Friberg, A, "Performance rules for computer controlled performance of contemporary keyboard music," *STL-QPSR*., vol. 28, pp.79–85, 1987.
[60] Tan, H., Luo, Y. & Herremans, D, "Generative Modelling for Controllable Audio Synthesis of Expressive Piano Performance," 2020, arXiv:2006.09833.
[61] Tchemeube, R., Ens, J., Plut, C., Pasquier, P., Safi, M., Grabit, Y. & Rolland, J, "Evaluating Human-AI Interaction via Usability, User Experience and Acceptance Measures for MMM-C: A Creative AI System for Music Composition," in *Proc. of the 32nd Int. Joint Conf. on AI.*, 2023, pp.5769–5778.
[62] Doorn, J., Ly, A., Marsman, M. & Wagenmakers, E, "Bayesian Rank-Based Hypothesis Testing for the Rank Sum Test, the Signed Rank Test, and Spearman's Q," *Journal Of Applied Stat.* , vol. 47, pp.2984–3006, 2020.
[63] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L. & Polosukhin, I, "Attention is all you need," in *Proc. of the 31st Int. Conf. on Neural Info. Proc. Syst.*, 2017, pp.1–11.
[64] Widmer, G, "Machine Discoveries: A Few Simple, Robust Local Expression Principles," *Journal Of New Music Research*., vol. 31, pp.37–50, Aug, 2002.
[65] Widmer, G. & Goebl, W, "Computational models of expressive music performance: The state of the art," *Journal Of New Music Research*., vol. 33, pp.203–21, Feb, 2004.
[66] Williamon, A., Thompson, S., Lisboa, T. & Wiffen, C, "Creativity, originality and value in music performance," in *Musical Creativity: Multidisciplinary Research In Theory And Practice*, I, Deliege., G, Wiggins, Eds. Hove, East Sussex, UK: Psychology Press, 2006, pp.161–180.
[67] Wu, S. & Yang, Y, "Compose & Embellish: Well-Structured Piano Performance Generation via A Two-Stage Approach," presented at *IEEE Int. Conf. On Acous., Speech And Sig. Proc. (ICASSP)*, Rhode Island, Greece, Jun. 04–10.
[68] Yin, Z., Reuben, F., Stepney, S. & Collins, T, "A Good Algorithm Does Not Steal – It Imitates: The Originality Report as a Means of Measuring When a Music Generation Algorithm Copies Too Much," *Artificial Intelligence In Music, Sound, Art And Design*, vol. 12693, pp.360–375, Apr. 2021.
[69] Yin, Z., Reuben, F., Stepney, S. & Collins, T, "Measuring When a Music Generation Algorithm Copies Too Much: The Originality Report, Cardinality Score, and Symbolic Fingerprinting by Geometric Hashing," *SN Computer Science*., vol. 3, pp.1–18, Jun. 2022.
[70] Yin, Z., Reuben, F., Stepney, S. & Collins, T, "Deep learning's shallow gains: A comparative evaluation of algorithms for automatic music generation," *Machine Learning*., vol. 112, pp.1785–1822, Mar. 2023.
[71] Zanon, P. & De Poli, G, "Estimation of Parameters in Rule Systems for Expressive Rendering of Musical Performance," *Computer Music Journal*., vol. 27, pp.29–46, 2003.

Kyle Worrall is a PhD student with the Centre for Doctoral Training in Intelligent Games and Game Intelligence (IGGI) at the University of York, specialising in music generation and expressive rendering for video games. Kyle received his MSc degree in Sound and Music for Interactive Games from Leeds Beckett University in 2016, and a BA (Hons) in Creative Music Production from Manchester Metropolitan University in 2015.

Zongyu Yin is a research scientist with the Speech, Audio and Music Intelligence team at TikTok. He received his PhD in new deep learning methods for the evaluation of music generation in 2022 from the University of York.

Tom Collins is an Associate Professor in Music Technology at Frost School of Music at the University of Miami, and also heavily involved in the music cooperative MAIA, Inc. Tom studied Music at Cambridge, Mathematics and Statistics at Oxford, and did his PhD on automatic pattern discovery and music generation at the Open University. He has held multiple positions in the US and Europe.