Automatic Generation of Expressive Piano Miniatures

Simon Colton, Louis Bradshaw, Berker Banar and **Keshav Bhandari**
School of Electronic Engineering and Computer Science, Queen Mary University of London, UK

Abstract

We describe an approach to the automatic generation of short piano compositions known as *miniatures*. At the heart of this is a transformer model which produces expressive performances of miniatures which are then transcribed into score form and edited. We present a pilot study to investigate the quality of the outputs and provide an illustrative example. We further look at the bigger picture of how generative AI systems such as ours can automatically add to musical culture.

Introduction

In an essay celebrating short piano compositions, Allan Hepburn (2006) describes *piano miniatures* as pieces that "usually last between 30 and 120 seconds" which "contest principles of repetition and duration" and "bespeak economy in art, not waste". They can be "as shocking as a scream or as intimate as a whisper" and often "push emotions to extreme limits". The author lists dozens of compendiums of piano miniatures from Glazunov (Drei Miniaturen, op. 42, 1893) to Stafford (Twelve Miniatures, op. 32, 1998), and argues that smallness and triviality should not be conflated in this context.

Described by Hepburn in this way, piano miniatures are a recognised musical form of potential interest to computational creativity research. This is firstly because they simplify the generative problem somewhat, reducing the need for cohesive and interesting long-term structure over lengthy durations, a well known difficulty in generative music (Bhandari and Colton 2024). Secondly, miniatures offer the opportunity to influence the generative process via the use of neural listening models to estimate the emotional import the music will have on human listeners. Piano miniatures are intended to be played from a musical score, but it is also useful to hear a performance of the piece. Hence, for the purposes of our work here, we define a piano miniature as a pair (C, R) where C is composition written in score form as a PDF file, and R is an expressive audio rendering of C as an MP3 file.

In the next section, we describe a generate and test approach to automatically producing MIDI piano miniatures from which audio files can be produced. Following that, we provide a pilot listening study to assess the value of this approach, from which we derive a method using listening models for estimating the musical value of a generated miniature. We then describe how we use this estimation method to further filter and edit miniatures, and how we transcribe them into score notation. We conclude with a discussion of how such an approach could add to musical culture and describe some avenues for future research.

Generate and Test Stages

Building on Huang et al. (2019) and Thickstun et al. (2023), we have trained 3 transformer neural models (Vaswani et al. 2017) of different sizes, labelled small, medium and large. Each generates token sequences easily converted to MIDI and rendered into audio using Fluidsynth (fluidsynth.org). Generation is performed in an *auto-completing* manner, i.e., in response to a given musical *prompt* which comprises an extract of tokenised music, given as input to the model. Notes for piano music are represented as token triples of the form:

$$\langle \texttt{piano}, p, v \rangle \quad \langle \texttt{onset}, o \rangle \quad \langle \texttt{dur}, d \rangle$$

with p and v being the MIDI pitch and volume of the note, o and d being its onset and duration (in milliseconds).

Prompts for the Aria models can be prefixed by *meta-tags*; for the experiments here, we prefix these two meta-tags:

$$\langle \texttt{prefix}, \texttt{instrument}, \texttt{piano} \rangle \quad \langle \texttt{prefix}, \texttt{genre}, g \rangle$$

where g is a genre, namely either `jazz` or `classical`. These condition the generation, increasing the likelihood of producing piano music in either a jazz or classical genre.

Full details of training the Aria models, and their positive evaluation against state of the art auto-completion MIDI generators, is as yet unpublished. Note that the models have been trained on MIDI data from multiple sources, some of which are not piano music, and which vary in quality (in terms of accuracy w.r.t. the composed/performed music). Importantly, much of the data is transcribed from human performances, hence the Aria *base* models are trained to produce expressively performed music (Cancino-Chacon et al. 2018).

Following a standard approach, we fine-tuned the base models on 10,000 high-quality jazz and classical piano transcriptions. Fine-tuning trains only the final layers of the transformer, and the idea is that the ability of the base models to generate music in general will transfer (Weiss, Khoshgoftaar, and Wang 2016) onto the task of making piano music, undertaken by the fine-tuned models. The base training and fine-tuning provided 6 separate models for experimentation, denoted BS, BM, BL, FS, FM, FL, spanning (B)ase/(F)ine-tuned and (S)mall/(M)edium/(L)arge models.

After some early experimentation with generated musical prompts, we realised that the Aria models can be used in an *ab-initio* way, i.e., with no musical input, only the meta-tag tokens. We found that the outputs from the ab-initio approach were musically interesting, good quality in general and covered a variety of styles and moods. We are interested in generating 30 second piano miniatures, with discernible beginnings, middle sections and endings. As such, generating a stream of music until 30 seconds worth is reached and then stopping is inappropriate, as this produces ill-formed compositions with abrupt endings.

Aria transformers produce music by generating one token after another, providing an opportunity to exert control at various points, for instance to prompt the music to end. To enable this, during training, after tokenisation of the music data, an additional $\langle D \rangle$ token was added 130 tokens before the end for each training example, and an $\langle S \rangle$ token was concatenated to the sequence. This meant that – to some extent – the Aria models learned how to react to the $\langle D \rangle$ token, by bringing a sequence to a musical end, e.g., by slowing it down and/or using cadences, repeating an earlier passage, etc., then signifying the end with an $\langle S \rangle$ token.

In practice for generating piano miniatures, we monitor the sequence of tokens a model is producing, then at the right moment after a certain number of tokens, we introduce a $\langle D \rangle$ token to start bringing the music to an end after roughly 30 seconds. That is, after each triple of tokens representing a note is generated, the average note duration is calculated, from which an estimate of how long the overall piece would be if it ended after a further 130 tokens were generated. When this duration estimate reaches 30 seconds, the $\langle D \rangle$ token is introduced. We found this very reliable in introducing the $\langle S \rangle$ after 30 seconds on average, and (subjectively) around 60% of miniatures end in an appropriate musical fashion.

To improve the yield of miniatures, we used batch processing, with a batch size up to 8, depending on the model size, due to memory limitations. We noticed that a 30 second piece of very fast music could require hundreds of tokens, while a slow piece may only require a few dozen. This meant that some sequences in the batch were finished considerably earlier than others, and we implemented a *re-prompting* method to restart the generation process for completed batch lines while others were still finishing. We also found that stopping the process after each batch line had produced at least one full piece improved the yield, and we were able to produce up to three times more miniatures in one run.

As evidenced in the pilot listening study below, the output miniatures from the Aria models are generally high quality. However, a number of output pieces are difficult to listen to, and we decided to implement around twenty rule-based methods to aggressively reject miniatures which had issues of one or more of the following types:

- **Ill-formed token sequences:** the Aria models are probabilistic, so occasionally token sequences prescribe non-piano instrument notes, or poorly-specified musical notes.
- **Too slow or inhumanly fast:** some pieces contained note sequences too fast to be feasibly played by a person on a piano, or notes which were tediously drawn out.
- **Too little variety:** some miniatures had all notes with the same dynamics (volume) and/or durations, which made them sound mechanical and not expressively performed.
- **Too much repetition:** some pieces had overt repetition of note sequences, often a single pitch repeated dozens of times.
- **Too much silence:** some miniatures were rather lumpy, with bursts of notes, then long periods with no notes playing, which made them awkward to listen to.
- **Melodies only:** some pieces had only one note playing at once, i.e., just a melody – while these were fine to listen to, they didn't fit the required form, which takes advantage of the polyphonic nature of the piano.
- **Out of duration range:** due to a poor estimation of when to introduce the $\langle D \rangle$ token, some miniatures were less than 20 seconds or longer than 40, which didn't fit our requirements.

	Genre Meta-tag			Any
Model	None	Jazz	Classical	Genre
BS	68.0	30.0	76.0	58.0
FS	40.0	30.0	42.0	37.3
BM	58.0	30.0	**78.0**	55.3
FM	35.0	27.0	37.0	33.0
BL	63.0	37.0	72.0	57.3
FL	39.0	**18.0**	32.0	29.7
Any Small	54.0	30.0	59.0	47.7
Any Medium	46.5	28.5	57.5	44.2
Any Large	51.0	27.5	52.0	43.5
Any Base	63.0	32.3	75.3	56.9
Any Fine-tuned	38.0	25.0	37.0	33.3
Any	50.5	28.7	56.2	45.1

Table 1: Percentage of rejected vignettes in terms of the generative setups used to produce them. The highest and least percentages are in bold.

Note that, due to the symbolic nature of Aria's output, we could fix some issues to avoid a rejection. In particular, we applied a stretching technique to extend/compress any pieces less than 15s or longer than 45s back to the acceptable 20s-40s range, by altering note onsets and durations. Also, we found that lengthening the last notes played in a miniature often made their endings less abrupt in an acceptable way.

There are 3 (B)ase and 3 (F)ine-tuned Aria models to experiment with, and options to seed miniature generation with the `classical` or `jazz` genre meta-tags in the prompt, or with neither. This gives 18 generative setups, and to investigate their output quality, we generated 100 miniatures for each setup. In table 1, we recorded the percentage outputs for each setup that were rejected for the reasons given above. We see that, in general, the larger the model size in the generative setup, the fewer miniatures were rejected, and fine-tuned models produced substantially fewer rejected outputs. Also, using the `jazz` meta-tag reduced the number of rejections in general, which may be due to the high quality of the jazz pieces in the training data. The 988 non-rejected miniatures were taken forward to the following listening study.

A Pilot Listening Study

The rejection method prunes miniatures with obvious issues, but doesn't help estimate the musical quality of the remaining pieces. To begin to investigate the human-rated quality of the outputs from the 18 generative setups, we undertook a listening study where four participants were given 81 pairs of miniatures, m_1 and m_2 to listen to and compare. m_1 and m_2 were produced by different generative setups. Participants – each of whom had a background in musical performance and composition – were asked to rate each of the pair as either

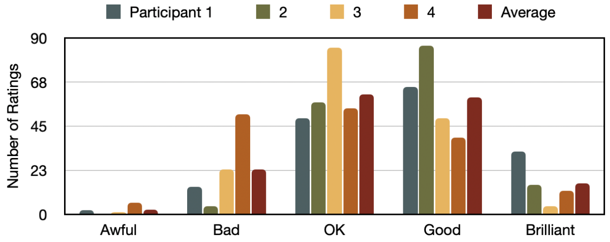


Figure 1: Overall ratings of miniatures in the listening study.

awful, *bad*, *OK*, *good* or *brilliant*, and to choose which of m_1 or m_2 they preferred due to its higher quality (or that they had no preference). The 81 pairs were chosen so that certain generative setups were compared against certain others.

In total, 648 ratings of 464 different miniatures were assigned by one or more of the participants. In figure 1, we chart the number of ratings in each of the categories, per participant. We see that, in general, the miniatures were well received by all four participants. Two participants chose *Good* more often than any other rating, while the other two chose *OK* more. Overall, 46.6% of miniatures were rated as *Good* or *Brilliant*, with 15.6% rated as *Awful* or *Bad*.

By assigning appropriate scores to the ratings, we were able to compare and contrast the outputs from the different generative setups. We found that, on average, large model outputs were rated higher than medium model ones, which in turn were rated higher than small model outputs. Also, outputs from fine-tuned models were rated higher than from the base models, although the difference was not particularly marked. The highest performing generative setup was FL (the fine-tuned large model) with the `jazz` meta-tag, scoring around 25% higher than the worst setup, which was was BS (the base small model) with the `classical` meta-tag.

By further looking at the direct comparison that participants made, we found that they preferred large and medium model outputs to those from small models, using a binomial test to reject the null hypothesis of no preference. Interestingly, though, we found that for 65% of the times when participants chose between a medium and a large model output, the medium one was was preferred: a statistically significant preference with p-value of 3.55×10^{-6} from a binomial test. While outputs from fine-tuned models were preferred slightly more often than the base model outputs, this was not statistically significant, so we hypothesise that fine-tuning improves output quality by reducing obvious issues for rejection, but not necessarily by increasing musical quality.

Finally, given the score from the ratings, we were able to investigate correlations between audible properties of the music and perceived quality. In particular, we used TensorFlow listening models (Alonso-Jiménez et al. 2020) for music categorisation, which we have previously found to reliably tag piano music in terms of genres (e.g., jazz) and moods (e.g,. sad). Testing against 197 such tags, we found that the activation of the 'classical' tag had the highest Pearson product-moment correlation (0.28) with participant ratings of the miniatures. We operationalised this finding to improve quality, by rejecting any miniature scoring below a threshold for the activation of the 'classical' tag. This threshold is only achieved by 10% of the 1,800 miniatures generated for the above experiment, and we found that using it substantially increases the average quality of non-rejected miniatures.

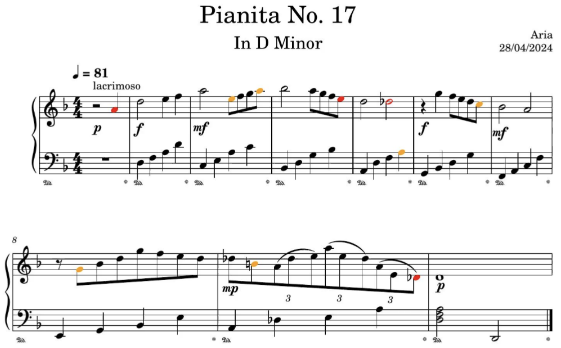


Figure 2: A generated piano miniature. In the expressive performance, orange notes are held slightly longer than prescribed in the score; red notes are held substantially longer.

Transcription and Editing Stages

To produce a score representation of the miniatures, we transcribe Aria-generated MIDI-files to MusicXML, and use MuseScore (musescore.com) to mark this up into a PDF. Producing the MusicXML file is not straightforward, however, as Aria's outputs are expressive performances. That is, they mirror human rubato playing and sustain pedal usage, as well as composer instructions for changes in bar tempi, e.g., with an accelerando marking. For instance, a note represented as a quaver may be audible in the MIDI file for an entire bar, if played when the sustain pedal is just pressed, but other quavers may be audible for just a small fraction of the bar.

We are currently investigating an approach to transcribing miniatures involving *projecting composer intent*, i.e., detecting regularities in parts of the music to ground the entire composition. To illustrate this, the 29-second miniature in figure 2 was produced by the FM Aria model and transcribed automatically. It is (subjectively) quite a melancholic, beautiful piece in the original Aria rendering. Here, the majority of the intervals between onsets of notes in the bassline are similar enough to project that all the notes should be the same duration on the score. With beat detection from the Librosa package (Ellis 2007), it is possible to further project a 4:4 time signature, with the bassline notes each being presented as a crotchet. With the bassline quantised in this way, it is possible to add in melody and chord notes which are synchronised with the bassline, to complete the transcription.

To extract melody and basslines to detect patterns in, we use a modified version of the skyline algorithm (Chai et al. 2001), which can use note pitch and/or note velocity to identify melody and bassline note sequences. In addition, we use the listening models mentioned above to estimate the mood of the piece, and project this with a composer's mark at the start (e.g., 'lacrimoso' in the D minor miniature of figure 2). Currently, the transcription process is semi-automated,

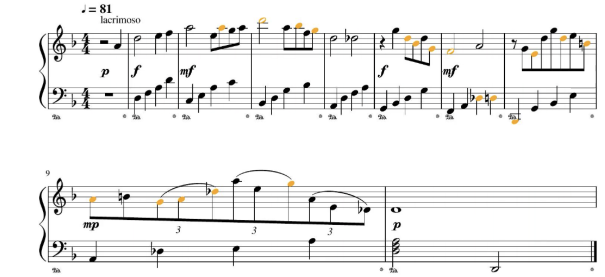


Figure 3: Version of the miniature from figure 2, edited w.r.t. the tag ‘inspiring’. Altered notes are highlighted in orange.

with the user setting various parameters, such as the smallest quantisation unit, and only around 10% of the miniatures can be transcribed by this approach. We plan to use more sophisticated methods, e.g., those described in Benetos et al. (2019), Liu et al. (2022) and Toyama et al. (2023).

With a transcribed version of a miniature, it is possible to see how the version prescribed by the score and the expressive version produced by Aria differ, in order to understand how it is performed. In figure 2, the coloured notes are those that, in the original performance, were played for noticeably longer than the score would suggest. We see that the Aria performance consistently uses rubato in the quartet of melody quavers in bars 3, 4 and 6 and the penultimate melody note is held significantly. Subjectively, this expressiveness in performance really adds emotional content and overall musical value to the piece. This analysis also affords an editing process, where the durations of notes are stretched or compressed in the same direction as in the original performance, producing an exaggerated version. We have found that for some of Aria’s pieces, such exaggeration improves the expressive quality of the performance; in others it degrades the quality, and we intend to investigate this further.

To develop a miniature, similar to our work in (Colton, Banar and Cardinale 2023) , we use the listening models mentioned above to edit it, hopefully increasing emotional impact. Given a *target tag* such as ‘melancholic’, for each note in a generated miniature, every different pitch within an octave neighbourhood for the note is substituted, if the pitch class for the substitution pitch is already used somewhere in the piece. For each substituted version, an audio file covering 3 bars around the substitution is generated and passed through the listening models. If a substitution increases the activation of the target tag and the ‘classical’ tag (as per the listening models) it is recorded. Increasing the ‘classical’ tag activation helps keep the overall musical quality of the edits high, given its correlation with perceived value identified in the listening study. The alternate pitch (if there is one) which improves both tags the most is then used as an edit to the piece.

Figure 3 portrays a variation after editing the miniature of figure 2 w.r.t. target tag ‘inspiring’. Subjectively, we find this version to have achieved a more inspiring sound, while retaining the beauty of the original piece. We have had similarly interesting and valid results using target tags including ‘dark’, ‘melancholic’ and ‘drama’ (see appendix).

Conclusions and Future Work

With commercial generative music services such as Suno (suno.com), Udio (udio.com) and Stable Audio (stability.ai/stable-audio) advancing at speed, it is important to remember that music is not just for entertainment, but also for education, inspiration, community building and as a pastime. We aim to produce and publish a book of diverse piano miniatures called *Pianitas*, accompanied with expressive audio performances in an album. We would hope that each piece is high enough quality to be considered a beautiful addition to the genre, delivering real emotional impact in a short form, as alluded to by Hepburn (2006). Each miniature will be annotated by margin notes describing a musical concept in the composition and/or the recording. By producing such annotated, playable, scores for miniatures, we aim to build generative AI systems which can *add to musical culture* themselves, as proposed in (Colton and Banar 2023).

As an example, when we played the miniature of figure 2 on the piano, we noticed that the D-flat note in the melody of bar 5 clashed with the F in the left hand. This was not audible in the original Aria performance, and we found this is because the F is played very quietly there and is shorter in duration than prescribed on the score. While not a particularly important musical concept, and not foregrounded (yet) explicitly by the system, it does highlight potential for conveying composition and performance concepts with our approach. Also, in the Aria performance, the penultimate note in the melody of figure 2 is held for a long time, so could have been transcribed with a fermata (pause) composer’s mark. Co-creative blurring of responsibilities for composer, performer and improviser (Jordanous and Keller 2012) is something we will investigate with the approach described here.

Influenced by work such as (Herremans and Chew 2016) and (Pearce, Meredith, and Wiggins 2002), we will investigate more fine-grained control of Aria’s step-wise token prediction process. That is, rather than generating and then trying to transcribe a miniature to a score, we will implement an approach which *models*, rather than *projects* composer intent. In particular, at each stage, Aria provides a probability for each of 12,000 tokens to appear next in the sequence. We will investigate how to intervene here, so that – rather than choosing the most likely – the next token is chosen as a highly-likely one which also satisfies certain constraints arising from an intended form or emotional expression. In this way, we believe it will be possible to keep the quality of Aria’s output – influenced as it is by the neural model’s understanding of music in general and the flow of the piece currently being composed – with the foregrounding of a musical concept such as a novel accompaniment style, method for achieving counterpoint or chord sequence.

For a concept to add to musical culture, it needs to be understood, owned and developed by the people exposed to it. Hepburn points out that miniatures are perfect to help with this: pithy observations highlighting a general truth, which “state principles...” so that ”[m]eaning shimmers around them”. As such, we believe that generating idea-rich piano miniatures to investigate musical concept formation and dissemination is a worthy target for computational creativity research.

Acknowledgements

We thank the anonymous reviewers for their valuable comments. This work is supported by the UKRI Centre for Doctoral Training in AI and Music (grant number EP/S022694/1).

References

Alonso-Jiménez, P.; Bogdanov, D.; Pons, J.; and Serra, X. 2020. Tensorflow audio models in Essentia. In *Proc. ICASP*.

Benetos, E.; Dixon, S.; Duan, Z.; and Ewert, S. 2019. Automatic music transcription: An overview. *IEEE Signal Processing Magazine* 36(1):20–30.

Bhandari, K., and Colton, S. 2024. Motifs, phrases, and beyond: The modelling of structure in symbolic music generation. In *Proceedings of the International Conference on Computational Intelligence in Music, Sound, Art and Design.*

Cancino-Chacon, C.; Grachten, M.; Goebl, W.; and Widmer, G. 2018. Computational models of expressive music performance. *Frontiers of Digital Humanities* 5.

Chai, W.; Vercoe, B.; Paradiso, J.; and Schmandt, C. 2001. Melody retrieval on the web. *Proceedings of SPIE - The International Society for Optical Engineering*.

Colton, S.; Banar, B.; Cardinale, S. 2023. Neuro-Symbolic Composition of Music with Talking Points. *Proc. ICCC*.

Colton, S., and Banar, B. 2023. Automatically adding to artistic cultures. In *Proceedings of the Int. Conference on Computational Intelligence in Music, Sound, Art and Design*.

Ellis, D. 2007. Beat tracking by dynamic programming. *Journal of New Music Research* 36(1):51–60.

Hepburn, A. 2006. Piano miniatures: An essay on brevity. *The Gettysburg Review* 19(1):89–105.

Herremans, D., and Chew, E. 2016. Morpheus: automatic music generation with recurrent pattern constraints and tension profiles. In *Proc. IEEE Region 10 Conference*.

Huang, A.; Vaswani, A.; Uszkoreit, J.; Simon, I.; Hawthorne, C.; Shazeer, N.; Dai, A.; Hoffman, M.; Dinculescu, M.; and Eck, D. 2019. Music transformer. In *Proc. ICLR*.

Jordanous, A., and Keller, B. 2012. What makes musical improvisation creative? *J. Inter. Music Studies* 6:151–175.

Liu, L.; Kong, Q.; Morfi, V.; and Benetos, E. 2022. Performance midi-to-score conversion by neural beat tracking. In *Proc. Int. Soc. for Music Information Retrieval Conference*.

Pearce, M.; Meredith, D.; and Wiggins, G. 2002. Motivations and methodologies for automation of the compositional process. *Musicae Scientiae* 6(2):119–147.

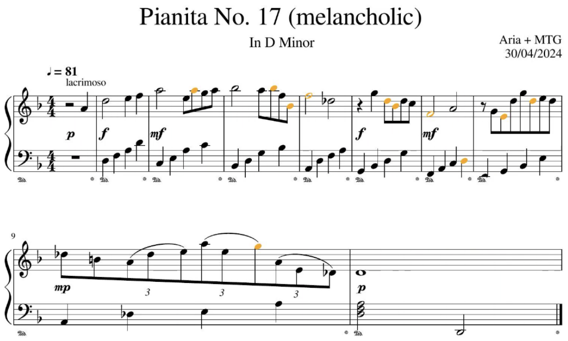
Thickstun, J.; Hall, D.; Donahue, C.; and Liang, P. 2023. Anticipatory music transformer. *arXiv:2306.08620*.

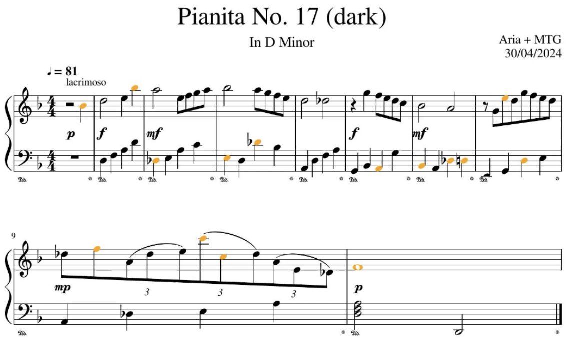
Toyama, K.; Akama, T.; Ikemiya, Y.; Takida, Y.; Liao, W.-H.; Mitsufuji, Y. 2023. Automatic piano transcription with hierarchical frequency-time transformer. In *Proc. 24th ISMIR*.

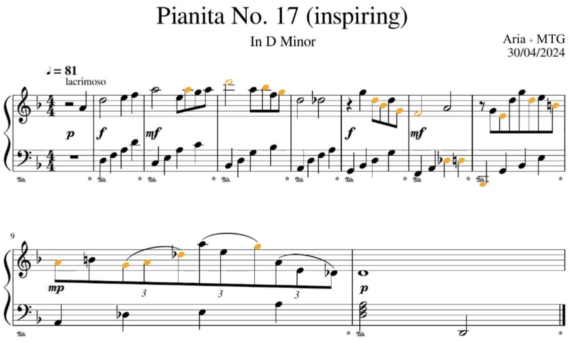
Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. *Proc. 30th NeurIPS*.

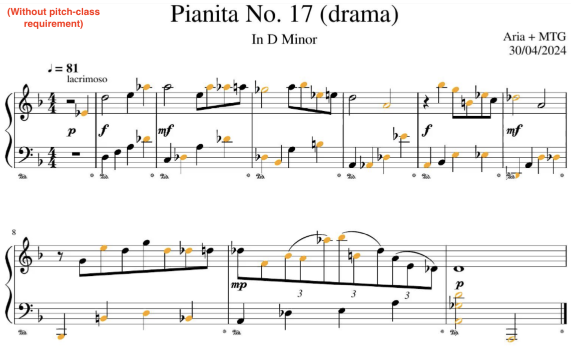
Weiss, K.; Khoshgoftaar, T.; and Wang, D. 2016. A survey of transfer learning. *Journal of Big Data* 3.

Appendix


Pianita No. 17 (melancholic)
In D Minor
Aria + MTG
30/04/2024
♩ = 81
lacrimoso
p
f
mf
f
mf
9
mp
3
3
3
p


Pianita No. 17 (dark)
In D Minor
Aria + MTG
30/04/2024
♩ = 81
lacrimoso
p
f
mf
f
mf
9
mp
3
3
3
p


Pianita No. 17 (inspiring)
In D Minor
Aria + MTG
30/04/2024
♩ = 81
lacrimoso
p
f
mf
f
mf
9
mp
3
3
3
p


(Without pitch-class requirement)
Pianita No. 17 (drama)
In D Minor
Aria + MTG
30/04/2024
♩ = 81
lacrimoso
p
f
mf
f
mf
8
mp
3
3
3
p