composer AI models. We focused on amateurs as a starting point, as we hypothesized that they could benefit most from a continuous interaction with a composing AI. In our approach, users query the AI for suggestions but make all decisions, keeping the composition personalized.

The AI analyst, on the other hand, is interested in investigating a composing AI's behaviour and the influence of different input melodies and parameters on output suggestions. Understanding this behaviour is crucial for the user's trust and leads to more efficient steering.

The main workflow of our approach looks as follows: Users start with recording or generating a seed melody. Based on it, they query the AI for multiple continuation suggestions with chosen parameters. We then visualize these samples to help users filter, listen to, select, and customize the most interesting continuations for a personal composition. Repeating these steps, or even creating multiple levels of continuations, results in a tree of melody samples, where a sample's children are the possible continuations for the remaining composition. Using the same steps to replace an existing part (fill-in) splits the composition node in the graph and adds the fill-in samples as nodes in between, resulting in a directed acyclic graph (Figure 2).

This approach is abstractly inspired by the idea of visual parameter space analysis [40], and also practically by the artist MJx Music. He used an AI to create an album by manually generating hundreds of samples, listening to all, selecting the most interesting, and combining them into songs [3]. In contrast to this brute-force workflow, we add visualization to provide similar flexibility but more usability, by showing overviews from which users then pick the most interesting samples to further investigate.

To make this general idea possible, we needed to investigate metrics and aggregations to be able to relate different samples to each other, as well as visualization designs which allow users to interact with the space of samples.

3.1 Metrics and Aggregations

We designed a range of metrics that help our visualizations arrange or summarize a collection of melody samples. A user might have some idea of prioritization when looking for samples with specific properties. Some of our visualizations therefore allow sorting by different aspects, such as the variance of interval sizes as a statistical metric to detect lively, varied, or monotone melodies. We can also sort by two metrics at once, with a scatterplot that uses each metric as one axis. As a starting point, we chose the following metrics: mean note duration, variance of interval sizes, similarity to composition, distinct pitches, range of pitches, temperature (AI parameter affecting randomness [4]), and number of noteshapes from composition. The latter indicates how often a shape of three consecutive notes (intervals between them) occur in the composition.

The similarity between samples can also be used for sorting, for example to focus on samples that are more similar to parts from the composition. Furthermore, pairwise similarities between all samples can be used to place them on a "map" where similar ones are closer to each other. We designed a rather simple similarity metric that takes both pitch and rhythm into account, and allows the user to choose a weight for controlling the impact of both.

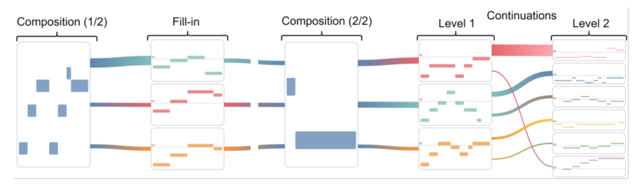

Figure 2. Node-link diagram with one fill-in and two levels of continuations. Colors help differentiate between melody samples. Here, nodes in the same level are sorted by variance of intervals descending, which is also encoded in the link width.

Above metrics can also be used to compactly represent samples, by only visualizing the metrics instead of all notes, as we do in our glyphs and histograms.

Our example metrics are not necessarily complete or optimal, but serve as a starting point for further research. For some tasks, what is optimal can even be hard to define at all or be subjective, such as sorting by "happiness" or determining the degree of "similar feeling", possibly requiring users to fine tune or train metrics by themselves. Our main approach works with any kind of metric and could be easily extended in the future.

3.2 Visualizations

Piano Rolls as Note Representation: Piano rolls serve both as representation and editor for notes. We chose piano rolls over staff notation for multiple reasons: They are more easy to read for beginners, represent each pitch (MIDI note) in its own row, visually show rhythm and breaks through block size and gaps, and are better readable when small, as there are less fine details. The central view of our tool is a large piano roll at the top that shows all melody samples and the composition at all time (Figure 1A). In this view, we allow adjusting pitch, start, and duration of notes, as well as adding or deleting them.

Sample Relation Graphs: Users can create a tree structure of melody samples to see different continuation paths for a given melody. Listening to all paths takes time, so we want to make selection more efficient by extending and supporting the sequential listening process with a parallel visual approach. To this end, we visualize the tree structure and its melody samples together (Figure 2).

Displaying all samples in a single piano roll at once would lead to overlap and clutter. Therefore, we show the tree structure in an icicle plot, using most space to display piano rolls (Figure 1B). Children (continuations) of a sample are displayed on the right of its node, dividing its height equally. All piano rolls share a common timeline on the X-axis while the Y-axis is different for each. With different color encodings, users can differentiate between melodies or get additional information, for example about

[3] magenta.tensorflow.org/mj-hip-hop-ep

[4] magenta.tensorflow.org/performance-rnn

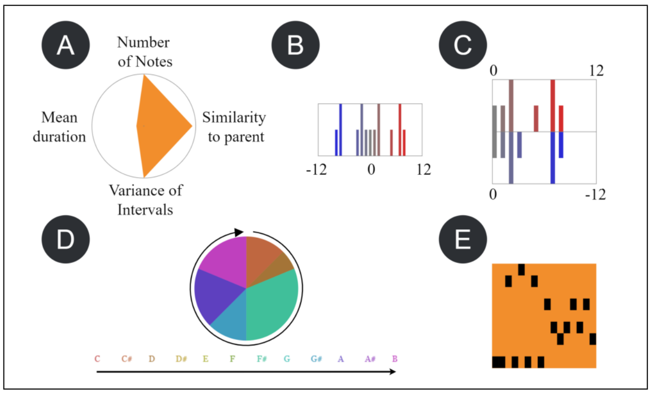

Figure 3. Comparison of our glyphs on the same data. (A) star glyph, (B, C) occurrences of intervals between consecutive notes, (D) chroma distribution, and (E) piano roll.

the parameters used to generate them. We allow listening to single samples or selecting a whole path or parts of it. Users can add a path to the composition and subsequently adjust notes manually or generate new continuations.

Sorting all melodies of the same tree level, which are alternatives for the same time span, could be useful when focusing on some priority. Since an icicle plot already encodes relations through position, sorting would break this encoding. Therefore, we added links between nodes to allow sorting melodies of each level while keeping relations visible, resulting in a node-link graph (Figure 2). Besides showing relationships, these links also encode the value of the selected sorting metric in their width.

Similarity-Based Layout and Glyphs: A composer can only use an AI efficiently if they roughly understand how it behaves. In our case, this means knowing which parameter values (e.g., temperature) to choose for an intended output. Since there is usually some uncertainty in the results, more than just a few samples are needed to draw conclusions.

Above visualizations do not scale to large numbers of samples, as small piano rolls are hard to read. While sorting and scrolling help, the overview gets lost. To provide an overview for hundreds of samples at once, we visualize them as dots in a scatterplot, placing similar once closer together via dimensionality reduction (Figure 1D).

Using a circular brush, users can select a neighborhood and take a look at its melodies, represented as piano rolls and statistical aggregations of all selected samples. To indicate the overall variety of the selection, we show two histograms for the occurrences of pitches and note durations.

Because selecting neighborhoods at random is inefficient, the user needs an impression of melodies directly inside the scatterplot. We therefore replaced the dots by glyphs, small symbols showing some kind of data, such as statistics or the melodic structure itself. Users can switch between these glyphs at any time to visually filter interesting melodies before looking at their neighbourhood in detail. As overlapping glyphs are unreadable, we apply gridification to produce a regular, occlusion-free layout.

We designed four types of glyphs to show different aspects of a melody (Figure 3): The first type shows multiple metrics in a *starglyph*. As default, four metrics are shown that convey both rhythm and melody: note count, mean note duration, variance of intervals between two notes, and similarity to notes in the current composition. Jumps between notes are often interesting for composers, for example when looking for a climax or a calm section after it, inspiring us to show the occurrences of intervals between two notes in a *histogram glyph*. Users can directly see when a melody mostly uses repeating notes and small changes or has larger jumps in it. The third type of glyph represents the tonality of melodies by a *pie chart of chroma occurrences*, i.e., one slice for each of the twelve notes. It helps selecting or investigating melodies based on contained notes and therefore tonality, as well as looking for outliers with unusual pitches. For example, a pie chart showing mostly C, E, and G likely fits into a C major composition. Because above glyphs represent metrics that can be hard to grasp for unfamiliar users, we added a fourth type that shows melodies as *small piano rolls*. With these glyphs, the rough contour of the melody is directly visible, so users can look for those matching their intention.

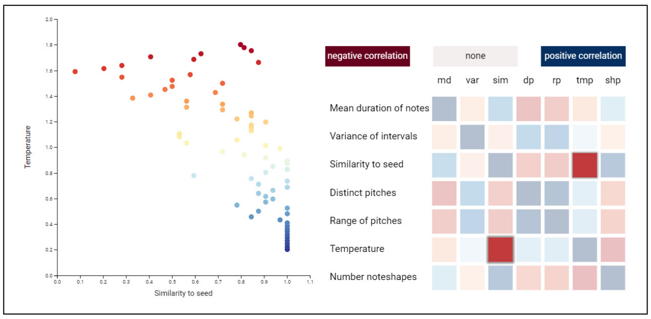

Figure 4. Negative correlation between temperature and similarity. Colors encode temperature (left) and correlation (right).

Correlation Analysis: An analyst can use the above scatterplot and glyphs to investigate sample metrics, but could also be interested in investigating correlation between metrics. We therefore provide a correlation matrix as overview of pairwise correlations between all metrics, which encodes Pearson correlation as color (Figure 4). We show negative, positive, and no correlation in red, blue, and white, so users can quickly filter interesting pairs. After choosing a pair of metrics by clicking, a 2D scatterplot shows all samples as points, positioned based on their metric values. One insight we made was a positive correlation between temperature and pitch range, so higher temperature values often lead the AI to generate more outlier notes.

Attribution: A common problem when co-creating with AI is authorship, as users are often hesitant to claim generated melodies as their own. To analyze this problem and provide a feeling for how much authorship a composer has, we added a coloring for the notes' attribution (Figure 5). We assign each note one of five attribution classes, covering the range between human- and AI-generated, and display the percentage of each class, to show composers how much and what they contributed to the composition.

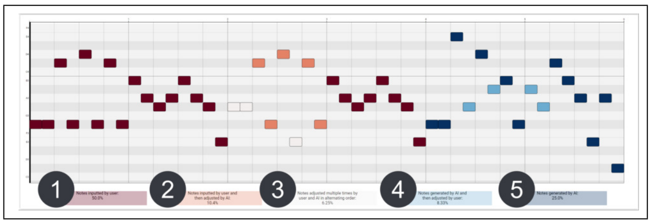

Figure 5. Color indicates the attribution of notes, from red (by user) to blue (by AI) along five steps: (1) completely human-created notes, (2) human-created but AI-adapted notes (fill-in), (3) notes changed by both parties multiple times, (4) AI-generated but user-updated notes, and (5) AI-generated, unedited notes.

3.3 Implementation

We implemented our prototype as a web app build with React, which allows publishing it as a website [1] . It works completely inside the browser with an AI model [5] made by Magenta [6] , which we chose due to easy access and usage, but other models from any framework or language can be hosted by others and integrated through plugins.

For our visualizations, we use D3 [44] for colormaps and scales, DruidJS [45] and MDS [46, 47] for computing dimensionality reduction layouts, and Hagrid [48] to avoid overlapping glyphs. Users can record melodies using MIDI [49] devices and export finished compositions as MIDI files for further processing with other tools.

4. EVALUATION

As music composition is a complex and time-consuming task without clear measures for efficiency and success, we chose a qualitative study design with five domain experts (P1 to P5). P1, P3, and P4 were pursuing a degree in composition for experimental music, P2 holds a degree in composition and studies performance, and P5 studied a music instrument major and composes occasionally. Although our approach primarily targets beginners, we chose experts, as they are more familiar with composition workflows. We therefore expected more detailed, thorough, and critical feedback from this group.

Our design brings potentially unfamiliar technology to participants, such as machine learning, visualization, and dimensionality reduction. To avoid them having to learn using our design, we conducted pair analytics [50], where domain experts and visualization designers jointly analyze data, while the designers serve as technical assistants.

We recorded screen and voice throughout the study with consent. The study concluded with a semi-structured interview to further inquire about general thoughts. On average, participants took 2.2 hours for the study. Due to space limits, we only discuss the main findings here, full details can be found in the supplemental material.

Results: All participants started with recording and editing a short melody. They were interested in how the AI would react to their melodies, especially P3: *"I was curious, so I put in some short and some long notes"*. Next, they generated some continuations for their melody.

Participants found our icicle plot helpful *"when generating many samples, [this] can help [to] see more quickly which samples are interesting and which are not"* (P4). We found that the icicle plot led to a faster selection of samples by just looking at them: *"I can see that these [points to two samples] are very interesting"* (P5). Especially P4 and P5 liked this representation due to the miniature piano rolls that helped P5 find certain intervals in melodies.

Participants used the node-link diagram to sort samples, inspect them, and select continuations. They liked sorting by metrics: *"I think it's good to sort by intervals [... it is] essential when selecting a melody with specific characteristics"* (P1), and found it helpful to use different metrics: *"I can imagine for complex music and many samples [...] and having an rough idea, [sorting] can help using parameters"* (P4), *"[When] composing with dissonance [...] sorting by dissonance would help"* (P3).

While composing, P3 repeatedly asked where recently added samples began within the composition. Our attribution visualization showed the difference between notes added by the AI and edited notes, which helped P3: *"where was the last note? [switching to attribution] This visual is very comfortable"* (P3).

We were specifically interested in how participants would select samples to interact with them. They often had some idea for what they were looking for, such as setting a contrast (P1), wanting *"to hear the most randomness"* (P2), or variations and less randomness (P4), where our visualization helped: *"[This] is the only one that goes down"* (P5). Participants sometimes did not like samples (*"I liked the first part, but the second was not good"* (P2)) and edited them after adding to the composition (P5) or even discarded all to *"change something on my own"* (P3).

As we also wanted to evaluate AI analysis, participants then generated 50 to 80 continuations to analyze with our visualizations. P1 found all glyphs helpful and told us these could *"extend the intuitive analysis of composers [... and] lead to a different style"* in composing. The participants were *"not used to look at the data"* (P2, P3) and these kinds of visualizations, but most learned quickly and found the scatterplot *"very interesting as an analysis tool, even independent of the AI"* (P1). Especially P1 was very interested in all the analysis possibilities and mentioned that to their knowledge, the lack of analysis is a drawback to algorithmic composition that works with randomness: *"I did see that really rarely and this is a huge shortcoming [...] when composing with the computer"* (P1).

Surprisingly, P1 told us how their professor answered the question of *"how to select the best samples?"*: *"You could generate as often as you like and listen to all and decide while listening intuitively"*. P1 complained about this strategy: *"You would listen to all and have no overview"*. Our scatterplot provides such an overview (P1).

[5] MusicRNN, magenta.github.io/magenta-js/music/classes/
[6] github.com/magenta/magenta-js

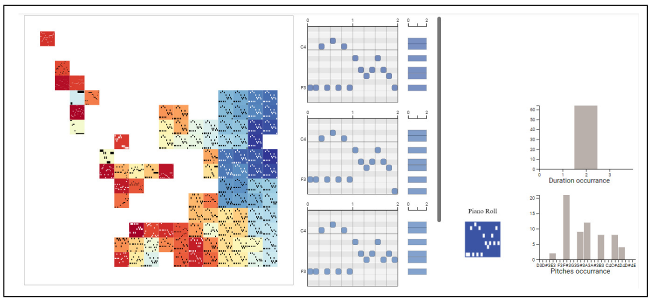

Figure 6. Similarity based scatterplot with piano roll glyphs colored by temperature (left). Selected samples are shown as piano rolls (center) and summarized as histograms (right).

We evaluated how users select neighborhoods in the scatterplot and found that the color (encoding temperature) impacted their decision. For example, P2 *"clicked there because of the color [...] ones with higher and lower temperature"* and P3 investigated *"how melodies vary"* based on temperature. Especially interesting to us was that P5 changed the similarity metric to rhythm only, to look for samples in a cluster with the original rhythm. P5 then clicked on an outlier sample: *"Why is this separate?"*.

The glyph representations were used by all participants. Piechart glyphs helped P3 select a melody sample, when they were looking for something calm that contrasts the composition. While analyzing, P5 searched for perfect fourths with the interval histogram glyphs, as their melody contained a lot of these. *"I am interested in finding melodies with perfect fourths [showing occurrences of interval glyph]. We can directly see [...] there is almost nothing"* (P5). Other participants used this glyph to select samples *"when I have a structure in mind [such as a rising line]"* (P1), which showed more positive intervals than negative, or to investigate the extreme melodies with many larger intervals. All participants used piano roll glyphs to compare melodic structures and find common pattern in clusters: *"I can see the same musical structure (melodic bows) [...] and further away it turns different"* (P1).

All participants found a relationship between temperature, shown through the color of piano rolls, and the structure (Figure 6): *"The blue ones have clearer structure while red ones are jumpier without that much connection"* (P1), *"I can see the randomness"* (P2). We found the piano roll to be more intuitive than other glyph types and therefore a good default, as it shows melodies directly: *"For me [the piano roll glyph] is very intuitive, I can directly imagine how they approximately could sound like"* (P4).

Especially P5 found that *"[correlation visualization] is fantastic [...]. This would make analysis much easier even without using the AI"*. After investigating some metric combinations, P5 surprised us by finding a pattern regarding similarity and temperature, where the correlation between an arbitrary metric and the similarity would always be the opposite to when combining the metric with temperature (Figure 4).

Our study concluded with short semi-structured interviews, summarized below. None of the participants preferred our approach over their current workflow – which we expected, as experts learned their workflow over years – but many could image using it for inspiration or certain tasks. All participants found the visualizations helpful for comparing, filtering, and selecting melody samples, especially for beginners, even when P2 had problems imaging the sound of a melody by its visuals. Most mentioned that it takes time to get used to the unfamiliar visualizations and metrics. Despite general scepticism against AI, all participants saw potential value in an AI-assisted approach.

5. LIMITATIONS

The sample relation visualizations do not scale for larger numbers of samples or levels, which could be addressed by not showing all samples at once but using scrolling or filtering. As alternative, the scatterplot can show many samples, but struggles with displaying similarities and complete glyphs. Using a gridified layout poses a trade-off between overlap and inexact positions.

Our example metrics and glyphs might not be optimal and miss some key characteristics. The current implementation is limited to monophonic melodies, but can be extended to polyphony through other AIs and metrics. Some glyphs only represent one aspect of music, but composers often need to consider multiple at once, such as pitch *and* time. For sorting, we only used statistical metrics of melodies, which are interesting for some experimental composers, but could be hard to interpret for beginners or other composers. These users would need more musical characteristics like mood, which is harder to calculate.

Another limitation of our approach is learnability, as users have to learn reading and interpreting a set of visual encodings. Based on our evaluation, we believe that regular users of our design can quickly get an intuition by looking at and listening to a range of samples.

6. CONCLUSION

As AI will likely not fully replace human composers in the foreseeable future [5], we advocate for an approach in which both, human and AI, cooperate. To this end, we propose a user-centered, interactive, and visual approach for iterative, AI-assisted music composition aimed at hobby musicians and experimental composers. We designed different visualizations as interface for interaction with generative models. Since these visualizations are AI-agnostic, a composer could use any kind of AI or general algorithm in combination with our approach.

In the future, we want to test different models [16, 22] to generate polyphonic music, harmonize melodies, extend metrics, and allow better steering. Analyzing the composition process could benefit from a history visualization that shows which notes were added or changed when and by whom. We further plan to explore aggregations and glyphs for more compact representations and extend our evaluation to more diverse composers, including beginners.

7. ACKNOWLEDGEMENTS

This work was funded by the Cyber Valley Research Fund and the Artificial Intelligence Software Academy (AISA).

8. REFERENCES

[1] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, “SampleRNN: An unconditional end-to-end neural audio generation model,” in *Proc. International Conf. Learning Representations (ICLR)*, 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1612.07837

[2] S. Oore, I. Simon, S. Dieleman, and D. Eck, “Learning to create piano performances,” in *NIPS Workshop on Machine Learning and Creativity*, 2017. [Online]. Available: https://nips2017creativity.github.io/doc/Learning_Piano.pdf

[3] D. Herremans, C.-H. Chuan, and E. Chew, “A functional taxonomy of music generation systems,” *ACM Comput. Surv.*, 2017. [Online]. Available: https://doi.org/10.1145/3108242

[4] C. Hernandez-Olivan and J. R. Beltran, “Music composition with deep learning: A review,” 2021. [Online]. Available: https://doi.org/10.48550/ARXIV.2108.12290

[5] S. Knotts and N. Collins, “A survey on the uptake of music AI software,” in *Proc. International Conf. New Interfaces for Musical Expression (NIME)*, 2020, pp. 499–504. [Online]. Available: https://doi.org/10.5281/zenodo.4813499

[6] T. Mejtoft, L. Lagerhjelm, U. Söderström, and O. Norberg, “Creative capabilities of machine learning: Evaluating music created by algorithms,” in *European Conf. Cognitive Ergonomics (ECCE)*, 2021, pp. 1–4. [Online]. Available: https://doi.org/10.1145/3452853.3452863

[7] C. A. Huang, H. V. Koops, E. Newton-Rex, M. Dinculescu, and C. J. Cai, “AI song contest: Human-AI co-creation in songwriting,” in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2020, pp. 708–716. [Online]. Available: https://doi.org/10.5281/zenodo.4245530

[8] R. Louie, J. Engel, and A. Huang, “Expressive communication: A common framework for evaluating developments in generative models and steering interfaces,” 2021. [Online]. Available: http://arxiv.org/abs/2111.14951

[9] F. Heyen, T. Munz, M. Neumann, D. Ortega, N. T. Vu, D. Weiskopf, and M. Sedlmair, “ClaVis: An interactive visual comparison system for classifiers,” in *Proc. International Conf. Advanced Visual Interfaces (AVI)*, 2020, pp. 1–9. [Online]. Available: https://doi.org/10.1145/3399715.3399814

[10] S. Bruckner and T. Möller, “Result-driven exploration of simulation parameter spaces for visual effects design,” *IEEE Trans. Visualization and Computer Graphics (TVCG)*, pp. 1468–1476, 2010. [Online]. Available: https://doi.org/10.1109/TVCG.2010.190

[11] K. Wongsuphasawat, J. A. Guerra Gómez, C. Plaisant, T. D. Wang, M. Taieb-Maimon, and B. Shneiderman, “LifeFlow: Visualizing an overview of event sequences,” in *Proc. Conf. Human Factors in Computing Systems (CHI)*, 2011, p. 1747–1756. [Online]. Available: https://doi.org/10.1145/1978942.1979196

[12] Z. Liu, B. Kerr, M. Dontcheva, J. Grover, M. Hoffman, and A. Wilson, “CoreFlow: Extracting and visualizing branching patterns from event sequences,” *Computer Graphics Forum (CGF)*, pp. 527–538, 2017. [Online]. Available: https://doi.org/10.1111/cgf.13208

[13] J.-P. Briot, G. Hadjeres, and F.-D. Pachet, *Deep learning techniques for music generation*. Springer, 2020. [Online]. Available: https://doi.org/10.1007/978-3-319-70163-9

[14] E. S. Koh, S. Dubnov, and D. Wright, “Rethinking recurrent latent variable model for music composition,” in *Proc. IEEE International Workshop Multimedia Signal Processing (MMSP)*, 2018, pp. 1–6. [Online]. Available: https://doi.org/10.1109/MMSP.2018.8547061

[15] G. Hadjeres and F. Nielsen, “Anticipation-RNN: Enforcing unary constraints in sequence generation, with application to interactive music generation,” *Neural Computing and Applications*, pp. 995–1005, 2020. [Online]. Available: https://doi.org/10.1007/s00521-018-3868-4

[16] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, and D. Eck, “A hierarchical latent vector model for learning long-term structure in music,” in *Proc. International Conf. Machine Learning (PMLR)*, 2018, pp. 4364–4373. [Online]. Available: https://doi.org/10.48550/arXiv.1803.05428

[17] A. Weber, L. N. Alegre, J. Torresen, and B. C. da Silva, “Parameterized melody generation with autoencoders and temporally-consistent noise,” in *Proc. International Conf. New Interfaces for Musical Expression (NIME)*, 2019, pp. 174–179. [Online]. Available: https://doi.org/10.5281/zenodo.3672914

[18] A. Pati, A. Lerch, and G. Hadjeres, “Learning to traverse latent spaces for musical score inpainting,” in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2019, pp. 343–351. [Online]. Available: http://doi.org/10.5281/zenodo.3527814

[19] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, “MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment,” in *Proc. AAAI Conf. Artificial*

Intelligence, vol. 32, no. 1, 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1709.06298

[20] L. C. Yang, S. Y. Chou, and Y. H. Yang, "MidiNet: A convolutional generative adversarial network for symbolic-domain music generation," in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2017, pp. 324–331. [Online]. Available: https://doi.org/10.5281/zenodo.1415990

[21] R. T. Dean and J. Forth, "Towards a deep improviser: a prototype deep learning post-tonal free music generator," *Neural Computing and Applications*, pp. 969–979, 2020. [Online]. Available: https://doi.org/10.1007/s00521-018-3765-x

[22] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, I. Simon, C. Hawthorne, A. M. Dai, M. D. Hoffman, M. Dinculescu, and D. Eck, "Music Transformer: Generating music with long-term structure," in *International Conf. Learning Representations (ICLR)*, 2019, pp. 1–13. [Online]. Available: https://doi.org/10.48550/arXiv.1809.04281

[23] J. A. T. Lupker, "Score-Transformer: A deep learning aid for music composition," in *Proc. International Conf. New Interfaces for Musical Expression (NIME)*, 2021. [Online]. Available: https://doi.org/10.21428/92fbeb44.21d4fd1f

[24] T. Nuttall, B. Haki, and S. Jorda, "Transformer neural networks for automated rhythm generation," in *Proc. International Conf. New Interfaces for Musical Expression (NIME)*, 2021. [Online]. Available: https://doi.org/10.21428/92fbeb44.fe9a0d82

[25] O. Lopez-Rincon, O. Starostenko, and G. A.-S. Martín, "Algoritmic music composition based on artificial intelligence: A survey," in *Proc. International Conf. Electronics, Communications and Computers (CONIELECOMP)*, 2018, pp. 187–193. [Online]. Available: https://doi.org/10.1109/CONIELECOMP.2018.8327197

[26] M. Suh, E. Youngblom, M. Terry, and C. J. Cai, "AI as social glue: Uncovering the roles of deep generative AI during social music composition," in *Proc. Conf. Human Factors in Computing Systems (CHI)*, 2021, pp. 1–11. [Online]. Available: https://doi.org/10.1145/3411764.3445219

[27] A. Roberts, J. Engel, Y. Mann, J. Gillick, C. Kayacik, S. Nørly, M. Dinculescu, C. Radebaugh, C. Hawthorne, and D. Eck, "Magenta Studio: Augmenting creativity with deep learning in Ableton Live," in *Proc. International Workshop on Musical Metacreation (MUME)*, 2019. [Online]. Available: https://doi.org/10.5281/zenodo.4285266

[28] C. A. Huang, C. Hawthorne, A. Roberts, M. Dinculescu, J. Wexler, L. Hong, and J. Howcroft, "The Bach Doodle: Approachable music composition with machine learning at scale." in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2019, pp. 793–800. [Online]. Available: http://doi.org/10.5281/zenodo.3527930

[29] T. Chakraborti, B. McCane, S. Mills, and U. Pal, "CoCoNet: A collaborative convolutional network applied to fine-grained bird species classification," in *35th International Conf. Image and Vision Computing New Zealand (IVCNZ)*, 2020, pp. 1–6. [Online]. Available: http://doi.org/10.1109/IVCNZ51579.2020.9290677

[30] R. Louie, A. Coenen, C. Z. Huang, M. Terry, and C. J. Cai, "Novice-AI music co-creation via AI-steering tools for deep generative models," in *Proc. Conf. on Human Factors in Computing Systems (CHI).* ACM, 2020, p. 1–13. [Online]. Available: https://doi.org/10.1145/3313831.3376739

[31] K. Chen, C. Wang, T. Berg-Kirkpatrick, and S. Dubnov, "Music SketchNet: Controllable music generation via factorized representations of pitch and rhythm," in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2020, pp. 77–84. [Online]. Available: http://doi.org/10.5281/zenodo.4245372

[32] Y. Tsuchiya and T. Kitahara, "A non-notewise melody editing method for supporting musically untrained people's music composition," *Journal Creative Music Systems (JCMS)*, 2019. [Online]. Available: https://doi.org/10.1007/BF02289565

[33] Y. Zhang, G. Xia, M. Levy, and S. Dixon, "COSMIC: A conversational interface for human-AI music co-creation," in *Proc. International Conf. New Interfaces for Musical Expression (NIME)*, 2021. [Online]. Available: https://doi.org/10.21428/92fbeb44.110a7a32

[34] C. A. Huang, D. Duvenaud, and K. Z. Gajos, "ChordRipple: Recommending chords to help novice composers go beyond the ordinary," in *Proc. International Conf. Intelligent User Interfaces (IUI)*, 2016, p. 241–250. [Online]. Available: https://doi.org/10.1145/2856767.2856792

[35] N. Privato, O. Rampado, and A. Novello, "A creative tool for the musician combining LSTM and Markov chains in Max/MSP," in *Proc. International Conf. EvoMUSART, Held as Part of EvoStar*, 2022, p. 228–242. [Online]. Available: https://doi.org/10.1007/978-3-031-03789-4_15

[36] R. Guo, I. Simpson, C. Kiefer, T. Magnusson, and D. Herremans, "MusIAC: An extensible generative framework for music infilling applications with multi-level control," in *Proc. International Conf. EvoMUSART, Held as Part of EvoStar*, 2022, pp. 341–356. [Online]. Available: https://doi.org/10.1007/978-3-031-03789-4_22

[37] T. Bazin and G. Hadjeres, "NONOTO: A model-agnostic web interface for interactive music composition by inpainting," in *Proc. International Conf. Computational Creativity (ICCC)*, 2019, pp. 89–91. [Online]. Available: http://doi.org/10.48550/ARXIV.1907.10380

[38] S. Park, T. Kwon, J. Lee, J. Kim, and J. Nam, "A cross-scape plot representation for visualizing symbolic melodic similarity," in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2019, pp. 423–430. [Online]. Available: https://doi.org/10.5281/zenodo.3527834

[39] M. Gotham and M. Ireland, "Taking form: A representation standard, conversion code, and example corpus for recording, visualizing, and studying analyses of musical form," in *Proc. International Society for Music Information Retrieval Conf. (ISMIR)*, 2019, pp. 693–699. [Online]. Available: https://doi.org/10.5281/zenodo.3527904

[40] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller, "Visual parameter space analysis: A conceptual framework," *IEEE Trans. Visualization and Computer Graphics (TVCG)*, pp. 2161–2170, 2014. [Online]. Available: https://doi.org/10.1109/TVCG.2014.2346321

[41] M. Ward, C.-h. Chen, W. K. Härdle, and A. Unwin, "Multivariate data glyphs: Principles and practice," in *Handbook of Data Visualization.* Springer, 2008, pp. 179–198.

[42] A. Klippel, F. Hardisty, and C. Weaver, "Star Plots: How shape characteristics influence classification tasks," *Cartography and Geographic Information Science (CaGIS)*, pp. 149–163, 2009. [Online]. Available: https://doi.org/10.1559/152304009788188808

[43] O. Gomez, K. K. Ganguli, L. Kuzmenko, and C. Guedes, "Exploring music collections: An interactive, dimensionality reduction approach to visualizing songbanks," in *Proc. International Conf. Intelligent User Interfaces (IUI)*, 2020, p. 138–139. [Online]. Available: https://doi.org/10.1145/3379336.3381461

[44] M. Bostock, V. Ogievetsky, and J. Heer, "D^3: Data-driven documents," *IEEE Trans. Visualization and Computer Graphics (TVCG)*, vol. 17, no. 12, pp. 2301–2309, 2011. [Online]. Available: https://doi.org/10.1109/TVCG.2011.185

[45] R. Cutura, C. Kralj, and M. Sedlmair, "Druid$_{JS}$ — a javascript library for dimensionality reduction," in *Proc. IEEE Visualization Conf. (VIS)*, 2020, pp. 111–115. [Online]. Available: https://doi.org/10.1109/VIS47514.2020.00029

[46] J. B. Kruskal, "Nonmetric multidimensional scaling: A numerical method," *Psychometrika*, pp. 115–129, 1964. [Online]. Available: https://doi.org/10.1007/BF02289694

[47] ——, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, pp. 1–27, 1964. [Online]. Available: https://doi.org/10.1007/BF02289565

[48] R. Cutura, C. Morariu, Z. Cheng, Y. Wang, D. Weiskopf, and M. Sedlmair, "Hagrid — gridify scatterplots with Hilbert and Gosper curves," in *International Symp. Visual Information Communication and Interaction (VINCI)*, 2021, pp. 1–8. [Online]. Available: https://doi.org/10.1145/3481549.3481569

[49] R. A. Moog, "MIDI: Musical instrument digital interface," *Journal Audio Engineering Society (AES)*, pp. 394–404, 1986. [Online]. Available: https://doi.org/10.4324/9780203484272-524

[50] R. Arias-Hernandez, L. T. Kaastra, T. M. Green, and B. Fisher, "Pair analytics: Capturing reasoning processes in collaborative visual analytics," in *Hawaii International Conf. System Sciences (HICSS)*, 2011, pp. 1–10. [Online]. Available: https://doi.org/10.1109/HICSS.2011.339

RETRIEVING MUSICAL INFORMATION FROM NEURAL DATA: HOW COGNITIVE FEATURES ENRICH ACOUSTIC ONES

Ellie Bean Abrams[1,2,3] **Eva Muñoz Vidal**[1,2,3] **Claire Pelofi***[1,2] **Pablo Ripollés***[1,2,3]

[1] Music and Audio Research Laboratory, New York University
[2] Center for Language, Music, and Emotion, New York University
[3] Department of Psychology, New York University
* denotes shared last authorship

`{ea84,elm8254,cp2830,pr82}@nyu.edu`

ABSTRACT

Various features – from low-level acoustics, to higher-level statistical regularities, to memory associations – contribute to the experience of musical enjoyment and pleasure. Recent work suggests that *musical surprisal*, that is, the unexpectedness of a musical event given its context, may directly predict listeners' experiences of pleasure and enjoyment during music listening. Understanding how surprisal shapes listeners' preferences for certain musical pieces has implications for music recommender systems, which are typically content- (both acoustic or semantic) or metadata-based. Here we test a recently developed computational algorithm, called the Dynamic-Regularity Extraction (D-REX) model, that uses Bayesian inference to predict the surprisal that humans experience while listening to music. We demonstrate that the brain tracks musical surprisal as modeled by D-REX by conducting a decoding analysis on the neural signal (collected through magnetoencephalography) of participants listening to music. Thus, we demonstrate the validity of a computational model of musical surprisal, which may remarkably inform the next generation of recommender systems. In addition, we present an open-source neural dataset which will be available for future research to foster approaches combining MIR with cognitive neuroscience, an approach we believe will be a key strategy in characterizing people's reactions to music.

1. INTRODUCTION

Musical surprisal, or, the relative expectations listeners have of ongoing musical events, is essential in understanding humans' engagement and experience with music [1]. Decades of theoretical and experimental work have defined the study of expectation and surprisal as cognitive processes, often in the context of language processing [2, 3]. Importantly, this line of inquiry has crucial implications for the study of music preference and pleasure [4,5]. Recent studies have shown that the relationship between information-theoretic measures such as surprisal, entropy, or complexity, and enjoyment and pleasure may be described by an inverted U-shape curve (also referred to as the Wundt effect) where stimulus enjoyment is enhanced with an increase in complexity of the song. But as surprisal increases to higher levels, its effect becomes unpleasant [4, 6]. The perceptual measures (e.g. complexity, familiarity/novelty, surprisal) that have been shown to modulate musical preferences are referred to as *collative* variables [7–9]. Supposedly, when other high-level variables are controlled, collative variables explain a large portion of variance in listeners' musical preference [8, 10] following the aforementioned parabolic function. The "sweet spot" of this inverse U-shaped curve may shift left or right, with respect to surprisal measures, depending on factors such as personality, openness-to-experience, or genre preferences [9, 11]. While most music recommender systems rely on acoustic (extracted from a user's music library) and semantic features (derived from subjective behavioral ratings) to predict listeners' preferences, the use of cognitive measures such as music surprisal can remarkably improve their performance, especially because these are known to accurately predict musical pleasure. The results presented here suggest that cognitive neuroscience methods can be leveraged to elucidate new ways of extracting information from music and better predict user preferences.

Early theoretical work on bottom-up and top-down processes modulating expectations by Leonard Meyer and Eugene Narmour [12–14] brought about efforts to model musical prediction, evolving more concretely into explorations of musical tension [15, 16], entropy [17], and the neural bases of surprisal [18, 19]. Crucially, computational models may be tested against both subjective behavioral and objective neurophysiological measures in order to provide a deeper understanding of whether – and how precisely – information-theoretic measures of music inform the cognitive processes underlying music perception and enjoyment. The Information Dynamics of Music (IDyOM) model [20] generates variable-order Markov probability distributions for each note in a melodic sequence by extracting statistics from both a music corpora and a short-term musical context, thereby incorporating long-term (top-down) and short-term (bottom-up) musical regularities. The IDyOM model has been shown to predict behavioral and physiological markers of listeners' expectations [21–25] and has recently been related to brain activity, showing that melodic expectation is also directly encoded in the neural signal [26–31]. While IDyOM is a well-validated, efficient, and widely used computational model of musical surprisal, it operates only on symbolic (MIDI) data and requires a training set of stimuli (the long-term component of the model) to generate predictions.

To circumvent these shortcomings and explore the behavioral and neural response to continuous audio signals, we turn to a computational model of surprisal recently developed by Skerritt-Davis and Elhilali at Johns Hopkins, the Dynamic Regularity Extraction (D-REX) model [32, 33]. D-REX uses a Bayesian framework to generate predictions and was originally designed to evaluate prediction errors over time for stochastic sound sequences. This model is relevant to the MIR community, as it can be run on any continuous audio input, which broadens its usability for the analysis of large, diverse collections of music. Our previous behavioral results have validated D-REX as predictive of subjective ratings of surprisal, showing that surprisal as calculated by D-REX predicted subjective behavioral surprisal ratings for 80 music excerpts [34]. Given the important role that prediction plays in musical pleasure, enjoyment, and engagement [4, 6, 35], D-REX is used here to explore whether *the brain* tracks musical surprisal. Concretely, we go beyond subjective behavioral responses and test whether musical surprisal as calculated by D-REX is directly represented in an objective neurophysiological signal. Specifically, we present an experimental work in which we recorded brain activity using magnetoencephalography (MEG) while twenty participants listened to musical excerpts. We then relate the neural signal to the D-REX model output using a decoding algorithm to determine whether surprisal is represented at the brain level.

2. DATA COLLECTION AND PROCEDURE

Twenty participants with self-reported normal hearing completed the experiment (11 female, 24.8 ± 2.9 years of age). Participants were presented with 30 one-minute-long musical excerpts (described in Section 3.1) while their brain activity was recorded using MEG. Participants began each trial by clicking a button to start playing each musical excerpt. At the end of each excerpt, participants moved to the next stage where they provided ratings across five measures using a 4-point scale (1 lowest to 4 highest): pleasure, valence, recognition, familiarity, and surprisal.

MEG measures the magnetic fields generated by the electrical activity of neurons in the brain. Unlike electrical currents captured by EEG devices, magnetic activity can pass through the cortex and skull without distortion, resulting in higher spatial resolution [36]. Continuous MEG data was collected using a 157-channel axial gradiometer system at NYU, at a sampling rate of 1000Hz with an online low-pass filter of 200Hz. Prior to conducting the main analysis, MEG data underwent preprocessing, starting with the noise-reduction of the signal using the continuously adjusted least squares method (CALM) with the MEG160 software [37]. The data was then exported into MNE-Python [38] and bad channels (e.g., channels which saturated during the recording) were removed through visual inspection and interpolated using a weighted sum of signals from neighboring channels. An independent component analysis (ICA) was fitted on the data using FastICA in MNE-Python to isolate independent sources of noise contaminating the channels. Components corresponding to system noise, heartbeat, and eye-blinks were removed from the raw recording after inspection of the topography and time-course of magnetic activity for each component. Finally, epochs were extracted from one second before stimulus onset to stimulus offset, resulting in 61-second-long epochs. For an additional data quality check, we inspected each participant's auditory response to a set of randomized 1000Hz tones and 250Hz tones and observed a higher amplitude response to the higher tone, thus confirming satisfactory data quality.

3. STIMULI

The musical stimuli used here have been previously used to validate D-REX as a predictor of behavioral subjective ratings of surprisal using different genres of music (classical and elevator music) [34]. Classical stimuli were taken from a list of musical excerpts rated by 65 participants for pleasantness in a previous study [39]. The other music stimuli were selected from a range of sources, including songs from Muzak Orchestra's Stimulus Progression albums, as well as more contemporary elevator music compositions (see Supplementary Table S1 [1] for a list of stimuli). The most interesting minute (highest accumulated surprisal) of each piece was selected using a shifting window of 60s across D-REX's surprisal output, so that any results showing lower correlations with brain activity would not simply be due to choosing a particularly unsurprising minute of the piece (see [34]). All excerpts were normalized to 70dB using Praat and python's AudioSegment package, and the sound faded 3s in 3s out.

4. MODELING MUSICAL SURPRISE

4.1 Acoustic Features

D-REX takes as input a set of acoustic features, which can be extracted using any existing MIR techniques. In our case, we extracted features using the NSL Auditory-Cortical Matlab Toolbox developed by the Neural Systems Laboratory at the University of Maryland. The toolbox is

[1] Supplementary materials may be downloaded at `https://osf.io/dbm49/`.