IEIE Transactions on Smart Processing and Computing, vol. 12, no. 4, August 2023
https://doi.org/10.5573/IEIESPC.2023.12.4.329

VirtuosoTune: Hierarchical Melody Language Model

Dasaem Jeong

Department of Art & Technology, Sogang University / Seoul, Korea dasaemj@sogang.ac.kr

Received March 7, 2023; Revised April 1, 2023; Accepted April 2, 2023; Published August 30, 2023

** Regular Paper*

***Abstract*:** The Ai Music Generation Challenge is a competition that evaluates music generation systems. The participants submit a model capable of generating a specific genre of traditional music, which human experts then evaluate. The 2022 competition aimed to generate the most plausible reel, a type of Irish traditional dance music. This paper presents our submitted model, VirtuosoTune, which utilizes musically-structured encodings and the hierarchical structure of gated recurrent units. One of the tunes generated by the model was awarded the first prize, achieving perfect scores from all four judges.

***Keywords*:** Artificial intelligence, Music generation, Language model, Hierarchical RNN, Folk music

1. Introduction

Can a machine generate novel and plausible music? This has been a topic of interest in artificial intelligence for several decades, and recent advances in deep learning have accelerated research in this area. While large generation models, such as Jukebox [1] and MusicLM [2], have shown remarkable results in audio generation and generate raw waveforms directly, many music generation models focus on symbolic music, which generates musical symbols, such as a sequence of discrete note events. For example, DeepBach [3] generates four-voice chorales in the style of J.S. Bach. Music Transformer [4] was trained on Bach's chorales and piano music, and the model was applied to different genres of music, such as game soundtracks [5].

One of the crucial limitations of research in music generation is the evaluation of the generated results. For a quantitative evaluation, negative log-likelihood or perplexity is used widely [3, 5, 6], which evaluates how good the model is at predicting a ground-truth upcoming token for given preceding tokens. On the other hand, the high likelihood cannot guarantee the quality of the generation result [7]. Therefore, a qualitative evaluation through human listening tests is considered the golden standard for evaluating music generation models. These listening tests can take various forms, such as asking whether the generated music sounds human-made in the style of the Turing test [3, 5] or comparing multiple generated pieces to determine which is preferable [4, 6].

The problem with the listening test is that the evaluation is entirely subjective. The musical background of the participants can vary, which can affect the result. In addition, it is difficult for them to recognize whether the model strictly imitates the training set or, in other words, plagiarizes because the participants cannot be familiar with the entire training set.

In this context, *Ai Music Generation Challenge* has been hosted annually since 2020, with three main goals: "1) to promote meaningful approaches to evaluating music Ai; 2) to see how music Ai research can benefit from considering traditional music, and how traditional music might benefit from music Ai research; 3) to facilitate discussions about the ethics of music Ai research applied to traditional music practices." [8] The task was to build a system that generates the most plausible tune of specific traditional music, which was double jig, an Irish traditional dance form in 2020, and slängpolska, a Scandinavian traditional dance form in 2021.

The challenge was assessed by proficient judges in traditional music, offering more detailed feedback than other evaluation methods. The judges' familiarity with the attributes of the target music style and its collection enhances the evaluation's credibility.

This paper presents the proposed melody language model submitted for the *Ai Music Generation Challenge 2022,* of which the goal was to generate the most plausible reel, Irish traditional dance music. The model generated a tune that won the first-prize award from a panel of human expert judges. The design of this model, called VirtuosoTune, was derived from our previous neural network system for expressive piano performance modeling, VirtuosoNet [9].

2. Related Work

Symbolic music generation can be approached in a similar manner to natural language processing by converting music scores into plain text format or a sequence of tokens. The ABC notation is a notable example of a text-like encoding of music scores, used widely for notating traditional folk music. The notation represents each note and musical symbol in human-readable text, including bar lines and repetition marks.

Sturm et al. [10] reported that the language model using long short-term memory (LSTM) could be trained on the ABC notation dataset. The model was later progressed as a *folk-rnn* [11] and employed as a benchmark model for *Ai Music Generation Challenge 2020*.

Casini and Sturm proposed a next version of *folk-rnn*, which used a transformer decoder instead of LSTM [12]. The model was trained in Irish folk music and fine-tuned in Scandinavian music. The fine-tuned model was submitted to the *Ai Music Generation Challenge 2021* and achieved the best evaluation score in sum from human expert judges.

3. The Proposed System

3.1 Training Data

The pre-processed ABC dataset of Irish traditional music called *folk-rnn data_v_3,* used for previous research, was used for model training [11, 12]. The ABC data was crawled from *The Session*, a community-driven website featuring transcriptions of Irish traditional music.

The pre-processed dataset transposed each tune to C and C# key. Among them, only the tunes in the C key were used. Any slurs or ornamentation were omitted during pre-processing. Although the pre-processed dataset lacked rhythmic genre information, such as ‘reel’ or ‘jig’, information was added by matching the tunes with those in *The Session*.

Among the dataset, we filtered out tunes with duration errors such as overfilled measure duration (e.g., nine 8th notes in 4/4 measure) or pickup measures without following corresponding pairs. Tunes with multiple time signatures or key signatures were excluded. Of the 22,925 tunes in the dataset, 18,612 tunes were retained after filtering. Furthermore, a few errors in repetition symbols, such as a missing seconda volta, were manually corrected.

3.2 Tokenizing

Based on the *pyabc* library, the ABC parser was implemented in parse repetition, and the ABC notation was tokenized.

When a token represents a note or a rest, the token has two features: pitch and duration. On the other hand, when a token did not represent a note or a rest, such as a barline, its duration was set to <pad>, and its pitch was set to the string of symbols. This is one of the key differences between the previous models [11, 12], which employ duration text as independent tokens. Using duration text as independent tokens results in a note being encoded with either one or two tokens because the ABC notation omits the duration character if the duration is equal to the unit length, with a duration value of 1. For example, “C2DE/E/” represents C with a duration of 2 and D with a duration of 1, and two E with a duration of 0.5. In our encoding scheme, each note is always encoded as a single token.

Other characters, such as barline “|”, repetition mark “:|”, and triplet mark “(3” are also grouped as a single token. Metadata, such as key information “K: C Major” were all handled as a single word in a vocabulary, without splitting the characters.

3.3 Encoding

Previous studies on Irish music using ABC notation, such as those by Sturm et al. [11, 12], used similar encoding methods to natural language processing, where the input sequence $\boldsymbol{X}$ is represented as $X \in \mathbb{W}^T$, with T being the number of tokens. Each token is represented solely by its index in a unified vocabulary, encompassing all the distinct musical symbols.

The generated results of previous work [11, 12] showed that deep neural networks could learn the meaning and semantic relationship between tokens in these previous encoding schemes. However, the encoding of features and the structure of the musical information in each token can have a significant impact on the performance of the model, as highlighted by Peracha [13] and Zeng et al. [14]. Considering the limited training set, this study aimed to encode tokens in a musically structured format with rich features.

We employ musically-structured encoding schemes as similar to previous research on handling symbolic music with neural network, such as expressive performance modeling [15] or symbolic music generation [14], to enrich the tokens with a more comprehensive set of features. This approach provides each note token with the pitch, duration, and several other features, including bar position, metrical position, time signature, or key signature.

Using this encoding method, the input sequence $\boldsymbol{X}$ can be expressed as $\boldsymbol{X} \in \mathbb{W}^{T \times F}$, where F is the number of feature categories. Table 1 provides a list of features.

A straightforward approach to encoding musical information is to assign a separate category to every unique numeric value. For example, if there were 25 different pitches from MIDI pitch 60 (C4) to 84 (C6), each pitch can be considered an independent "word" that is unrelated to any other pitch. Nevertheless, this approach disregards musical knowledge, such as the fact that pitches with the octave relation share similar characteristics, such as 60 (C4), 72 (C5), and 84 (C6). By encoding pitch as a combination of the pitch class (e.g., C, G) and octave (e.g., 4, 5, and 6), these notes can be grouped in the pitch class feature and represented by the same embedding. Hence, the MIDI pitch was also included as a distinct feature so the model could learn individual embeddings for each pitch and concatenate them with other feature embeddings.

In addition, the measure position of the current token was included as a feature. This encoding captures the

Table 1. List of Encoding Features.

Feature Name	Description
Pitch or Text	MIDI pitch of a note or text of the musical symbol
Duration	duration of note in unit length
Offset from repeat start	number of measures passed after the repetition start or beginning of the piece
Token offset in measure	accumulated duration of preceding notes in the measure
Offset modulo 4	the measure's offset from repeat start to modulo 4
Is on beat	whether the token locates on the beat
Is on a stronger beat	whether the token locates on the strong beat that is not downbeat
Pitch class	pitch class of the note
Octave	octave index of the note
Key	the tonal key signature of the piece
Meter	the time signature of the piece
Rhythm genre	rhythmic genre of the piece, such as reel or jig
Numerator TS	numerator of time signature
Denominator TS	denominator of time signature
Is compound meter	whether the time signature is compound meter (e.g., 6/8, 9/8)
Is triple meter	whether the time signature is in triple (e.g., 3/4, 9/8)

token offset from the start of the measure and the measure's offset from the start of the repeat. The measure offset from the repeat start provides valuable information to the model for learning typical 4 or 8-bar phrase patterns because repetition often occurs at the 4th or 8th bar in Irish traditional music. A modulo 4 value of the measure offset was included as an independent feature for each token to enhance the model understanding of a typical 4-bar phrase. This additional feature enables the token to convey information about its position within a 4-bar phrase.

3.4 Hierarchical GRU Language Model

Our model VirtuosoTune is trained in the form of a language model, which predicts the probability distribution of the next token x_t given the preceding tokens $x_0,\ldots,x_{t-1}$. The model is trained to minimize the negative log-likelihood over a sequence of tokens $x_{1:T}$, defined as $-\sum_{t=1}^{T}\log p\left(x_t|x_{0:t-1}\right)$, where T is the length of the sequence. The negative log-likelihood measures how well the model predicts the ground-truth sequence of tokens.

One of the key aspects of Irish traditional music is its clear measure structure. In addition, ABC notation always includes a bar (measure) boundary as a barline character "|", so the bar boundary is provided. To exploit this information, instead of using a single recurrent neural network (RNN), we employed a hierarchical RNN, which was also used for VirtuosoNet, a system for modeling

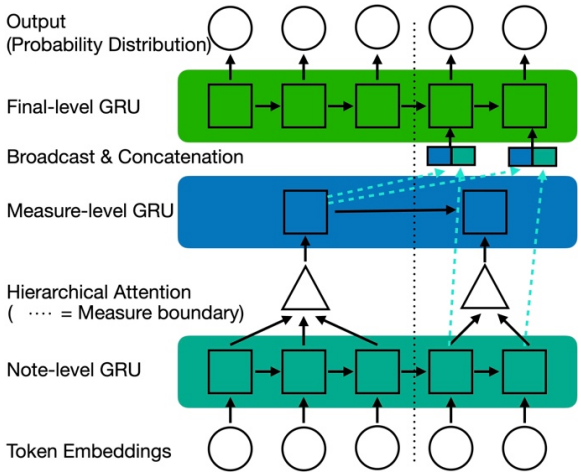

Fig. 1. Diagram of the proposed hierarchical GRU.

expressive piano performance from a given music score [9, 16]. VirtuosoNet structure was modified into a melody language model, VirtuosoTune, with the gated recurrent unit (GRU) [17] as an RNN cell, because GRU showed better performance than vanilla RNN or long short-term memory in these experiments. Three GRUs, a note-level GRU, a measure-level GRU, and a final GRU, were used to compose hierarchical structures, as shown in Fig. 1.

First, the input sequence $X \in \mathbb{W}^{T\times F}$ was transformed into a sequence of the dense vector by concatenating the embedding vector of each feature in Table 1. For each feature f there is a corresponding embedding layer $E_f \in \mathbb{R}^{v_f \times d_f}$, where v_f and d_f represent the vocabulary size and embedding size of feature f, respectively.

The token-level embeddings were fed into note-level GRU. The output of note-level GRU was summarized in each measure using context attention [18]. For a given sequence of note-level hidden states of single measure $\boldsymbol{h}=\left[\mathrm{h}_0, \mathrm{h}_1, \ldots, \mathrm{h}_t\right]$, hierarchical context attention summarized it as a $\boldsymbol{y}=\sum_{t}^{T}\alpha_t h_t$, where $\alpha_t = \mathrm{Softmax}\left(\tanh\left(\boldsymbol{W}\mathrm{h}_t+\mathrm{b}\right)^{\top}\mathrm{c}\right)$, and $\boldsymbol{W}$, b, and c represent the weight, bias, and context vector of the attention module, respectively, which are trainable parameters.

The measure-level summarizations were fed into the measure-level GRU. The measure-level output was then broadcasted into the note level so that each note or token's hidden output was concatenated with the measure-level output of the previous measure. The final GRU took the concatenated vectors as its input and returned the hidden states, which were then projected to the logit for the entire vocabulary of pitch and duration. The softmax was applied separately for pitch and duration.

4. Training

The model was trained with batch size 32, Adam optimizer of learning rate 0.0003, step learning rate

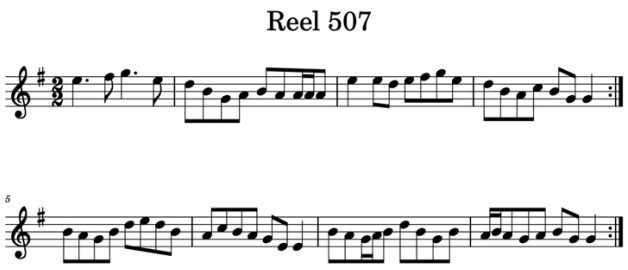

Fig. 2. Score of Reel 507, which won first prize.

scheduler with step size 10,000, and multiplication ratio of 0.5. The dataset was split into 19:1 so that only every 20th tune was used for the validation to include as many tunes as possible in the training set. A test split, the same as in previous research on the same task, was not used because the generation quality of the model was evaluated manually [11].

Every tune was trained in its entirety, from the first note to the last note. The loss function was the negative log-likelihood, which was calculated separately for the predicted probability of pitch and duration of each token. The duration loss was only calculated for tokens with duration, excluding non-note tokens, such as the barline or repetition.

Several models were trained with different architectures, encoding schemes, and hyperparameters. The final model was selected based on the validation loss and validation accuracy, and manual monitoring of the generated results. The final submitted model consisted of a note-level GRU with three layers, a measure-level GRU with three layers, final GRU with three layers, all with a 512 hidden size. The model was trained with RTX A6000 GPU for 100,000 iterations.

5. Evaluation at Ai Music Generation Challenge 2022

One thousand tunes generated by this model were submitted to the *Ai Music Generation Challenge 2022*. Among these, one tune was selected manually by us, and nine other tunes were chosen randomly for evaluation. Four expert judges in Irish traditional music evaluated the tunes based on various criteria. The evaluation process was comprised of two stages. First, the judges checked for plagiarism or any rhythm or mode that deviated from the characteristics of an Irish reel. Second, the judges assessed the plausibility of the melody and structure on a scale of 1 to 5.

5.1 Generating Tunes

The model generated a tune using various random seeds and fixed metadata conditions, including a meter of 4/4, C major, and reel rhythm. Sampling strategies, such as beam search or nucleus sampling, were not used, unlike previous work [12]. Following the generation process, basic rules were applied to ensure the generated tune adhered to valid measure durations, repeats, structure, rhythm, and cadence. The generation of tunes was continued until the model had 1000 tunes.

A rule-based post-processing technique was applied to transpose the generated tune. The transposed key was determined by the pitch range of the generated tune while imitating the key distribution among the 350 reels in "The Dance Music of Ireland: O'Neill's 1001".

5.2 Result

The evaluation result was presented on the challenge site [19], where the submission was named Clare. Among the ten tunes of the submission selected for evaluation, three judges flagged one as plagiarized and another as having a rhythm that did not meet the Irish reel standards. These results indicate room for improvement in the model performance.

On the other hand, the tune titled "Reel 507", presented in Fig. 2 received perfect scores (five out of five) from all four judges and was nominated as a potential award winner. Ultimately, this tune was selected as the first-prize winner.

The judges' remarks for "Reel 507" were as follows [19]: "Easy to remember, easy to follow and catch; uplifting, bright and fun to dance to"; "Excellent! Everything right! That's a 5+. Not plagiarised as far as I can see (I suspected it might be since it was so good)"; "Very simple reel, but adheres to all the variables regarding reel structure. Great Rhythm. Makes good use of repetition. Very simple but consistent and phrases are easy to remember. Excellent simple reel that sits well with the tradition."

As the evaluation comments showed, the generated tune proved that the AI model could produce a novel tune that was both novel and well-suited to the characteristics of traditional music. Among the *Ai Music Generation Challenge* from 2020 to 2022, this tune was the first to obtain perfect scores from all the human judges.

6. Conclusion

This paper proposed a music language model that uses musically-structured encoding schemes and hierarchical GRU. The human evaluation showed that the model could create a novel and interesting musical piece that follows specific musical traditions and characteristics, scoring perfect scores from all the judges. This significant achievement demonstrated the potential of deep learning models in music generation, particularly for traditional music. The next goal is to apply the VirtuosoTune to other genres of music, such as traditional Korean music, and to apply transformer architecture to enhance the long-term modeling of music sequences. The code, data, and the pre-trained weights of the submitted model have been released: https://github.com/jdasam/VirtuosoTune

Acknowledgements

This work was supported by Basic Science Research

Program through the National Research Foundation of Korea (NRF) funded by the Korea Government (MSIT) (NRF-2022R1F1A1074566). The implementation of data cleaning, parsing, and discussion was done together with students in Music and Art Learning (MALer) Lab of Sogang University: Hyerin Kim, Danbinaerin Han, Dasol Lee, Jiwoo Ryu, Daewoong Kim, Jongmin Jung, Joohye Kim, and Jiin An.

References

[1] P. Dhariwal, et al., "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020, Article (CrossRef Link)

[2] A. Agostinelli et al., "MusicLM: Generating music from text," *arXiv preprint arXiv:2301.11325, 2023,* Article (CrossRef Link)

[3] G. Hadjeres et al., "DeepBach: a steerable model for Bach chorales generation," in *Proc. of International Conference on Machine Learning*. PMLR, 2017, pp. 1362-1371, Article (CrossRef Link)

[4] C.-Z. A. Huang et al., "Music transformer," in *Proc. of International Conference on Learning Representations*, 2019. Article (CrossRef Link)

[5] C. Donahue et al., "LakhNES: Improving multi-instrumental music generation with cross-domain pre-training," in *Proc. of International Society for Music Information Retrieval Conference,* 2019. Article (CrossRef Link)

[6] B. Yu et al., "Museformer: Transformer with fine- and coarse-grained attention for music generation," in *Advances in Neural Information Processing Systems,* 2022. Article (CrossRef Link)

[7] L. Theis et al., "A note on the evaluation of generative models,", in *Proc. of International Conference on Learning Representations*, 2016. Article (CrossRef Link)

[8] B. L. Sturm and H. Maruri-Aguilar, "The Ai Music Generation Challenge 2020: Double jig in the style of O'Neill's 1001," *Journal of Creative Music Systems*, vol. 5, pp. 1-26, 2021. Article (CrossRef Link)

[9] D. Jeong et al, "VirtuosoNet: A hierarchical RNN-based system for modeling expressive piano performance," in *Proc. of 20th International Society for Music Information Retrieval Conference*, 2019. Article (CrossRef Link)

[10] B. Sturm et al., "Folk music style modeling by recurrent neural networks with long short term memory units," in *Late Breaking/Demo from 16th International Society for Music Information Retrieval Conference*, 2015. Article (CrossRef Link)

[11] B. L. Sturm et al., "Music transcription modeling and composition using deep learning", in *Proc. of 1st Conference on Computer Simulation of Musical Creativity*, 2016. Article (CrossRef Link)

[12] L. Casini and B. Sturm, "Tradformer: A transformer model of traditional music," in *Proc. of International Joint Conference on Artificial Intelligence*, 2022. Article (CrossRef Link)

[13] O. Peracha, "Improving polyphonic music models with feature-rich encoding," in *Proc. of International Society for Music Information Retrieval Conference*, 2020. Article (CrossRef Link)

[14] M. Zeng et al., "MusicBert: Symbolic music understanding with large-scale pre-training," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, 2021. Article (CrossRef Link)

[15] D. Jeong et al., "Score and performance features for rendering expressive music performances", in *Proc. of Music Encoding Conference*, 2019. Article (CrossRef Link)

[16] D. Jeong et al., "Graph neural network for music score data and modeling expressive piano performance," in *Proc. of International Conference on Machine Learning.* PMLR, 2019, pp. 3060-3070. Article (CrossRef Link)

[17] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation, in *Proc. of Empirical Methods in Natural Language Processing,* 2014. *Article (CrossRef Link)*

[18] Z. Yang et al., "Hierarchical attention networks for document classification", in *Proc. of the 2016 conference of the North American chapter of the association for computational linguistics: human langugage technologies*, 2016. pp. 1480-1489. Article (CrossRef Link)

[19] B. Sturm, "The Ai Music Generation Challenge 2022." [Online]. Available: https://github.com/boblsturm/aimusicgenerationchallenge2022

Dasaem Jeong is currently working as an Assistant Professor in the Department of Art & Technology at Sogang University in South Korea since 2021. Before joining Sogang University, he worked as a research scientist in T-Brain X, SK Telecom from 2020 to 2021. He obtained his Ph.D. and M.S. degrees in culture technology, and B.S. in mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST). His research primarily focuses on a diverse range of music information retrieval tasks, including expressive performance modeling, symbolic music generation, and cross-modal generation.