Using Machine Learning to Support Pedagogy in the Arts

Dan Morris
Microsoft Research
dan@microsoft.com

Rebecca Fiebrink
University of Washington
Princeton University
fiebrink@princeton.edu

Abstract
Teaching artistic skills to children presents a unique challenge: high-level creative and social elements of an artistic discipline are often the most engaging and the most likely to sustain student enthusiasm, but these skills rely on low-level sensorimotor capabilities, and in some cases rote knowledge, which are often tedious to develop. We hypothesize that computer-based learning can play a critical role in connecting "bottom-up" (sensorimotor-first) learning in the arts to "top-down" (creativity-first) learning, by employing machine learning and artificial intelligence techniques that can play the role of the sensorimotor expert. This approach allows learners to experience components of higher-level creativity and social interaction even before developing the prerequisite sensorimotor skills or academic knowledge.

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Keywords
Machine learning, education, creativity

ACM Classification Keywords
H5.m. Information interfaces and presentation (HCI): Miscellaneous, K.3.1. Computer Uses in Education.

Introduction
Artists—from hobbyists to professionals, in virtually all artistic disciplines—employ both high-level creative and lower-level sensorimotor skills in their work. Most artists will report that they derive their excitement from high-level creative thinking, and that this is the level on which artists collaborate and converse with other artists. However, these skills depend on a base of sensorimotor skills, and in some cases rote knowledge, that often fade into subconscious thinking as an artist progresses.

This presents a unique challenge for education in the arts: a guitarist generally needs to learn basic fingering patterns, which is often tedious and frustrating, before she can even engage in the truly creative or social aspects of musicianship. This challenge is magnified when the student is a child and may be less easily motivated by long-term goals or by friends or colleagues who have developed their skills to the point of long-term value and enjoyment. This challenge is exemplified by a common trend within music education:

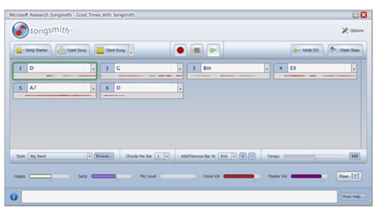

Figure 1. Songsmith automatically generates chords and accompaniment for vocal melodies, allowing musical novices to create original music by singing.

many children abandon instrumental education even after years of formal training in scales and technique, before the connections to creativity, social interaction, and "fun" are ever drawn.

We hypothesize that computer-based learning can play a critical role in connecting "bottom-up" (sensorimotor-first) learning in the arts to "top-down" (creativity-first) learning, by employing machine learning and artificial intelligence techniques that can play the role of the sensorimotor expert. This approach allows learners to experience components of higher-level creativity and social interaction even before developing the prerequisite sensorimotor skills or academic knowledge.

For example, a painting module might allow a student to explore scene composition, driven by an algorithmic system for rendering brush strokes from high-level instructions, while still learning about brush strokes and developing the fine motor control required for accurate painting. A program for teaching songwriting might allow a student to control the high-level variables that songwriters often contemplate (e.g. melodic arc, mood, dynamics), driven by an algorithmic system that has learned a mapping from these variables to low-level musical elements, while still developing motor skills on an instrument and a basic academic knowledge of harmony and chord theory. In each of these cases, the scaffolding provided by the computer is not a replacement for hard work; the learner is still constrained by the specific support provided. Rather, the computer enables learning and practice of high-level creative thinking in parallel with the learning of lower-level concepts, and serves as a motivational tool to keep students engaged.

This approach is employed regularly in other disciplines where no computer intelligence is necessary: teaching computer programming, for example, increasingly relies on human-created "skeleton code" that lets a student's very first computer program produce a rich, graphical, interactive system, leading to more sustained enthusiasm than might result from traditional text-based introductory programming curricula. This is more difficult in the arts, where the underlying tools are not only cognitive, but sensorimotor as well. We hypothesize that machine learning algorithms can open early pedagogical pathways to high-level skills in a variety of artistic disciplines.

In this paper, we will explore two case studies from our own work in which we have employed this approach to assist in music pedagogy. After discussing our experiences, we briefly explore possible extensions to other artistic disciplines.

Goals for the Workshop

Our primary goals in participating in this workshop are (1) to seek opportunities for machine learning tools in artistic education, (2) to seek out non-artistic domains that may benefit from this approach, and (3) to discuss, with child-computer interaction experts, more rigorous approaches to validating our hypotheses and building on our initial case studies.

Case Study 1: Songsmith

Songsmith (Figure 1) is a computational creativity tool that lets novice musicians create music just by singing a melody. A machine learning system analyzes the user's voice to choose appropriate chords, then renders those chords as a music arrangement. The user can use intuitive GUI controls to adjust style and chord progressions, without understanding the details of the underlying algorithms and without possessing any knowledge of music theory. The primary goal of the software is to give the novice user a taste of music creation, at the level a songwriter might think of music creation, without the underlying instrumental skills or music theory understanding. Songsmith's core technology is described by Simon et al [4].

Songsmith as an educational tool

Though Songsmith was not originally designed as an educational tool, preliminary feedback after its release suggested that Songsmith could assist teachers in encouraging students to be creative: many music teachers know that sometimes just helping kids "find their spark" is the hardest part of stimulating musical creativity. Furthermore, teachers inquired about using Songsmith to teach musical concepts that are sometimes difficult, particularly how chords are used in pop music and how melodies and chords fit together. Even outside of music classes, Songsmith showed promise for encouraging creative approaches to learning. Teachers sent examples of students writing songs about science concepts, and parents described children using Songsmith to compose musical mnemonics for multiplication tables. In all of these scenarios, Songsmith essentially replaces low-level skills with algorithms, allowing students to interact with music at a level that novices find compelling.

Consequently, we decided to further explore the educational opportunities for this tool by releasing Songsmith into several educational environments; we will discuss some of these in the next section.

Feedback from teachers

In the two years since its release, Songsmith has been deployed in several classroom environments, including public schools (including a deployment across a large school system in Australia), specialized music programs (such as the Seattle Symphony's "Soundbridge" program), and some classrooms not devoted specifically to music (for example science classes or English-as-a-second-language classes). Early interactions between teachers and students suggest that in fact this approach does provide the engagement we hoped it would. A thorough validation of this approach is beyond the scope of this workshop paper; in this section, we will present quotes from instructors in several classroom scenarios that indicate further investigation is warranted.

In the context of an out-of-school musical enrichment program in which children were engaged for only a brief period, instructors hoped to use Songsmith to stimulate interest in music education that would persist after the program. Preliminary feedback indicated that the teachers were more than happy with the results: "It always elicits squeals of delight when the song is played back, and kids get to listen to their very own song." […] "Eventually, we hope to also use it as a composition tool for older kids with serious musical aspirations. (I'm still amazed that you can sing any song, in any key, and Songsmith will give you the complete chord progression!)"

Another teacher contacted us regarding Songsmith's use in teaching English to non-native speakers, reporting satisfaction for this application as well. Here the goal was not to stimulate musical creativity per se, but to improve the overall engagement with the

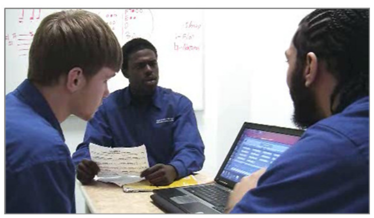

Figure 2. Songsmith in use in a high school music classroom. Students were assigned the task of creating an original song in small groups; Songsmith allowed those students with limited instrumental experience to participate.

material. “I teach English, and chants and songs are a wonderful way of teaching the language. I am good at making up tunes in my head, but I cannot play an instrument. […] I have been using Songsmith to great effect in my classes, and the children love singing along to the songs I have created.”

Songsmith was also requested by several teachers in a large school district in Australia, and was subsequently incorporated in a district-wide software deployment program. Preliminary feedback from non-music teachers was positive from this program as well, also for the purpose of generating classroom enthusiasm for other subject material: “It is a fantastic program and I have begun using it with my Year 2 class. It is a fantastic tool for presenting work in a new way. Children are able to write songs that reflect what they have learnt or to teach others about their learning. It truly takes my teaching to another level.”

Finally, Songsmith was deployed in a high school music classroom in an urban U.S. area, with the intent of scaffolding songwriting and music creation pedagogy for students of various musical experience levels (Figure 2). Positive results were reported here as well: “One of the great things about using Songsmith is it caters to multiple students’ interests. On the most basic form, the students are able to sing the songs and hear what they’re singing would sound like as a song. For the students that are a little bit more music-savvy, they understand a little bit more the demonstration of the chords and how the progressions work together and they’re able to take it to another level on their instruments.”

Collectively, we believe this feedback supports our hypothesis that algorithmic support for musical creativity—particularly the partial replacement of lower-level skills with machine learning tools—offers significant pedagogical value. We look forward to discussing these experiences in more detail at the workshop.

Case Study 2: PLOrk

The Princeton Laptop Orchestra (PLOrk; Figure 3) was created in 2005 as an undergraduate teaching initiative and performance ensemble [5]. In concerts, groups of five to thirty PLOrk students play new compositions using laptop-based instruments, controlling the computers’ sounds in real-time using input devices ranging from the mouse and keyboard to accelerometers, webcams, and custom sensor devices. In the classroom, students from a wide range of academic majors learn about computer programming, music composition, and interactive systems-building through creating their own computer music compositions and laptop-based instruments.

Pedagogical innovation has been a core motivation of PLOrk since its inception, and in 2008 the ensemble was one of seventeen winners of the John D. and Catherine T. MacArthur Foundation’s Digital Media and Learning Competition. Much of our work under the MacArthur grant has focused on exploring new approaches to creating laptop-based instruments, and incorporating those approaches into the PLOrk classroom curriculum. Laptop-based instruments can make it easier for novice musicians to engage in expressive music performance, since—unlike many acoustic instruments—they can be designed to be easily playable without requiring years of practice. Many

Figure 3. Two PLOrk students performing in a concert. In the laptop instrument used in this piece, performers control the synthesized sound by tilting and hitting the laptop. The instrument uses the Wekinator to detect hits to different locations of the laptop, playing a different sound for each location.

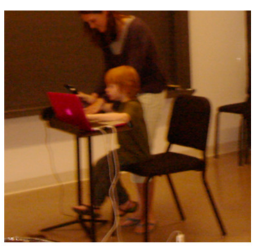

Figure 4. A child plays an instrument created by a PLOrk student using the Wekinator. In this instrument, joystick position controls which chord is played.

PLOrk students have little or no formal musical training, and laptop-based instruments enable us to train these students in performance practice and improvisation. Unfortunately, the development of truly playable and expressive laptop-based instruments presents its own hurdles, particularly the need to write software that encodes appropriate relationships or "mappings" between performers' actions (e.g., as sensed by gestural controllers) and the control parameters of sound- or music-generating algorithms. A major research project at Princeton has therefore been the development of machine learning software that allows an instrument-building user to focus on crafting the desired relationship between performer gesture and computer sound, without programming and without attending to the low-level details of the controllers or sound synthesis algorithm. In other words, machine learning is used to facilitate the high-level design of musical systems.

Our software, called the Wekinator [1], allows users to interactively design a gesturally-controlled instrument by iteratively providing examples of performer gestures paired with the computer sound that should result from that gesture. For example, a user can create a webcam-controlled drum machine by demonstrating a few examples of one gesture in front of the camera matched with one synthesized rhythm, then a few other examples pairing a different gesture with a different rhythm. A machine learning algorithm can then infer the relationship between gesture and sound from the examples, and the user can easily test whether the learned model produces the desired sounds for new gestures in front of the camera. If the model does not behave as desired, the user can often improve it by providing additional examples.

To date, twenty-two PLOrk students have used the Wekinator to build their own instruments for course projects (e.g., Figure 4). The use of the software has greatly accelerated the process of building a working instrument, which can now take minutes instead of hours or weeks. Students can now spend less time debugging code and more time experimenting with many different instrument prototypes, allowing them to learn more about the musical consequences of different designs. Also, many students have enjoyed using the Wekinator to discover new sounds and instrument designs that they hadn't imagined themselves: when using continuous learning algorithms (here, neural networks), students can rely on instruments to "interpolate" between and beyond the sounds present in the training examples, sometimes in surprising and musically useful ways. Professional composers who have used the Wekinator have similarly valued how it facilitates rapid prototyping and exploration and allows serendipitous discovery of new gesture-sound relationships [1].

Because of these experiences, we have long been interested in applying the Wekinator to allow children to build their own laptop-based musical instruments. Given a set of controllers (e.g., Nintendo Wii controllers, Microsoft Kinects, joysticks, webcams) and a set of pre-fabricated musical software components (e.g., for playing rhythms and melodies), young students could use the same interactions of demonstrating gestures and sounds in order to create and evolve their own instruments. When we provided grade school children with hands-on demos of PLOrk instruments at a HASTAC[1] event in Chicago in 2009,

[1] Humanities, Arts, Science, and Technology Advanced Collaboratory: http://www.hastac.org/

they were enthralled to discover that computers could be used to perform music. We anticipate that the opportunity to actually design these instruments themselves would offer a compelling way for students to learn about sound and music composition, to express themselves through music without prior musical instruction or practice, and to creatively design unique interactive computer systems without programming expertise.

Moving Into Other Domains

Our discussion so far has focused on musical creativity only because that is our area of expertise; we are excited to see the general approach of using machine learning for pedagogical support applied to other artistic domains as well, part of our motivation for participating in this workshop. For example, Hart et al. [2] discuss algorithmic support for visual creativity, and Howe et al. [3] discuss similar techniques that might be applied to literary creativity. We therefore look forward to working with other instructors, technologists, and artists to further validate the hypothesized benefits of algorithmic support for artistic creativity, and to working with domain experts in other artistic disciplines to develop techniques suitable for those applications.

References

[1] Fiebrink, R. 2011. Real-time Human Interaction with Supervised Learning Algorithms for Music Composition and Performance. PhD thesis, Princeton University.

[2] Hart, J. 2009. Assistive Technology for the Aesthetically Impaired. *Proceedings of the CHI 2009 workshop on Computational Creativity Support*.

[3] Howe, D. 2009. RiTa: Creativity Support for Computational Literature. *Proceedings of the CHI 2009 workshop on Computational Creativity Support*.

[4] Simon I., Morris D., and Basu S. 2008. MySong: Automatic Accompaniment Generation for Vocal Melodies. *Proceedings of ACM CHI 2008*.

[5] Trueman, D., Cook, P. R., Smallwood, S., and Wang, G. 2006. PLOrk: The Princeton Laptop Orchestra, Year 1. *Proceedings of the International Computer Music Conference.*