Project Idea
Music Composition Software

Proposal by Essential Solutions

(Harper / Pota / Sheelin / J. Thomson / R. Thompson / Wellard)

Glossary of Terms

DAW - Digital Audio Workstation

MIDI - Musical Instrument Digital Interface. MIDI can be thought of as the 'code' which conducts the performance of a digital musical instrument. MIDI carries event messages; data that specify the instructions to play music. This includes a note's pitch, notation, velocity, duration

Tempo - The "speed" of a piece of music. Tempo is measured in beats per minute (BPM)

Key - This refers to the root note around which a musical composition is based

Chord - Two or more notes played concurrently

Melody - A sequence of notes in succession

Overview

This project aims to create a software tool for songwriting and music composition. It will be implemented as a 'plugin', to be used inside any DAW (Digital Audio Workstation). The user will set parameters such as song tempo, key, desired emotion, level of complexity,. The software will generate unique musical phrases in accordance with these choices, both exactly to the specifications of the input, and including improvisation in the direction indicated by the user. These ideas can be altered and auditioned easily in real time, then exported as MIDI data - ready to use immediately in any active music project. The range of options available to the user will be labeled with familiar keywords such as "'emotion", "temperature", "complexity", rather than ugly and technical musical jargon. In my opinion there is a huge scope for merging the vast accumulated knowledge in songwriting with the cutting edge digital technologies that have catalysed many other industries like it.

There are many possible reasons this artform hasn't been accelerated by computer power. Let's consider a couple. Perhaps, songwriting as a craft is so intertwined with the deep and unique personal experiences of the songwriter, that introducing computers to the process might harm its integrity. Or, that somehow, it's "cheating" the true process. As someone who has been writing songs for most of my life, I couldn't disagree more. Creativity is about experimenting, pushing limits, taking risks, connecting dots which otherwise wouldn't be connected.

This is also part of a larger concern which arises often - that the best human creative achievements risk being outdone by an emotionless machine. Admittedly it's a scary thought, but it doesn't need to be. There are several factors which contribute to the value of a piece of music, and whether or not a computer generated options for some of the basic chord sequences is almost irrelevant in the scheme of things.

Another possible reason this kind of tool hasn't been developed before, is that musicians as a group generally aren't technically inclined, and wouldn't take well to the idea of incorporating an algorithm into their craft. While I do partially agree with this, the reality is - the generation of people who will be implementing this tool are way more tech savvy than their predecessors. The generalisation of musicians as artsy types, who prefer to feel rather than think, is now obsolete. Many people in the music business today have found music via exploration which started with computers, as opposed to before when musicians were forced to learn how to use a computer just to keep up with industry standards.

Motivation

In 2020, songwriting generated over $470 million AUD in royalties for Australian writers alone. Countries like the USA and UK are proportionally more lucrative in accordance with their economic and population size. While the content delivery side of the entertainment business has more or less adapted to the digital age with streaming & subscription services, the process of songwriting is still largely stuck in the pre-internet era.

As someone who's been writing music on computers for too long, I'm acutely aware of the amount of human labour which is needlessly repeated on tasks which could be automated. One goal of this project will be to automate these tasks for efficiency of workflow. The other (and more interesting) goal relates to the 'Randomize' function, and its ability to intelligently suggest musical options which could extend the creative ability of the writer to a level which may not have been possible by simply relying on the circuitry of the human brain.

I intend to use my experience as a songwriter, in conjunction the collective experiences and skills of all of our group members, to create this tool to help others increase their efficiency, productivity, and possibly even creativity.

Description

The primary function of this plugin is to enable users to custom build complex and emotive musical compositions without getting bogged down with the technical aspects that are normally associated with it. Users will input their choices into the plugin, which will display and audition the suggested ideas in real time. These musical ideas can subsequently be exported from the plugin and used to conduct a performance from any digital synthesiser. I would like to highlight three of the key features -

- 1. *Jargon-free communication of musical ideas*. This plugin will represent commonly used musical "building blocks" as more familiar phrases such as "sad", "happy", "uplifting", "suspenseful", as opposed to the confusing musical notation in which these are normally written. In doing so, users with limited knowledge of music theory can simply cycle through options generated by the plugin, make decisions based on listening and feeling rather than thinking. The intensity of the performance can be increased or decreased by tweaking a knob with labels indicating "cool" on one end and "warm" on the other. This helps to address one of the main concerns of creatives who are apprehensive about incorporating an algorithm into their personal creative process - that the art will lose its feel and become clinical and lifeless. By removing the clunky and scary looking musical notation we increase the chance of tapping into the emotional part of creatives and thereby elevate their creative output rather than hinder it.

- 2. *Randomise function.* Users will be able to use this feature as an intuitive extension of their natural songwriting ability. With a slight nudge in the right direction, this feature can enable the user to audition a large number of usable musical idea options in a small amount of time, much faster than would be possible if manually cycling through options one by one while noodling on the keyboard, as is usually the way this is done. With the help of Machine Learning, we can narrow the range of suggested ideas, and ideally even cater it to the user's personal style in ways not immediately obvious to them if they were just using a trial and error method. This is possibly the most exciting area of exploration in this project.

- 3. *Humanize.* One of the biggest drawbacks of digital music composition is that the music can often sound a little too "perfect" and "robotic", as opposed to the more human and natural feeling present in analog recordings of the past. The way certain circuits were configured in hardware gear often produced glitches and artefacts that were musically pleasing and gave the music an organic aesthetic. In a similar way, when a musician performs a piece, the notes in one chord don't all start at the same exact time and run for

the exact same duration. There are slight variations, both intentional and unintentional, which "colour" the performance and give it a unique sonic fingerprint. This plugin will enable users to control how "perfectly" the musical information is performed, and if required, vary the note information around an algorithm designed to imitate the natural articulations of a human performance. As discussed in the 'Randomize' section above, we can expand this function further by incorporating machine learning data to generate new possibilities for options which may not have been stumbled on before.

Tools + Technologies

The bulk of this project can be completely designed, coded and tested on a personal computer. Many parts of this plugin including the main composition algorithm can be developed without a GUI. This will make it easier to focus on the most important part of the project. As a Mac user, I could do this project in X Code 4. My research has showed that C++ is the most popular language for coding audio plugins. Many developers use JUCE. Since this plugin will only be outputting MIDI data, no additional codecs or system files will need to be installed on users' computers to run this software.

Cyber Security

With data security and integrity at the most risk it has ever been due to the nature of technology and growing ability to take and compromise data at will, wanting the programs you make on Digital Audio Workstation (DAW) to be safe and give you less to worry about is not an unreasonable request. Fortunately, data protection can be assured by implementing an encryption software within the DAW. A model similar to Folder Lock software would be ideal, with the below attributes being the most suitable for a DAW. Folder Lock allows the user to create:

- Locked/Hidden folders
  - Enables user to Lock and Hide files and folders within seconds.
  - Enables Password Protection and restricts unwanted eyes from viewing files, folders and drives.
  - Once a folder has ben locked, it will be hidden from its previous location and can only be accessed through the DAW interface.
- Encryption of files
  - Creates virtual Lockers. Files are transported to a digital 'Locker'.
  - You can create 'Locker(s)' which are protected by AES 256-bit encryption.
  - Simply copying to a to a Locker automatically encrypts them.
- Cloud Backup
  - Folder Lock infrastructure offers a 2-way encryption and backup method, letting you backup your encrypted 'Lockers' online.
  - Allows a password protected backup of your Lockers so that the information is not only secured but also protected from loss or damage.
  - If your hardware gets stolen or data gets erased, simply recover your data back from an online account.

Machine Learning

Machine learning (ML) nowadays has a broad range of applications in many aspects such as image recognition, natural language processing, data classification/generalisation etc. In addition, acoustic modelling is another emerging exemplary use of ML.

Tension chord is the one of characteristics of musical melody lies in the composition to set a music tone or make a music harmoniously enchanting. The presence of tensioned chord and relax chord and its movement resembled chord progression give a music a unique profile. Navarro-Cáceres et al (2020) led research to develop a predictive model using ML, in which progressive chords can be computer-generated according to the setting change of tonal tension profile by a user. The tension profile serves as both sample data and test data. Through feeding a computer a myriad of sample data of different chord combinations, the computer will statistically study the underlying patterns and variations and then it can populate some possible progressions. This tool could not only facilitate novice writers to compose complex sophisticated notes but also help professional composer to fine-tune their work in quicker and more diverse way.

Source - https://www.mdpi.com/2076-3417/10/17/6039/htm
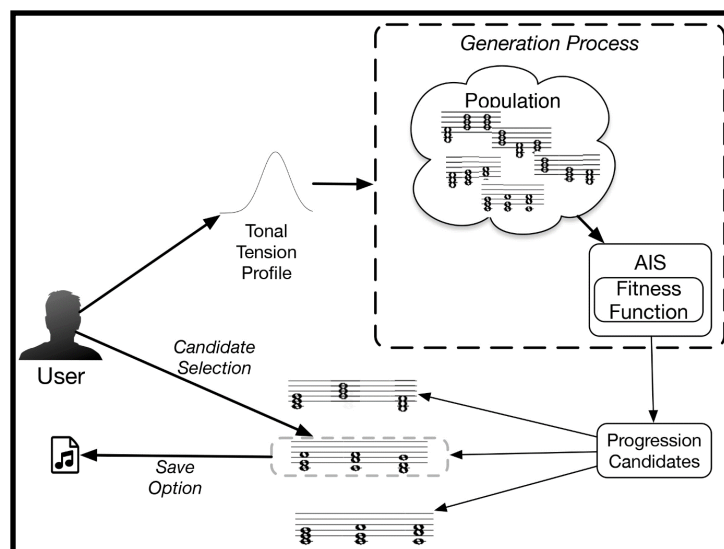
Skills

In order to determine what kinds of skills are needed for this project, we must first look at what we aim to achieve, and from there we can work backwards to gain an insight into the actual steps required.

Successful execution of this project involves two main parts -

1. *Design a bank of music theory knowledge which can be drawn on by our program to make decisions.* This is more than just a case of storing pre-made chords and melodies which the user can shuffle through. That would be the old way of doing it. In our case, it would require actually teaching the software the "rules" of what is acceptable and what isn't when it comes to combining notes to create a chord, combining chords to create a sequence, and combining chord sequences to make a song.
2. *Create an algorithm to generate music ideas for the user, based on their inputs, which can be edited in real time..* This will involve some mathematical knowledge combined with music theory knowledge. This algorithm is what will be the most unique part of our project compared to other programs in the same lane, because much of the music knowledge we will be drawing on is commonly known stuff which anybody can use.

Mathematically speaking, music composition is a lot more formulaic than most people realise, given the profound emotional impact of the right notes at the right time. For any given key / scale combination in Western music, there are a set number of chord sequences which "work"'. These sequences are defined by the relative positions of successive and overlapping notes across the subjective musical scale, not their objective position as sounds on the frequency spectrum. Also, the same chord progressions that work in one scale can be transposed to any other scale, simply shifted up or down. In other words, pretty much all the pop music you've ever heard is made up of the same few musical building blocks, which can only be played and combined in relatively limited and predictable ways. Our goal is to create an algorithm which can intuitively follow directions from the user to generate usable idea options, hence allowing the rapid execution of multi-step processes which would otherwise require a lot more brain power.

Outcome

This plugin will enable users with limited musical knowledge to write compelling compositions in a matter of seconds and seamlessly incorporate them into any music project. It will achieve this by drawing on a bank of pre-programmed music theory data, as well as simple algorithms designed to use the power of sorting and randomising functions to quickly generate musical phrases. The available options are given "layman" labels such as emotion, colour, temperature rather than technical music jargon. The MIDI files which will be generated by this plugin can be used to trigger sounds from and essentially "play" any synthesiser, both software and hardware alike, making it highly versatile and usable.

References

2020, *APRA AMCOS 2019 - 2020 Year In Review*, Sydney, Australia
https://assets.apraamcos.com.au/images/PDFs/About/APRA_AMCOS_Year_in_Review_2020.pdf

Navarro-Cáceres, M., Merchán Sánchez-Jara, J., Reis Quietinho Leithardt, V. and García-Ovejero, R., 2020. '*Assistive Model to Generate Chord Progressions Using Genetic Programming with Artificial Immune Properties*' viewed 12/10/2021
https://www.mdpi.com/2076-3417/10/17/6039/htm

Ackerman M., Loker D. (2017) *Algorithmic Songwriting with ALYSIA*. In: Correia J., Ciesielski V., Liapis A. (eds) Computational Intelligence in Music, Sound, Art and Design. EvoMUSART 2017. Lecture Notes in Computer Science, vol 10198. Springer, Cham. https://doi.org/10.1007/978-3-319-55750-2_1

Agarwala, N., Inoue, Y., & Sly, A. (2017). *Music composition using recurrent neural networks.* [online]. Web.stanford.edu. Retrieved from https://web.stanford.edu/class/cs224n/reports/2762076.pdf.