Lab 1 - CrossFade Product Description

Arnaud Blay, Dylan Hopper, Hyacinth Abad,
Joseph Wassell, Mohamed Abdullahi, Selai Naim, Virginia Vano Ruano

Old Dominion University

CS411W

CrossFade

11/17/2023

Version 2

Table of Contents

Figures and Tables

1. Introduction

Starting in a new field can be challenging. This is the case in the music industry, where there is a need for technical equipment and knowledge to share your music. There is a need for a keyboard or a soundboard to create a MIDI file, which is one of the standard formats for sharing music over the Internet, and these devices can be very expensive. Starting musicians also face the problem of not knowing the standard music notation represented in sheet music. That makes the requirements for learning music overwhelming, which often causes people to abandon the field early. The barrier of entry is very high for starting musicians. There is a need for a program which lowers the barrier of entry, allowing for the transcription of one's music even without having an advanced understanding of music theory, and which also removes the need of specific equipment to produce music in written format. CrossFade is the solution that will bring those features.

2. Product Description

CrossFade is a program designed to lower the barrier of entry for starting musicians. It both removes the requirement for expensive technical equipment and allows musicians to write down their music even with little or no experience in music notation. This will allow starting musicians to share their music while still learning.

2.1. Key Product Features and Capabilities

When using CrossFade, music can be input to the program via a microphone or a recording. The program will then automatically transcribe it to the desired music format: a MIDI file, an MusicXML file, or sheet music. Then, the program will detect usual errors made by automatic transcription algorithms, and correct them, and it will allow you to make further edits in case there is something that needs to be changed. The program will also accept MIDI and MusicXML input since they can be edited through the program if there are errors in

existing files. CrossFade will also recognize different instruments, giving each one its own section in the music files, and making corrections depending on notation conventions typical to each instrument.

2.2. Major Components (Hardware/Software)

CrossFade will require the use of a computer, in which the program is run, and a microphone and instrument if the audio transcription function is going to be used. Regarding to its software, it will be written in Python, but it will use C++ to run the heavier music algorithms. CrossFade will use *Yafee* and *Superpowered* as the libraries used for audio parsing and interpretation of the input, and it will use *IRMAS* and *Philharmonia* samples afor its database for instrument recognition. It will provide support for MusicXML files through the libraries *libmusicxml*, *pymusicxml*, and *music21*, and it will use artificial intelligence, like *TensorFlow* and *mlpack* to learn from its mistakes and provide a more accurate error correction.

This will require the development of several algorithms. There will be five algorithms that will be used:

- Audio file and live audio transcription: This will start by creating a new MIDI and MusicXML file, ask the user to select the musical clef and key signature, and then listen to the recording note by note writing them down in the appropriate format. Once there are no more notes, it will adjust the notation.
- MIDI file transcription: This will start by creating the project and asking for the musical clef and key signature. It will also ask if the user wants all the channels in the MIDI file in each own track, or all of them together. Then, it will convert the file to MusicXML, allowing the user to correct the errors.

- Automatic transcription correction: This will use a deep learning model, analyzing similarities. When it detects an error, it will fix it. Afterwards, it will ask for feedback, which it will use to improve its ability.
- Manual transcription correction: This will allow the user to look at the whole transcription, selecting the section that they want to edit and making the changes they consider adequate. Once the user is satisfied with the correction, output the corrected transcription.
- Musical instrument recognition: This algorithm will also use deep learning to analyze the pitches conforming the audio input. It will detect the pitches which typically correspond to individual notes in each individual time frame, and then streams them into a single pitch trajectory, differentiating them from one another.

On Figure 1, the input used is placed on the left side, with the microphone representing the life audio. On the bottom we see the user interacting with the program on the PC. The program interacts with the database for the music transcription, which is depicted on the top of the diagram. The program output is located on the right side, which can be either a MIDI file or sheet music.

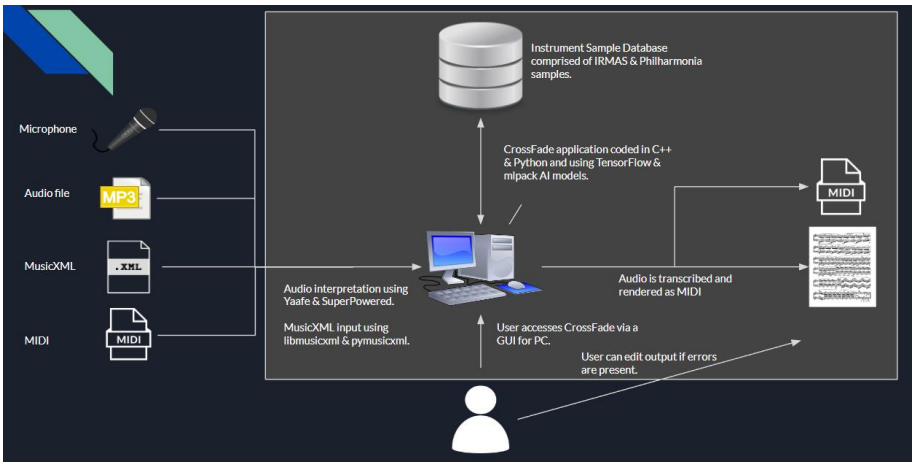


Figure 1 - Major functional component diagram

3. Identification of Case Study

This product is being created for starting musicians, since the main purpose of the program is lowering the barrier of entry to the music industry. It can also be used by music creators with more experience, because of the edition feature that it provides.

4. Product Prototype Description

CrossFade has two main features: transcription and error correction. For the prototype, there will be a focus in the error correction feature, as that one is less common. Table 1 shows the transcription error features that the prototype will have in comparison to the complete real product. Table 2 shows the transcription features in the same manner, and Table 3 shows the note recognition features.

Transcription Error Correction		
Feature	Real World Product	Prototype
Transcription Error Correction	✓	✓
Highlight Possible Errors	✓	✓
Take User Feedback	✓	✓
Offer Possible Solutions	✓	✓
Compare Single Note from Original to Written	✓	✓
Compare a Segment from Original to Written	✓	✓

Table 1 - Feature comparison between full product and prototype (Transcription Error Correction)

Transcription		
Feature	Real World Product	Prototype
Live Audio Transcription	✓	
MIDI Input/File Transcription	✓	
MIDI Transcription Correction	✓	✓
Multitrack Transcription	✓	
MusicXML Compatibility	✓	✓
MIDI File Export	✓	✓

Table 2 - Feature comparison between full product and prototype (Transcription)

Note Recognition		
Feature	Real World Product	Prototype
Monophonic Note Recognition	✓	
Polyphonic Note Recognition	✓	
Instrument Recognition	✓	
Instrument Distinction	✓	

Table 3 - Feature comparison between full product and prototype (Note Recognition)

4.1. Prototype Architecture (Hardware/Software)

The prototype will accept MIDI and MusicXML files. It will then check the import for known errors and will change them and highlight them so that the user knows what has been changed. It will also offer other solutions for the error, and it will allow the user to play anything that the user has selected, or the whole piece if nothing was selected, to check if they like the changes that were made.

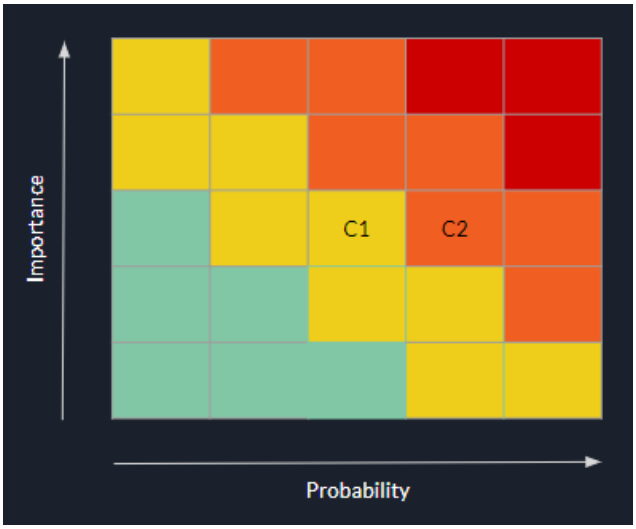
4.2. Prototype Features and Capabilities

The prototype will present the ability to correct MIDI files, it will be compatible with MusicXML files, and allow the files to be exported. It will correct the errors detected in the input, highlighting the changes made and offering other solutions. It will allow to play the written music and compare it with the original audio if it was provided, whether a single note is selected or several. If nothing is selected, it will play the whole piece. It will also ask for feedback to improve its algorithm.

4.3. Prototype Development Challenges

Table 4 shows the customer risks, representing how probable and how important they are for the application.

The Customer Risks are:



- C-1 Difficulty using the application.
- C-2 Unfamiliarity with the terms.

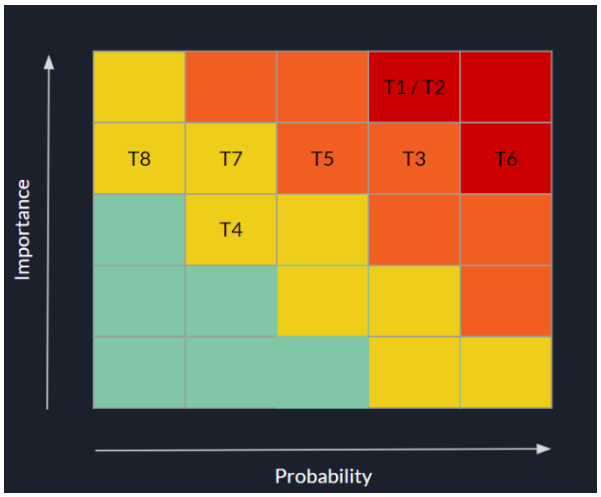
Table 4 - Customer Risks

It is possible that the users do not know how to use the application, or they are unfamiliar with the notation. The program will include a tutorial to show how to use it and will provide definitions for the different notations. That way, the program also serves its purpose of helping starting musicians to learn the music notation.

Table 5 shows the technical risks, representing how probable and how important they are for the application.

The Technical Risks are:

Table 5 - Technical Risks



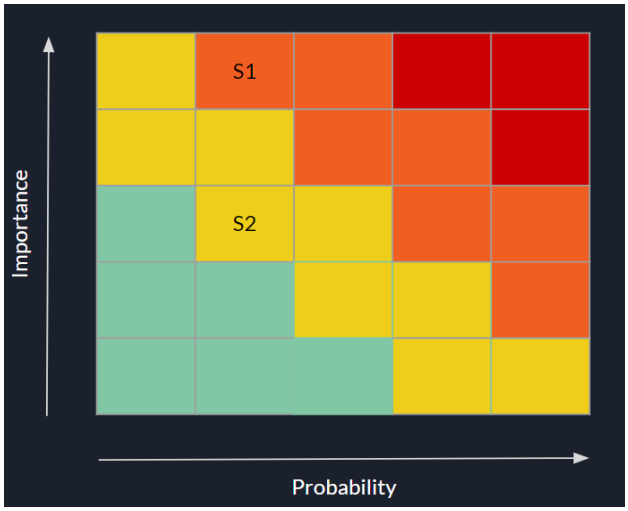
- T-1 Multi-pitch detection feature difficulty.
- T-2 Note Tracking Feature Difficulty.
- T-3 Overestimating system performance.
- T-4 Limited system performance due to musical diversity.
- T-5 Library Availability.
- T-6 User Inputted Recognizability.
- T-7 Applying incorrect correction rules when correcting music notation.
- T-8 Bias in the correction process.

The program consists of two parts: music transcription and error correction. The prototype will focus on the error correction, which brings its own problems, like correcting incorrectly or favoring specific styles. This would be corrected by allowing the user to edit the transcription and freeing the model of bias. Looking at the features for transcription, one can notice that they are complex algorithms that might not work properly, which implies that allowing the user to edit complicated sections of music will allow the program to provide a full, accurate transcription.

Table 6 shows the security risks, representing how probable and how important they are for the application.

The Security Risks are:

Table 6 - Security Risks

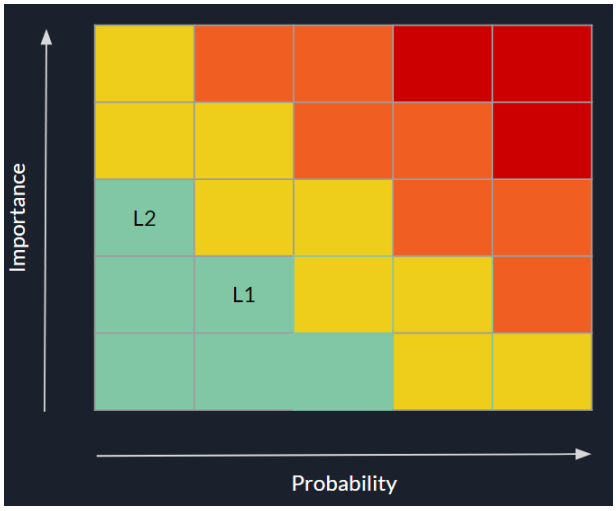


- S-1 Intellectual property.
- S-2 Application vulnerability.

The prototype will face security issues involving application vulnerability and intellectual property violations, which the team will avoid by using best practices and securing the data so that it cannot be accessed by unauthorized entities.

Table 7 shows the legal risks, representing how probable and how important they are for the application.

The Legal Risks are:



- L-1 Unlawful product use.
- L-2 Copyright Law.

Table 7 - Legal Risks

Users might try to distribute CrossFade's music transcriptions as their own or use copyrighted material. The team will include a water mark for CrossFade's products, and it will use non copyrighted material as training data.

5. Glossary

Convolutional Neural Network (CNN) - Deep learning algorithm which can differentiate one image from another by assigning weights and biases to different aspects of the images. It is used in audio to differentiate different frequencies in a visual format.

Deep Learning - Subfield of machine learning which uses neural networks to solve complex problems. Learning comes directly from the data, instead of being hand-engineered by humans.

Keyboard - An electronic piano used to produce sound and MIDI information.

Monophony - A phrase of music in which only a single voice is played at a time.

Musical Instrument Digital Interface (MIDI) - A communications protocol used to connect physical and virtual music devices and instruments. MIDI files store note information which can be used to trigger instruments and devices.

MusicXML - A markup language format used to interchange and distribute digital sheet music.

Polyphony - A phrase of music in which more than a single voice is played at a time.

6. References

Automatic Music Transcription and Ethnomusicology: A user study. (n.d.). Retrieved from https://diva-portal.org/smash/get/diva2:1474663/FULLTEXT01.pdf

H. Takeda, T. Nishimoto and S. Sagayama, "Rhythm and Tempo Analysis Toward Automatic Music Transcription," 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, Honolulu, HI, USA, 2007, pp. IV-1317-IV-1320, doi: 10.1109/ICASSP.2007.367320.

Huang, Z., Jia, X., & Guo, Y. (2019, June 29). State-of-the-art model for music object recognition with Deep Learning. MDPI. Retrieved February 8, 2023, from https://www.mdpi.com/2076-3417/9/13/2645

Jovanovic, J. (2015, February 2). How does Shazam work? music recognition algorithms, fingerprinting, and processing: Toptal®. Toptal Engineering Blog. Retrieved February 8, 2023, from https://www.toptal.com/algorithms/shazam-it-music-processing-fingerprinting-and-recognition

Scarlatos, L. L. (n.d.). Continuous media. Audio. Retrieved March 1, 2023, from https://www3.cs.stonybrook.edu/~lori/classes/GUI/sound.htm

Solanki, A., & Pandey, S. (2019, January 30). Music instrument recognition using deep convolutional neural networks - International Journal of Information Technology. SpringerLink. Retrieved February 8, 2023, from https://link.springer.com/article/10.1007/s41870-019-00285-y

Zhang, X. (2022, March 11). Aided recognition and training of music features based on the internet of things and Artificial Intelligence. Computational intelligence and neuroscience. Retrieved February 8, 2023, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8933112/