

Singing Data Labeling Tool Milestone 4 Progress Evaluation Report

Group members:

Nandith Narayan nnarayan2018@my.fit.edu
Avinash Persaud apersaud2018@my.fit.edu
Carlos Cepeda ccepeda2018@my.fit.edu

Advisor:

Dr. William Shoaff wds@fit.edu

Client:

Caleb Matthew Long, Appalachian State University

Task	Completion %	Nandith Narayan	Avinash Persaud	Carlos Cepeda	To Do
Add Timeline to UI	100%	30%	70%		
Quality of life UI changes	100%	70%	30%		
Refactoring	100%	10%	90%		

Task Discussion:**Task 1:**

To add the timeline widget to the GUI, we had to make a couple of changes to various parts of the system. Firstly, we had to create a generic widget class to represent a “track”. When labeling, labels (features like phonemes, syllables, etc...) can be layered on top of each other. To allow for this, we use tracks. In a single track, all labels must be non-overlapping. However, all tracks are overlaid on top of each other when producing the output. This allows for multiple labels to be overlaid at the same time. We made every track handle mouse inputs, thus allowing us to sync cursors across the wave draw element and the spectrogram element. After this we made changes to our intermediate data structure to include a representation of labels that supports multiple tracks/channels.

Task 2:

To improve user experience we made a number of changes to the GUI. First, we added a “cursor” to the wave display elements of the GUI. This cursor is a vertical line that can be moved with the mouse. It is used to precisely select a start time for creating a new label. It can be used

to select and move labels as well. This cursor syncs across the various UI elements, which allows the user to easily view the location of features within the audio. For example, this cursor can be used to manually find out when a word is spoken. Another such feature we added was being able to scroll through all the present tracks in the event that the number of tracks doesn't fit in the window.

Task 3:

To improve scalability, we refactored the GUI side of the code to use the widget class provided by our graphics library, Qt. All GUI elements now are instances of classes that inherit from Qt provided classes. This allows us to use a lot of the functionality provided by Qt such as resizable windows, user input, and scroll bars. Our project now follows the MVC(Model View Controller) pattern. This led to a significant speed increase when drawing the waveform. Previously, we redrew the waveform every time an update was made. Now, we create Qt-line objects once, and simply update their position when an update occurs.

Contribution Discussion:

Milestone 5 task matrix:

Task	Nandith Narayan	Avinash Persaud	Carlos Cepeda
Generate output file	40%	40%	20%
Integrate phoneme detection and classification networks into the application	50%	50%	
Create note view UI element	30%	50%	20%

Milestone 5 Task Discussion:

Task 1:

To generate the output file, we must first interweave all the overlapping tracks. Then, we'll have to go through every label, format it, and write the data to a file. We've taken the time to understand the HTS singing label format, so we should be able to overcome this technical challenge during milestone 5.

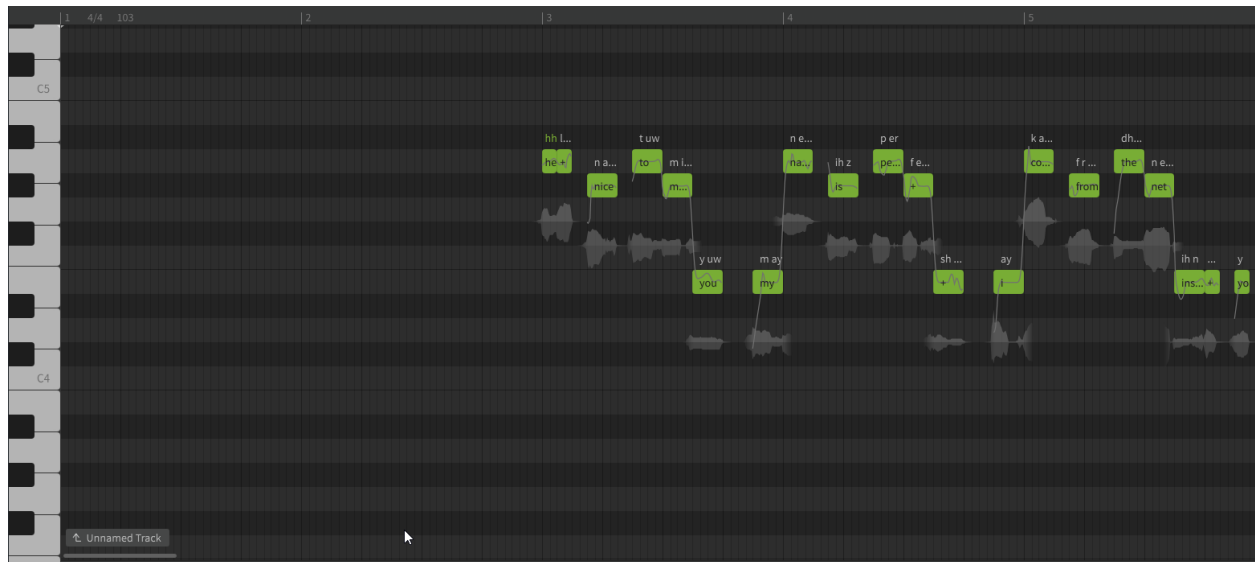
Task 2:

Since we used the same machine learning libraries for both networks, integrating it into our application only requires a couple of libraries namely, TensorFlow, and Keras. One of the

challenges we face with integrating these networks is that these networks take a lot of computational power to run. We have thought of 2 potential solutions. The first is to run the models on a different thread. This way, the user can continue to use the application while the networks run. However, this can cause issues if the user modifies data that the networks are using. The other solution was to have a loading bar that prevents the user from doing anything until the networks finish running. Further evaluation is required.

Task 3:

The note view will display all the notes present in the song. In addition to the notes, it will display the section of the waveform that makes up each note. The below image depicts a similar feature present in a vocal synthesis program called “SynthesizerV”.



Screenshot of piano roll feature present in SynthesizerV

Client Feedback

Discussed with the client over discord occasionally.

Date of meeting with Faculty Advisor: **2/14/22**

Faculty Advisor feedback on milestone tasks:

- Evaluation by Faculty Advisor
- Faculty Advisor: detach and return this page to Dr. Chan (HC 214) or email the scores to pkc@cs.fit.edu
- Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)

Carlos Cepeda	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Avinash Persaud	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Nandith Narayan	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10

■ Faculty Advisor Signature: _____ Date: _____