Requirements Document

Singing Labeling Data Tool

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

Advisor:
Dr. William Shoaff

Table of Contents:

1. Introduction

1.1 Purpose

The purpose of the Singing Data Labeling Tool is to provide an easy and friendly interface for preparing singing data for machine learning. The goal of this document is to explain the features of this project in detail.

1.2 Scope

The scope of this project is to assist the user with labelling singing data for vocal synthesis. This tool will automate parts of the labelling process. It will be able to output to multiple output formats. It will contain shortcuts for commonly used features. It will display key features of the data in a graphical format.

1.3 Definitions

1.3.1 Tagging - The process of applying labels to regions of audio that defines a certain property (phoneme, syllable number, note pitch, etc.)

1.3.2 Mono Label - A file that contains phoneme identity, start times, and end times.

1.3.3 Phoneme - Considered a fundamental sound in the production of speech.

1.3.4 IPA - The International Phonetic Alphabet is a system by which sounds can be written to express phonemes.

1.3.5 Sampa - A sampa is a transformation of the IPA to allow easy typing on a keyboard. Some languages have a specialized sampa, such as English's Arpabet or romaji for Japanese.

1.3.6 Music Score - The definition of how a song should be sung/sound.

1.3.7 HTS singing label format - A label format used by the HMM/DNN-based Speech Synthesis System, or HTS for short.

1.3.8 F0 - The fundamental frequency or perceived pitch of speech or singing.

2. General Description

2.1 Product Perspective

This application is meant to be used by any individual who is interested in creating a singing dataset for machine learning purposes. For a general interaction model, please refer to Figure 1.

2.2 Product Functions

2.2.1 Data Handling Features

The tool will allow the user to import singing data which consists of audio data, lyrics, notes, and score.

2.2.2 Time Level User Interface Features

The tool will display the raw audio data as a waveform.

It will also display the spectrogram and pitch of the audio data.

Shortcuts for editing labels, copying labels, pasting labels, deleting labels, duplicating labels and selecting labels will be provided to the user as both keyboard shortcuts and in a dropdown menu when the user right clicks.

Text boxes will be provided for the user to input and edit properties of phonemes and other labels.

2.2.3 Score Level User Interface Features

Display the music data as piano roll and overlay pitch data on top. The user can draw in notes and input lyrics/phonemes. There will be a grid that can be offset to properly align it to the audio. Allow specification of tempo and key signature changes.

2.2.4 Output Feature

The tool will output the labelled data in a user specified format. It will be featured enough to satisfy the HTS singing label format.

2.2.5 Automatic Labelling Features

The tool will allow the user to automatically detect phonemes.

The tool will allow the user to automatically align the generated

2.2.6 Saving and Loading Features

The tool will allow the user to save their work as a project file.

The tool will allow for the saving of all the labels as well as the imported songs.

The tool will allow the user to load a previously saved project and resume working on it.

The tool will provide an auto-save feature.

2.2.7 Automatic Conversion Tools

There will be subtools that allow for conversion between certain feature layers. Such as words to phonemes using a dictionary system, phonemes to syllables, and syllables to notes.

2.3 Targeted Audience

The user of this tool is expected to be familiar with the field of vocal synthesis and have prior experience with labelling singing data. The user is expected to know which label types they require, as well as what each label type represents. This tool targets members of the vocal synthesis field who desire to create their own training data at both a research and hobby level.

3. Specific Requirements

3.1 Data I/O

3.1.1 Singing Data Format

The program must be able to read uncompressed wav files as the source data for labeling.

3.1.2 Music Score Import Formats

The music score must be able to be imported from an ust, vsqx3, vsqx4, and musicxml at minimum.

3.1.3 Output Format

The output format will be user defined. This definition includes what label feature to iterate over, what to do on error, static text, and dynamic entries which are filled in using inputted code snippets.

3.1.4 Project File

The project file will actually consist of several files to facilitate collaboration using version control. This should include paths to the audio, label data, output formats, and language definitions.

3.2 Language Definition

3.2.1 Phoneme Definitions

The user will be able to define the sampa they are using as it relates to the IPA. Each user sampa can map the multiple IPA letters, but an IPA letter can only belong to one user defined phoneme.

3.2.2 Word Dictionary

The conversion of words to phonemes will be done using a dictionary system.

3.2.3 Syllable Definition

The user can input a weighted table/graph that models the phonotactic structure of the language.

3.3 Time Level User Interface

3.3.1 Waveform

The ability to zoom in and out, scroll, select regions, and playback regions are necessary.

3.3.2 Spectrogram

The spectrogram provides in-depth information on the phonetic properties to the user. Customization of color, contrast, hop size, and sample size will be needed. It will also display the f0 to aid in identifying note transitions.

3.3.3 Feature Layers

These layers will be the method by which the user can input time markers to input labels. There will be built in layers for phoneme, syllable, and notes. The user can add additional layers from a list of preset types, including auto-number layers and string layers. These layers can have a parent-child relationship to aid in grouping and access.

3.3.4 Project Feature Layer

One layer at a time can be overlaid onto the waveform and spectrogram. There will be an option to have the regions cycle through colors to make them more distinct from each other.

3.3.5 Shortcuts

The editor should support common tasks such as undo, redo, cut, copy, paste, and duplicate.

3.4 Score Level User Interface

3.4.1 Piano Role

Must have enough octaves to cover the theoretical range of human singing. The rows will be color coded for sharps/flats. It will have a grid so the notes can snap to it. This grid can be modified for the note to snap to different sizes. The grid can also be offset to align it to the audio.

3.4.2 Feature Overlays

An overlay of the waveform and f0 will be on display to aid manual input.

3.4.3 Notes

Contain words and phonemes. Can be resized and cut. Phonemes can be locked separate to words. Each note can have additional features assigned to them such as tempo.

3.4.4 Playback

Be able to select regions and play back. Also generate tone for passed notes as a reference.

3.5 Conversion Tools

3.5.1 Lyrics to Phonemes

Using the provided dictionary data, convert lyrics into phonemes to be aligned.

3.5.2 Phonemes to Syllables to Notes

Using the provided graph, output a list of syllables and assign each a note. Estimate note identity based on average pitch at that moment.

3.6 Software restrictions

3.6.1 Portability

This application should be able to natively run on Windows, Mac, and Linux.

3.7 Stretch Requirements

3.7.1 Grapheme to Phoneme

By inputting the written word into some algorithm or structure, output predicted phonemes

3.7.2 Robust Linguistic Analysis

Generate syllables using a more featured toolset. For example, a decision tree and rules based on phoneme properties.

Appendix

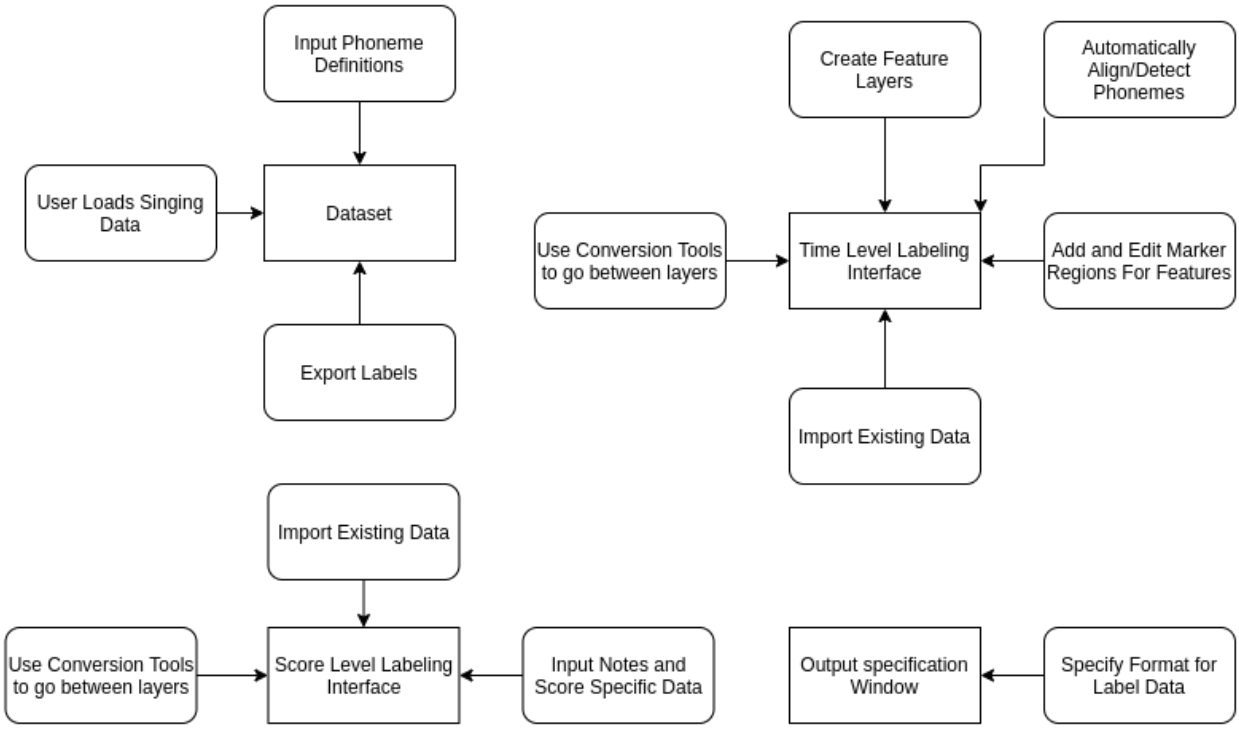

Figure 1