**Author to the papers: Steve Rubin (**[**http://ssrubin.com/**](http://ssrubin.com/)**)**

* **Tools available:** Radio tool (Radiotool is a python library that aims to make it easy to create audio by piecing together bits of other audio files. This library was originally written to enable my research in audio editing user interfaces, but perhaps someone else might find it useful.)

Link:

1. <https://github.com/ucbvislab/radiotool>
2. <http://ucbvislab.github.io/radiotool/>

* **LEARNING TO GENERATE MUSIC WITH SENTIMENT**

(Paper: <http://www.lucasnferreira.com/papers/2019/ismir-learning.pdf>

Article: <https://www.semanticscholar.org/paper/Learning-to-Generate-Music-With-Sentiment-Ferreira-Whitehead/5dc16d4b89e67ef8436e23343addbca55ed5d90a>

Annotation tool: <https://github.com/lucasnfe/adl-music-annotation>

VG midi Dataset: <https://github.com/lucasnfe/vgmidi>)

* **Generating Emotionally Relevant Musical Scores for Audio Stories**

(Paper: <http://vis.berkeley.edu/papers/emotionscores/emotionscores.pdf>

Article: <http://vis.berkeley.edu/papers/emotionscores/>

Demo Video: <https://www.youtube.com/watch?time_continue=2&v=hrHprLYDkN4&feature=emb_logo>)

1. Hand-labeling the music. If users wish to personalize the emotion labels of the speech, they can listen to and assign labels for each music segment (Figure 3) As in hand-labeling the speech, this method is preferable for users that have time and desire a personalized musical score.
2. Crowd-labeling the music. To crowd-label the emotions of the music, our system asks workers on Mechanical Turk to listen to and label the emotions of the music segments for an entire track (Figure 3). Our system then selects a final emotion labeling by finding the worker’s labeling that best represents all of the worker labelings (see earlier section on Crowd-labeling the speech).
3. Automatically labeling the music: SCORE GENERATION ALGO

* **UnderScore: Musical Underlays for Audio Stories**

(Paper: <http://vis.berkeley.edu/papers/underscore/underscore.pdf>

Article: <http://vis.berkeley.edu/papers/underscore/>

Product: <http://vis.berkeley.edu/papers/underscore/demo/>

Demo Video: <https://www.youtube.com/watch?v=ZMazbW8w9VI>)

DESIGN GUIDELINES FOR MUSICAL UNDERLAYS:

1. Marking speech emphasis points
2. Selecting music.
3. Aligning music and speech.
4. Adjusting dynamics (The goal of adjusting the underlay dynamics is to further heighten the impact of the change point in the music while ensuring that the speech is clearly audible.)

ALGORITHMIC METHODS: UnderScore relies on a variety of audio analysis algorithms to facilitate underlay creation. Audio analysis is a well studied area [29] and the algorithms typically operate on features that capture specific properties of music and speech such as volume, harmonic content, and timbre. The features are usually computed on standard 44.1 kHz pulse code modulated (PCM) wave files with their left and right channels summed together. Our tools use the following features:

1. RMS energy (measure the perceived loudness of speech and music using root mean squared (RMS) energy)
2. Chroma (measure the harmonic content of music using chroma vectors)
3. MFCCs (measure the timbre of music (which often correlates with instrumentation) by computing its mel frequency cepstral coefficients (MFCCs))

* **Tools for Creating Audio Stories**

(Paper: <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-237.pdf>

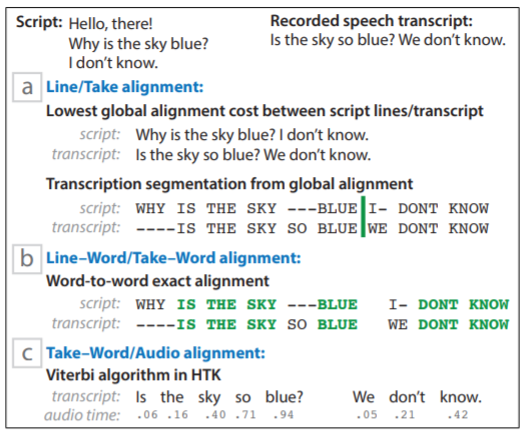
Article: <https://www.semanticscholar.org/paper/Tools-for-Creating-Audio-Stories-Rubin/815b7240e2b4339933cf40d3c699790d4efd5602>

Article: <http://vis.berkeley.edu/papers/audiostories/>

Demo Video: <https://www.youtube.com/watch?v=RHtI4G5L31w&feature=emb_logo>)

METHOD:

1. *Script Analysis* provides suggestions for words in the script to emphasize by applying rules proposed by voiceover experts [45]. Our system suggests users emphasize: descriptive words (adjectives), proper nouns, action verbs, and words at the beginnings and ends of sentences



1. *Transcript-guided recording* While the user is speaking, our system records the audio and runs Mac OSX’s Enhanced Dictation tool in the background to compute a text transcript.
2. *Speech feedback:* While pitch accents are necessary for emphasis, they are not sucient. For example, a word preceding a pitch accented word may contain a much larger pitch accent, overpowering any emphasis that a listener may otherwise hear in the second word. We apply the AuToBI pitch accent detector to nd a subset of words that have pitch accents
3. *Speech resynthesis:* First, for each missed target word, our system checks if the user properly emphasized the word in another take of the same script line. If she did, Narration Coach creates a new version of the current take with the target words replaced with the audio of the properly emphasized words. Second, if the user did not emphasize the target word in another take, our system applies a parameter-based resynthesis method.
4. *Constructing the final narration*