References:
Organized by chapter

This section lists the references segmented by chapter. Note that the book's scripts can be downloaded from the GitHub repository and all code contained within this repository are licensed under the Apache 2.0 license.

Frequently Asked Questions (FAQs)

Core reference:

1. Voicebook repository (GitHub). https://github.com/jim-schwoebel/voicebook

Introduction: Defining Voice Computing

Figure I.2.1: Timeline of Voice Computing: Historical Periods.

2. Figure drawn in sketch.io by author.

Table I.2.1: History of Voice Computing.

3. Hyoid bone. https://en.wikipedia.org/wiki/Hyoid_bone
4. Hyoid bone. https://www.livescience.com/7468-hyoid-bone-changed-history.html
5. Language. https://en.wikipedia.org/wiki/Language
6. Musical notation. https://en.wikipedia.org/wiki/Musical_notation
7. Printing press. https://en.wikipedia.org/wiki/Printing_press
8. Wolfgang von Kempelen's speaking machine. https://en.wikipedia.org/wiki/Wolfgang_von_Kempelen%27s_speaking_machine
9. Luigi Aloisio Galvani. https://en.wikipedia.org/wiki/Luigi_Galvani
10. Music box. https://en.wikipedia.org/wiki/Music_box
11. Charles Wheatsone. https://en.wikipedia.org/wiki/Charles_Wheatstone
12. Phonautograph. https://en.wikipedia.org/wiki/Phonautograph
13. Player piano. https://en.wikipedia.org/wiki/Player_piano#/media/File:Pianola1.JPG
14. Alexander Graham Bell. https://en.wikipedia.org/wiki/Alexander_Graham_Bell
15. Phonograph. https://en.wikipedia.org/wiki/Phonograph
16. Carbon Microphone. https://en.wikipedia.org/wiki/Carbon_microphone
17. David Edward Hughes. https://en.wikipedia.org/wiki/David_Edward_Hughes
18. Thomas Edison. https://en.wikipedia.org/wiki/Thomas_Edison
19. Emil Berliner. https://en.wikipedia.org/wiki/Emile_Berliner
20. Santiago Ramón y Cajal. https://en.wikipedia.org/wiki/Santiago_Ram%C3%B3n_y_Cajal
21. Cajal drawings. https://hyperallergic.com/361875/a-19th-century-illustrated-tour-through-the-brain-goes-on-tour/
22. Valdemar Poulsen. https://en.wikipedia.org/wiki/Valdemar_Poulsen
23. Sound film. https://en.wikipedia.org/wiki/Sound_film
24. Microphone. https://en.wikipedia.org/wiki/Microphone
25. Nathaniel Baldwin. https://en.wikipedia.org/wiki/Nathaniel_Baldwin
26. Western Electric. https://en.wikipedia.org/wiki/Western_Electric
27. Chester Williams Rice. https://en.wikipedia.org/wiki/Chester_Williams_Rice
28. Edward W. Kellogg. https://en.wikipedia.org/wiki/Edward_W._Kellogg
29. The Jazz Singer. https://en.wikipedia.org/wiki/The_Jazz_Singer
30. Ribbon Microphone. https://en.wikipedia.org/wiki/Ribbon_microphone

31. Magnetic Tape. https://en.wikipedia.org/wiki/Magnetic_tape
32. Past, present, and future of speech recognition technology. https://medium.com/swlh/the-past-present-and-future-of-speech-recognition-technology-cf13c179aaf
33. Koss Corporation. https://en.wikipedia.org/wiki/Koss_Corporation
34. IBM Shoebox. https://en.wikipedia.org/wiki/IBM_Shoebox
35. Compact Cassette. https://en.wikipedia.org/wiki/Compact_Cassette
36. Electret microphone. https://en.wikipedia.org/wiki/Electret_microphone
37. Dynamic microphone. https://en.wikipedia.org/wiki/Microphone#Dynamic_microphone
38. Sony. https://en.wikipedia.org/wiki/Sony
39. Bop Till You Drop. https://en.wikipedia.org/wiki/Bop_till_You_Drop
40. Personal computer. https://en.wikipedia.org/wiki/Personal_computer
41. Philips. https://en.wikipedia.org/wiki/Philips
42. Silicon microphone. https://en.wikipedia.org/wiki/Microphone
43. PlainTalk. https://en.wikipedia.org/wiki/PlainTalk
44. Ray Kurzweil. https://en.wikipedia.org/wiki/Ray_Kurzweil
45. IBM Tangora. https://en.wikipedia.org/wiki/Timeline_of_speech_and_voice_recognition
46. James McClelland (psychologist). https://en.wikipedia.org/wiki/James_L._McClelland
47. David Rumelhart. https://en.wikipedia.org/wiki/David_Rumelhart
48. Worlds of Wonder (toy company). https://en.wikipedia.org/wiki/Worlds_of_Wonder_(toy_company)
49. Digital Compact Cassette. https://en.wikipedia.org/wiki/Digital_Compact_Cassette
50. The internet. https://en.wikipedia.org/wiki/Internet
51. Nuance Communications. https://en.wikipedia.org/wiki/Nuance_Communications
52. Python programming language. https://en.wikipedia.org/wiki/Python_(programming_language)
53. Guido van Rossum. https://en.wikipedia.org/wiki/Guido_van_Rossum
54. WorldWideWeb. https://en.wikipedia.org/wiki/WorldWideWeb
55. Tim Berners-Lee. https://en.wikipedia.org/wiki/Tim_Berners-Lee
56. Sound eXchange (SoX). https://en.wikipedia.org/wiki/SoX
57. Waveform Audio File Format (WAV). https://en.wikipedia.org/wiki/WAV
58. CMU Sphinx. https://en.wikipedia.org/wiki/CMU_Sphinx
59. DVD. https://en.wikipedia.org/wiki/DVD
60. FLAC. https://en.wikipedia.org/wiki/FLAC
61. FFmpeg. https://en.wikipedia.org/wiki/FFmpeg
62. NLTK. https://github.com/nltk/nltk
63. Sound card. https://en.wikipedia.org/wiki/Sound_card
64. iTunes Store. https://en.wikipedia.org/wiki/ITunes_Store
65. Skype. https://en.wikipedia.org/wiki/Skype
66. Blue ray disks. https://en.wikipedia.org/wiki/Blu-ray
67. Google translate. https://en.wikipedia.org/wiki/Google_Translate
68. National security agency. https://en.wikipedia.org/wiki/National_Security_Agency

69. GOOG-411. https://en.wikipedia.org/wiki/GOOG-411
70. Scikit-learn. https://en.wikipedia.org/wiki/Scikit-learn
71. David Cournapeau. https://en.wikipedia.org/wiki/David_Cournapeau
72. Mini and micro-cards. https://en.wikipedia.org/wiki/Secure_Digital#Micro
73. Google Chrome. https://en.wikipedia.org/wiki/Google_Chrome
74. Voice search. https://en.wikipedia.org/wiki/Voice_search
75. Kaldi. https://en.wikipedia.org/wiki/Kaldi
76. Kaldi website. http://kaldi-asr.org/
77. Siri. https://en.wikipedia.org/wiki/Siri
78. Opus (audio format). https://en.wikipedia.org/wiki/Opus_(audio_format)
79. PS4. https://en.wikipedia.org/wiki/PlayStation_4
80. Amazon Alexa. https://en.wikipedia.org/wiki/Amazon_Alexa
81. Cortana. https://en.wikipedia.org/wiki/Cortana
82. Tensorflow. https://en.wikipedia.org/wiki/TensorFlow
83. Speech Recognition. https://en.wikipedia.org/wiki/Speech_recognition
84. Librosa library. http://conference.scipy.org/proceedings/scipy2015/pdfs/brian_mcfee.pdf
85. History of Microphones. https://thoughtco.com/history-of-microphones-1992144.
86. History of recording. http://www.charm.rhul.ac.uk/history/p20_4_1.html
87. Liquid and condenser microphones. https://microphone238.weebly.com/liquid--condenser-microphones.html
88. How MEMS microphones work. https://www.edn.com/design/analog/4430264/Basic-principles-of-MEMS-microphones-
89. Number of android users. https://www.theverge.com/2017/5/17/15654454/android-reaches-2-billion-monthly-active-users
90. The Machines that Learned to Listen. http://www.bbc.com/future/story/20170214-the-machines-that-learned-to-listen
91. From Audrey to Siri. http://www.icsi.berkeley.edu/pubs/speech/audreytosiri12.pdf
92. Timeline of speech and voice recognition. https://en.wikipedia.org/wiki/Timeline_of_speech_and_voice_recognition
93. Speech recognition before Siri and what is to come. https://www.androidauthority.com/speech-recognition-life-before-siri-and-whats-to-come-67994/
94. PyAudioAnalysis. http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0144610
95. Keras. https://en.wikipedia.org/wiki/Keras
96. AirPods. https://en.wikipedia.org/wiki/AirPods
97. Google Assistant. https://en.wikipedia.org/wiki/Google_Assistant
98. Wavenet. https://en.wikipedia.org/wiki/WaveNet
99. Homepod. https://en.wikipedia.org/wiki/HomePod

Figure 1.3.1: Smart speaker market.

100. Figure taken from Voicebot.ai. https://voicebot.ai/2018/06/03/u-s-smart-speaker-market-share-apple-debuts-at-4-1-amazon-falls-10-points-and-google-rises/

Table I.3.1: Industries Affected by Voice Computing.

101. Voice computing growth rates. Taken from 4-traders and Businesswire.
102. Machine learning CAGR / market size. Taken from Healthitanalytics.com
103. Voice assistants CAGR/ market size. Taken from Marketresearchfuture.
104. Connected cars CAGR / market size. Taken from Globalnewswire.
105. Call centers CAGR / market size. Taken from PRNewswire.
106. Speech recognition CAGR / market size. Taken from Statistica and MarketsandMarkets.
107. Voice biometrics CAGR / market size. Taken fromTechnavio and Techsciresearch.
108. Digital health CAGR / market size. Taken from Staista
109. Voice messaging CAGR / market size. Taken from IDC.
110. Text to speech software CAGR / market size. Taken from MarketsandMarkets and MarketsandMarkets.
111. Smart Agriculture CAGR / market size. Taken from Zion Market Research.
112. Cloud computing infrastructure CAGR / market size. Taken from Forbes and Marketsandmarkets.
113. Connected home CAGR / market size. Taken from Statista.
114. Digital logistics CAGR / market size. Taken from PRNewswire.
115. Recruiting CAGR / market size. Taken from Statista and PRNewswire.
116. Interactive Voice Response CAGR / market size. Taken from MarketsandMarkets and Marketresearchfuture.
117. Market research CAGR / market size. Taken from Statista and Ibisworld.
118. Radio broadcast advertising CAGR / market size. Taken from PwC.

Figure I.4.1: Data scientists and degree requirements.

119. Data scientists and degree requirements. https://www.forbes.com/sites/louiscolumbus/2017/05/13/ibm-predicts-demand-for-data-scientists-will-soar-28-by-2020/#24eb12f47e3b

Other references

120. Best graduate degree CS programs. https://www.usnews.com/best-graduate-schools/top-science-schools/computer-science-rankings
121. Eric Schrock. The Future of Enterprise Voice Computing. https://medium.com/@eschrock/the-future-of-enterprise-voice-computing-890f28763c6a

122. Voice computing growth stats. Taken from 4-traders and Businesswire.

123. Past, present, and future of speech recognition technology. https://medium.com/swlh/the-past-present-and-future-of-speech-recognition-technology-cf13c179aaf

Chapter 1: Fundamentals

Chapter 1: Voice Computing Fundamentals

Table 1.1.1: List of microphones and their utility.

124. Condenser microphone. https://en.wikipedia.org/wiki/Condensor_microphone
125. Dynamic microphone.https://en.wikipedia.org/wiki/Dynamic_microphone
126. Ribbon microphone. https://en.wikipedia.org/wiki/Ribbon_microphone
127. Condenser MEMS microphone. https://en.wikipedia.org/wiki/Microphone
128. Piezoelectric MEMS microphone. https://en.wikipedia.org/wiki/Microphone
129. Electret microphone. https://en.wikipedia.org/wiki/Electret_microphone
130. Noise-cancelling microphone. https://en.wikipedia.org/wiki/Noise-canceling_microphone

Figure 1.1.1: Sample figure illustrating how a codec works.

125. Figure drawn in sketch.io by author.
126. Hard disk (image). https://en.wikipedia.org/wiki/Hard_disk_drive
127. Codec example. https://www.pcmag.com/encyclopedia/term/56334/codec-examples

Figure 1.1.2: Difference between mono vs. stereo signals.

128. Figure drawn in sketch.io by author.
129. Difference between mono and stereo. https://music.stackexchange.com/questions/24631/what-is-the-difference-between-mono-and-stereo
130. The 7 Best PC Sound Cards to Buy in 2018. https://www.lifewire.com/best-pc-sound-cards-833111

Table 1.1.3: Common Audio Coding Formats.

131. Timeline of audio formats. https://en.wikipedia.org/wiki/Audio_format
132. MP3 format. https://en.wikipedia.org/wiki/MP3
133. WAV format. https://en.wikipedia.org/wiki/WAV
134. AAC format. https://en.wikipedia.org/wiki/Advanced_Audio_Coding
135. FLAC format. https://en.wikipedia.org/wiki/FLAC
136. OPUS format. https://en.wikipedia.org/wiki/Opus_(audio_format)

Table 1.1.4: Speaker types and their utility.

137. Headphones. https://en.wikipedia.org/wiki/Headphones
138. Dynamic speaker. https://en.wikipedia.org/wiki/Loudspeaker
139. Piezoelectric speaker. https://en.wikipedia.org/wiki/Piezoelectric_speaker
140. Electrostatic loudspeaker. https://en.wikipedia.org/wiki/Electrostatic_loudspeaker
141. Loudspeaker. https://en.wikipedia.org/wiki/Loudspeaker
142. Ribbon loudspeaker. https://en.wikipedia.org/wiki/Loudspeaker

Table 1.3.1: How to Read/Write Audio files.

143. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Table 1.4.1: Libraries for manipulating audio files.

144. Pydub. https://github.com/jiaaro/pydub
145. Pysox. https://github.com/rabitt/pysox
146. Wave. https://docs.python.org/3/library/wave.html#wave.open
147. Librosa. https://librosa.github.io/librosa/generated/librosa.core.load.html
148. Scipy. https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.io.wavfile.read.html

Table 1.4.2: Useful Sox Commands in Terminal.

149. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Table 1.4.3: Useful Pydub Commands.

150. Pydub. https://github.com/jiaaro/pydub
151. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.5.1: How to Playback Audio FIles Synchronously.

152. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.5.2: How to Playback Audio Files Asynchronously.

153. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.6.1: How to check available microphones.

154. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.6.2: How to record audio files synchronously.

155. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.6.3: How to record audio files asynchronously.

156. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.7.1: How to convert audio files using ffmpy module.

157. Ffmpy. http://ffmpy.readthedocs.io/en/latest/
158. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.7.2: How to convert audio files using FFmpeg.

159. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals
160. FFmpeg. https://www.ffmpeg.org/

Table 1.8.1: Comparison between open source and proprietary transcription engines.

161. PocketSphinx. https://github.com/cmusphinx/pocketsphinx-python
162. Deepspeech. https://github.com/mozilla/DeepSpeech
163. Google Speech API. https://cloud.google.com/speech-to-text/docs/
164. Watson Speech to Text. https://www.ibm.com/watson/services/speech-to-text/
165. Bing Speech. https://azure.microsoft.com/en-us/services/cognitive-services/speech/

Figure 1.8.1: How to transcribe with Pocketsphinx.

166. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.8.2: Output from Figure 1.8.1.

167. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.8.3: How to setup environment variables for the Google Speech API.

168. Stackoverflow. https://stackoverflow.com/questions/30461201/how-do-i-edit-path-bash-profile-on-osx

Figure 1.8.4: How to transcribe with Google Speech API.

169. Google Cloud Speech-to-text. https://cloud.google.com/speech-to-text/
170. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.9.1: How to playback text with default settings.

171. Pyttsx3. https://github.com/nateshmbhat/pyttsx3
172. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.9.2: How to play back text with default settings.

172. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Figure 1.9.3: How to play back text using Google TTS.

173. Google TTS. https://en.wikipedia.org/wiki/Google_Text-to-Speech
174. Voicebook repository: Chapter 1. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_1_fundamentals

Python libraries

175. FFmpeg. https://www.ffmpeg.org/
176. SoX. http://sox.sourceforge.net/Docs/Features
177. Librosa. https://librosa.github.io/librosa/generated/librosa.core.load.html
178. Opuslib. https://github.com/OnBeep/opuslib
179. Os. https://docs.python.org/3/library/os.html
180. Pydub. https://github.com/jiaaro/pydub
181. Pygame. https://www.pygame.org/news
182. Pykaldi. https://github.com/pykaldi/pykaldi
183. PySoundfile. https://pysoundfile.readthedocs.io/en/0.9.0/
184. Pysox - https://github.com/rabitt/pysox
185. Scipy. https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.io.wavfile.read.html
186. Simpleaudio. https://pypi.org/project/simpleaudio/
187. Snack Soundkit. http://www.speech.kth.se/snack/
188. SpeechRecognition. https://pypi.org/project/SpeechRecognition/
189. Wave. https://docs.python.org/3/library/wave.html#wave.open

Tutorials

190. How to setup Google Client Libraries (Google Speech API). https://cloud.google.com/speech-to-text/docs/quickstart-client-libraries
191. How to setup Google TTS. https://cloud.google.com/text-to-speech/
192. Wave generation in python (WAVE library). http://blog.acipo.com/wave-generation-in-python/

Stack Overflow Conversations

193. Converting stereo to mono format. https://stackoverflow.com/questions/30401042/stereo-to-mono-wav-in-python

194. Debugging Python opus. https://stackoverflow.com/questions/17728706/python-portaudio-opus-encoding-decoding
195. Difference between synchronous and asynchronous programming. https://stackoverflow.com/questions/748175/asynchronous-vs-synchronous-execution-what-does-it-really-mean
196. How to record audio in python. https://stackoverflow.com/questions/892199/detect-record-audio-in-python
197. How to play back audio in python. https://stackoverflow.com/questions/307305/play-a-sound-with-python
198. How to train Google’s WaveNet model. https://stackoverflow.com/questions/41679110/how-to-use-tensorflow-wavenet

Other links and references.

199. Audio codec. https://en.wikipedia.org/wiki/Audio_codec
200. Analog-to-digital converter. https://en.wikipedia.org/wiki/Analog-to-digital_converter
201. Analog signal. https://en.wikipedia.org/wiki/Analog_signal
202. Audio format. https://en.wikipedia.org/wiki/Audio_coding_format
203. Capacitor. https://en.wikipedia.org/wiki/Capacitor
204. Digital-to-analog converter. https://en.wikipedia.org/wiki/Digital-to-analog_converter
205. Difference between stereo and mono signals. https://music.stackexchange.com/questions/24631/what-is-the-difference-between-mono-and-stereo
206. Environment variable. https://en.wikipedia.org/wiki/Environment_variable
207. FFmpeg. https://en.wikipedia.org/wiki/FFmpeg
208. Kaldi. http://kaldi-asr.org/
209. Microphone. https://en.wikipedia.org/wiki/Microphone
210. Opus. http://opus-codec.org
211. Podcasting equipment. https://www.podcastinsights.com/podcast-equipment/
212. Pressure. https://en.wikipedia.org/wiki/Pressure
213. Signal. https://en.wikipedia.org/wiki/Signal
214. Sound card. https://en.wikipedia.org/wiki/Sound_card
215. Sound pressure. https://en.wikipedia.org/wiki/Sound_pressure
216. Sound localization. https://en.wikipedia.org/wiki/Sound_localization
217. SoX. https://en.wikipedia.org/wiki/SoX
218. Stereographic projection. https://en.wikipedia.org/wiki/Stereographic_projection
219. Useful FFmpeg commands. https://www.labnol.org/internet/useful-ffmpeg-commands/28490/
220. Useful SoX Commands. https://www.thegeekstuff.com/2009/05/sound-exchange-sox-15-examples-to-manipulate-audio-files/

221. Wolfram alpha, word count from characters. https://m.wolframalpha.com/input/?i=1%2C000%2C000+characters&lk=3

Chapter 2: Collection

Table 2.1.1: List of microphones and their utility.

222. Condenser microphone. https://en.wikipedia.org/wiki/Condensor_microphone
223. Dynamic microphone.https://en.wikipedia.org/wiki/Dynamic_microphone
224. Ribbon microphone. https://en.wikipedia.org/wiki/Ribbon_microphone
225. Condenser MEMS microphone. https://en.wikipedia.org/wiki/Microphone
226. Piezoelectric MEMS microphone. https://en.wikipedia.org/wiki/Microphone
227. Electret microphone. https://en.wikipedia.org/wiki/Electret_microphone
228. Noise-cancelling microphone. https://en.wikipedia.org/wiki/Noise-canceling_microphone

Table 2.1.2: Distance from microphone and volume drop.

229. N/A. Figure constructed from author's intuition.

Table 2.1.3: Recording environments and noise profiles

230. N/A. Figure constructed from author's intuition.

Table 2.1.4: Common audio recording settings

231. Nyquist rate. https://en.wikipedia.org/wiki/Nyquist_rate

Table 2.2.1: Types of microphone arrays.

232. UMA-8 microphone array. https://www.minidsp.com/products/usb-audio-interface/uma-8-microphone-array
233. Respeaker. Amazon link.
234. Matrix creator. https://www.element14.com/community/docs/DOC-86328
235. Blue Yeti microphone. Amazon link.

Figure 2.3.1: How mixers work and connect to voice computers.

236. Figure created by author.
237. Sound card. https://en.wikipedia.org/wiki/Sound_card
238. Hard disk. https://en.wikipedia.org/wiki/Hard_disk_drive

Figure 2.4.1: Active-synchronous (AS) mode

239. as_record.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.4.2: Active-asynchronous (AA) mode.

240. aa_record.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.4.3: Passive synchronous (PS) mode.

241. ps_record.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.4.4: Passive-asynchronous (PA) mode.

242. pa_record.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.4.5: Active-passive-asynchronous (APS) mode.

243. aps_record.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.4.6: Active-passive-asynchronous (APA) mode.

244. record_apa.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.1: How to remove noise from file.

245. Remove_noise.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.2: How to change volume.

246. Change-volume.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.3: How to trim audio files.

247. Trim_audio.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.4: How to combine audio files.

248. Combine.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.5: How to transcode audio files.

249. Transcode.py.
   https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.6: How to convert sample rates.

250. Change_samplerate.py
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.7: How to change number of channels.

251. Change_channels.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.8: How to trim silence.

252. Trim_silence.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.5.9: List of software packages that can help analyzing audio files.

253. TenTopReviews.
http://www.toptenreviews.com/software/multimedia/best-voice-recording-software/

Figure 2.6.1: How to implement speaker diarization in the terminal.

254. Diarize.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.6.2: Output of speaker diarization script.

255. Picture taken by author on Macbook pro.
256. Diarize.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.6.3: Transcript output of speaker diarization script.

257. Diarize.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.7.1: Converting voice files to .FLAC format.

258. Convert_flac.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.7.2: Converting voice files to .OPUS format.

259. Convert_opus.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.7.3: Unpacking compressed .FLAC or .OPUS files.

260. Unpack_files.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Table 2.7.1: List of storage mediums and cost.

261. USB flash drive. Amazon.

262. SD card. Amazon.
263. Hard disk. Amazon.
264. Cloud provider. AWS pricing.

Figure 2.7.4: Uploading files to a FTP server.

265. Store_ftp.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Figure 2.7.5: Uploading files to google storage.

266. Store_gcp.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Table 2.8.1: Ways to publish non-streaming voice content.

267. Gmail. https://en.wikipedia.org/wiki/Gmail
268. iPhone. https://en.wikipedia.org/wiki/IPhone
269. Facebook. https://en.wikipedia.org/wiki/Facebook
270. YouTube. https://en.wikipedia.org/wiki/YouTube
271. Google Assistant. https://en.wikipedia.org/wiki/Google_Assistant
272. Siri. https://en.wikipedia.org/wiki/Siri
273. Cortana. https://en.wikipedia.org/wiki/Cortana
274. Amazon Alexa. https://en.wikipedia.org/wiki/Amazon_Alexa
275. iTunes. https://en.wikipedia.org/wiki/ITunes
276. Dropbox (Service). https://en.wikipedia.org/wiki/Dropbox_(service)
277. Google Drive. https://en.wikipedia.org/wiki/Google_Drive
278. LinkedIn.https://en.wikipedia.org/wiki/LinkedIn
279. Twitter. https://en.wikipedia.org/wiki/Twitter
280. Spotify. https://en.wikipedia.org/wiki/Spotify
281. Pandora Radio. https://en.wikipedia.org/wiki/Pandora_Radio
282. Soundcloud. https://en.wikipedia.org/wiki/SoundCloud
283. Slack Technologies. https://en.wikipedia.org/wiki/Slack_Technologies
284. Kaggle. https://en.wikipedia.org/wiki/Kaggle

Table 2.8.2: Ways to publish streaming voice content.

285. Cell phone. https://en.wikipedia.org/wiki/Mobile_phone
286. Facetime. https://en.wikipedia.org/wiki/FaceTime
287. NPR. https://en.wikipedia.org/wiki/NPR
288. CNN. https://en.wikipedia.org/wiki/CNN
289. Facebook Live.
https://en.wikipedia.org/wiki/List_of_Facebook_features#Live_streaming
290. Hangouts on Air.
https://en.wikipedia.org/wiki/Google%2B#Hangouts_and_Hangouts_On_Air
291. Slack calling feature. https://en.wikipedia.org/wiki/Slack_Technologies

292. GoToMeeting. https://en.wikipedia.org/wiki/GoToMeeting
293. Amazon Chime. https://aws.amazon.com/chime/
294. Google Duo. https://en.wikipedia.org/wiki/Google_Duo
295. Zoom Video Communications. https://en.wikipedia.org/wiki/Zoom_Video_Communications
296. Uberconference. https://en.wikipedia.org/wiki/UberConference

Figure 2.9.1: Using the MEMUPPS mindset to label audio files.

297. label_memups.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_2_collection

Python libraries

298. Pysoundfile. https://pysoundfile.readthedocs.io/en/0.9.0/
299. Sounddevice. https://python-sounddevice.readthedocs.io/en/0.3.11/
300. OS module. https://docs.python.org/3/library/os.html
301. Sox. http://sox.sourceforge.net/
302. PyAudioAnalysis. https://github.com/tyiannak/pyAudioAnalysis/wiki/5.-Segmentation
303. FFmpeg. https://www.ffmpeg.org/
304. Opus codec. http://opus-codec.org/
305. FLAC codec. https://xiph.org/flac/
306. Google Cloud Storage client library. https://cloud.google.com/storage/docs/reference/libraries
307. FTPlib. https://docs.python.org/3/library/ftplib.html
308. Shutil. https://docs.python.org/3/library/shutil.html
309. Zipfile. https://docs.python.org/3/library/zipfile.html
310. Pytaglib. https://pypi.org/project/pytaglib/

Other references

311. [August 2017]. Bankole, Kalonji. Enhancing Voice Recognition with a Matrix Microphone Array. Medium. https://medium.com/kkbankol-events/raspberry-pi-15662c3ca881

Chapter 3: Featurization

Table 3.1.1: Useful numpy commands.

312. Numpy_commands.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.2.1: List of common audio features.

313. Mel spectrogram frequency coefficients. https://dsp.stackexchange.com/questions/12812/mfcc-calculation
314. Mel spectrogram frequency delta coefficients. https://dsp.stackexchange.com/questions/12812/mfcc-calculation
315. Fundamental frequency. https://en.wikipedia.org/wiki/Fundamental_frequency

316. Jitter. https://en.wikipedia.org/wiki/Jitter
317. Shimmer. http://www.wevosys.com/knowledge/_data_knowledge/107.pdf
318. Formant Frequencies. http://clas.mq.edu.au/speech/acoustics/frequency/tubes.html
319. File duration. https://stackoverflow.com/questions/7833807/get-wav-file-length-or-duration
320. Root mean square energy. https://en.wikipedia.org/wiki/Root_mean_square
321. Spectral centroid. https://en.wikipedia.org/wiki/Spectral_centroid
322. Spectral flux. https://en.wikipedia.org/wiki/Spectral_flux
323. Onset strength. http://docs.twoears.eu/en/latest/afe/available-processors/onset-strength/
324. Spectral contrast. https://pdfs.semanticscholar.org/ed54/e0250f106a208cb693d50dbfd13df2d9cd9e.pdf
325. Spectral flatness. https://en.wikipedia.org/wiki/Spectral_flatness
326. Spectral rolloff. https://www.sciencedirect.com/science/article/pii/S1051200412002473
327. Zero crossing rates. https://en.wikipedia.org/wiki/Zero-crossing_rate

Figure 3.2.1: Extracting audio features with librosa library.

328. Librosa_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.2.2: Extracting audio features with PyAudioAnalysis.

329. Pyaudio_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.2.3: Extracting audio features with SoX CLI.

330. Sox_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.1.2: Standardized Audio Research Feature Embeddings.

331. VGGish Audioset embedding. https://github.com/tensorflow/models/tree/master/research/audioset
332. OpenSMILE embeddings. https://audeering.com/technology/opensmile/

Figure 3.2.4: Extracting AudioSet features.

333. Audioset_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.2.7: Extracting all audio features.

334. Audio_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.3.1: List of common text features.

335. Keyword frequency. https://en.wikipedia.org/wiki/Reserved_word
336. Character frequency. https://en.wikipedia.org/wiki/Character_(symbol)
337. Part of speech tag. https://en.wikipedia.org/wiki/Part_of_speech

338. Sentiment polarity. https://textblob.readthedocs.io/en/dev/
339. Brunet's index. https://www.cs.toronto.edu/~kfraser/Fraser15-JAD.pdf
340. Morphological features. https://spacy.io/usage/linguistic-features
341. Syntactic features. https://en.wikipedia.org/wiki/Syntax
342. Named entity recognition. https://en.wikipedia.org/wiki/Named-entity_recognition

Figure 3.3.1: Various ways to transcribe audio data.

343. Record_transcribe.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.3.1: Common ways to manipulate strings in python.

344. String_commands.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.3.2: Extracting text features using nltk.

345. Nltk_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.3.2: Extracting linguistic features using spacy.

346. Spacy_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.3.3: Using word2vec models for featurization with gensim.

347. Gensim_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.3.4: Extracting all text features.

348. Text_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.4.1: Building mixed features.

349. Make_mixed_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.4.1: List of random mixed features from executing make_mixed_features.py.

350. Make_mixed_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.5.1: Table of meta voice features.

351. Table assembled from author's intuition and understanding of various datasets and machine learning models within NeuroLex.
352. Accent dataset. https://www.kaggle.com/rtatman/speech-accent-archive/version/1

Figure 3.5.1: Extracting meta features (71).

353. Meta_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.6.1: List of automated feature selection methods.

354. Principal Component Analysis. https://en.wikipedia.org/wiki/Principal_component_analysis
355. Independent Component Analysis. https://en.wikipedia.org/wiki/Independent_component_analysis
356. K-means clustering. https://en.wikipedia.org/wiki/K-means_clustering
357. Canonical correlation analysis. https://en.wikipedia.org/wiki/Canonical_correlation
358. Partial least squares regression. https://en.wikipedia.org/wiki/Partial_least_squares_regression
359. Manifold learning. https://en.wikipedia.org/wiki/Nonlinear_dimensionality_reduction#Manifold_learning_algorithms
360. Supervised dictionary learning. https://en.wikipedia.org/wiki/Sparse_dictionary_learning
361. Linear discriminant analysis. https://en.wikipedia.org/wiki/Linear_discriminant_analysis
362. Correspondence analysis. https://en.wikipedia.org/wiki/Correspondence_analysis
363. Variational autoencoders. https://en.wikipedia.org/wiki/Autoencoder

Figure 3.6.1: Implementing unsupervised techniques.

364. Dimensionality_reduction.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Figure 3.6.2: Implementing supervised techniques.

365. Dimensionality_reduction.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.6.1: Feature selection methods.

366. Introduction to feature selection methods. https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/

Figure 3.7.1: How to select features.

367. Sleect_features.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_3_featurization

Table 3.7.1: Python libraries for feature selection.

368. Scikit-learn. http://scikit-learn.org/stable/
369. MLpy. http://mlpy.sourceforge.net/
370. Scikit-feature. http://featureselection.asu.edu/
371. ReliefF. https://pypi.org/project/ReliefF/
372. MLfeatureselection. https://github.com/duxuhao/Feature-Selection
373. MLxtend. http://rasbt.github.io/mlxtend/
374. Scikit-rebate. https://github.com/EpistasisLab/scikit-rebate

375. FeatureselectionGA. https://github.com/kaushalshetty/FeatureSelectionGA/blob/master/feature_selection_ga.py
376. Bortua. https://pypi.org/project/Boruta/
377. EL5. http://eli5.readthedocs.io/en/latest/blackbox/permutation_importance.html
378. Yellowbrick. http://www.scikit-yb.org/en/latest/

Research papers

379. ALU, D., ZOLTAN, E., & STOICA, I. C. (2017). Voice Based Emotion Recognition with Convolutional Neural Networks for Companion Robots. SCIENCE AND TECHNOLOGY, 20(3), 222-240.
380. Eyben, F., Scherer, K. R., Schuller, B. W., Sundberg, J., André, E., Busso, C., ... & Truong, K. P. (2016). The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing. IEEE Transactions on Affective Computing, 7(2), 190-202.
381. Gemmeke, J. F., Ellis, D. P., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., ... & Ritter, M. (2017, March). Audio set: An ontology and human-labeled dataset for audio events. In Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on (pp. 776-780). IEEE.
382. Fraser, K. C., Meltzer, J. A., & Rudzicz, F. (2016). Linguistic features identify Alzheimer’s disease in narrative speech. Journal of Alzheimer's Disease, 49(2), 407-422.

Libraries

383. Librosa. https://librosa.github.io/librosa/core.html
384. PyAudioAnalysis. https://github.com/tyiannak/pyAudioAnalysis
385. SoX. http://sox.sourceforge.net/
386. OpenSMILE. https://audeering.com/technology/opensmile/
387. AudioSet. https://github.com/tensorflow/models/tree/master/research/audioset
388. NLTK. https://www.nltk.org/
389. Spacy. https://spacy.io/
390. Gensim. https://radimrehurek.com/gensim/
391. Textblob. https://textblob.readthedocs.io/en/dev/
392. Scikit-learn. http://scikit-learn.org/stable/auto_examples/index.html#general-examples
393. Keras. https://keras.io/
394. Tensorflow. https://www.tensorflow.org/
395. Megaman. https://github.com/mmp2/megaman
396. MLpy. http://mlpy.sourceforge.net/
397. Scikit-feature. http://featureselection.asu.edu/
398. Yellowbrick. http://www.scikit-yb.org/en/latest/
399. Microsoft Azure text feature library. https://azure.microsoft.com/en-us/services/cognitive-services/text-analytics/

Tutorials

400. Feature selection tutorial. https://hub.packtpub.com/4-ways-implement-feature-selection-python-machine-learning/

401. Univariate selection. https://blog.datadive.net/selecting-good-features-part-i-univariate-selection/

402. Collinear features tutorial. https://towardsdatascience.com/a-feature-selection-tool-for-machine-learning-in-python-b64dd23710f0

403. Feature importance tutorial. https://machinelearningmastery.com/feature-selection-in-python-with-scikit-learn/)

404. http://scikit-learn.org/stable/modules/feature_selection.html

405. Grid search tutorial. https://medium.com/open-machine-learning-course/open-machine-learning-course-topic-6-feature-engineering-and-feature-selection-8b94f870706a

406. Visualizing heat maps. https://www.kaggle.com/kanncaa1/feature-selection-and-data-visualization

407. Linear models / LARS LASSO tutorial. http://scikit-learn.org/stable/modules/linear_model.html#lars-lasso

Chapter 4: Data modeling

Figure 4.1.1: Difference between classification and regression models.

408. Figured made by author.

409. Kismet (image). https://en.wikipedia.org/wiki/Kismet_(robot)

Figure 4.1.2: Difference between unsupervised and supervised models.

410. Quora. https://www.quora.com/What-is-the-difference-between-supervised-and-unsupervised-learning-algorithms

Figure 4.1.3: Deep learning model.

411. Deep learning book (image). http://neuralnetworksanddeeplearning.com/chap6.html

Figure 4.2.1: How to build YouTube speech datasets.

412. Make_playlist.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

413. Download_playlist.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.2.1: List of open source datasets.

414. Common voice dataset. https://www.kaggle.com/mozillaorg/common-voice

415. Google Audioset. https://research.google.com/audioset/

416. Urban sound dataset. https://serv.cusp.nyu.edu/projects/urbansounddataset/urbansound8k.html

417. NeuroLex disease dataset. https://github.com/NeuroLexDiagnostics/train-diseases
418. NeuroLex emotion dataset. https://github.com/NeuroLexDiagnostics/train-emotions
419. Karoldvl. https://github.com/karoldvl/ESC-50

Table 4.3.3: Proprietary voice datasets.
420. Linguistic data consortium. https://www.ldc.upenn.edu/

Figure 4.3.1: How to consistently label samples manually.
421. Label_samples.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.3.2: Downloading videos from spreadsheet.
422. Y_scrape.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.3.3: Automated labeling of emotions using YouTube.
423. Figure constructed using author's intuition.
424. OpenCV. https://media.readthedocs.org/pdf/opencv-python-tutroals/latest/opencv-python-tutroals.pdf
425. Scikit-video. http://www.scikit-video.org/stable/io.html

Table 4.4.1: Various machine learning algorithms and their biases.
426. Thoughtful Machine Learning with Python (pages 17 and 18). *Added in various other supervised techniques and removed some unsupervised techniques.*

Figure 4.4.2: Downloading training data using train_diseases repo.
427. Y_scrape.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.4.3: Building machine learning algorithms in terminal.
428. Train_audioclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling
Figure 4.4.3: Text output of the training session.
429. Schiozphrenia_controls.txt. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.4.2: Performance of various machine learning models and feature embeddings.
430. Train_audioclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling
431. Train_textclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling
432. Train_audiotextclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling
433. Train_w2vclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.4.5: Applying model to predict new classes.

434. Load_audioclassify.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.6.1: Types of regression models: pros/cons.

435. Ordinary least squares. https://en.wikipedia.org/wiki/Ordinary_least_squares
436. Ridge regression. https://en.wikipedia.org/wiki/Tikhonov_regularization
437. Lasso (statistics). https://en.wikipedia.org/wiki/Lasso_(statistics)
438. Multi-task lasso. http://scikit-learn.org/stable/modules/linear_model.html
439. Elastic net. https://en.wikipedia.org/wiki/Elastic_map
440. Multi-task elastic net. http://scikit-learn.org/stable/modules/linear_model.html
441. Least angle regression (LARS). https://en.wikipedia.org/wiki/Least-angle_regression
442. LARS lasso. http://scikit-learn.org/stable/modules/linear_model.html
443. Orthogonal matching pursuit. https://en.wikipedia.org/wiki/Matching_pursuit
444. Bayesian ridge regression. https://en.wikipedia.org/wiki/Bayesian_linear_regression
445. Automatic relevance determination. http://scikit-learn.org/stable/modules/linear_model.html
446. Logistic regression. https://en.wikipedia.org/wiki/Logistic_regression
447. Stochastic gradient descent. https://en.wikipedia.org/wiki/Stochastic_gradient_descent
448. Perceptron. https://en.wikipedia.org/wiki/Perceptron
449. Passive aggressive algorithms. http://scikit-learn.org/stable/modules/linear_model.html
450. Robustness regression. https://en.wikipedia.org/wiki/Robust_regression
451. Random sample consensus. http://scikit-learn.org/stable/modules/linear_model.html
452. Theil-Sen estimator. http://scikit-learn.org/stable/modules/linear_model.html
453. Huber regression. http://scikit-learn.org/stable/modules/linear_model.html
454. Polynomial regression. http://scikit-learn.org/stable/modules/linear_model.html

Figure 4.6.1: Building regression models in python from CLI.

455. Train_audioregression.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.6.2: Building regression models in python from CLI.

456. Train_audioregression.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.7.1: Deep learning terms and definitions.

457. Neural network. https://en.wikipedia.org/wiki/Artificial_neural_network
458. Layers. https://devblogs.nvidia.com/deep-learning-nutshell-core-concepts/
459. Neuron. https://devblogs.nvidia.com/deep-learning-nutshell-core-concepts/
460. Weights. https://datascience.stackexchange.com/questions/19099/what-is-weight-and-bias-in-deep-learning

461. Epoch. https://www.quora.com/What-is-an-epoch-in-deep-learning

462. Batch size. https://stats.stackexchange.com/questions/153531/what-is-batch-size-in-neural-network

463. Regularization. https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/

464. Loss functions. https://medium.com/data-science-group-iitr/loss-functions-and-optimization-algorithms-demystified-bb92daff331c

465. Backpropagation. https://en.wikipedia.org/wiki/Backpropagation

Figure 4.7.1: Building a simple deep learning model in keras (MLP).

466. Keras_mlp.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.7.2: Complete list of neural networks.

467. Becominghuman.ai. https://becominghuman.ai/cheat-sheets-for-ai-neural-networks-machine-learning-deep-learning-big-data-678c51b4b463.

Figure 4.7.3: Training a keras model from audio features.

468. Train_audiokeras.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.7.4: Training a keras model from audio features.

469. Train_audiokeras.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.7.5: Applying keras-based deep learning models on new data.

470. Load_audiokeras.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.8.1: Tuning parameters for the gradient boosting classifier.

471. Sklearn.ensemble. http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

FIgure 4.8.2: How TPOT works.

472. TPOT. https://github.com/EpistasisLab/tpot

Figure 4.8.3: How to train a classification model using TPOT.

473. train_audioTPOT.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.8.4: How to load a TPOT classification model.

474. Load_audiotpot.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.8.5: How to train a regression model using TPOT.

475. Train_audioTPOT.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Figure 4.8.6: How to load a regression model using TPOT.

476. Load_audioTPOT.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_4_modeling

Table 4.8.1: List of AutoML engines and repositories.

477. Devol. https://github.com/joeddav/devol
478. Clarifai. https://clarifai.com/
479. H20.ai. https://www.h2o.ai/
480. Datarobot. https://datarobot-public-api-client.readthedocs-hosted.com/en/v2.11.1/
481. Google cloud ML platform. https://cloud.google.com/ml-engine/docs/tensorflow/technical-overview
482. Microsoft Azure ML platform. https://azure.microsoft.com/en-us/services/machine-learning-studio/

Datasets

478. Common voice dataset. link
479. Google Audioset. link
480. NeuroLex disease dataset. link.

Libraries

481. Pandas. link.
482. Xlsxwriter. link.
483. Pytube. link.
484. SpeechRecognition. link.
485. Librosa. link.
486. PyAudioAnalysis. link.
487. Spacy. link.
488. NLTK. link.
489. Gensim. link.
490. Numpy. link.
491. Scikit-learn. link.
492. Statsmodels. link.
493. Keras. link.
494. Tensorflow. link.
495. TPOT. link.
496. Devol. link.
497. Clarifai. link.

498. H20.ai. link.
499. DataRobot. link.
500. Google Cloud ML engine. link.
501. Microsoft Azure ML. link.

Other references

502. Deep learning book. Book to get started.
503. Udacity class**.** Free class to get started with ML.
504. KDNuggets. Blog dedicated exclusively to data science / machine learning.
505. Machine Learning Yearning. Andrew Ng's new machine learning book to guide you through building ML algorithms.
506. Machine learning - Reddit. Reddit posts about machine learning.
507. Google alert - ML. often good to keep up with what google is doing
508. Machine learning mastery blog. lots of tutorials to get started.
509. Machine learning weekly. hand-curated tutorials 1 /mo.
510. MIT news machine learning. Keep up with what MIT is doing with ML.
511. Cortana machine learning. Keep up with what microsoft is doing with Cortana.
512. Stats and Bots. Medium blog with lots of code examples.
513. Indico. Blog about ML
514. FastML. Machine learning made easy.
515. Machine learning theory. All about machine learning theory.
516. BigML. ML made simple (2 posts/wk).
517. DataDive. Rich blog history on machine learning from researcher at microsoft (visualizations).

Chapter 5: Generation

Figure 5.1.1: How Google Duplex works.

518. Image source: Google AI blog. here.

Table 5.2.1: Types of machine-generated data.

519. Text messaging. https://en.wikipedia.org/wiki/Text_messaging
520. Email. https://en.wikipedia.org/wiki/Email
521. Poetry. https://en.wikipedia.org/wiki/Poetry
522. Summary. https://en.wikipedia.org/wiki/Abstract_(summary)
523. News article. https://en.wikipedia.org/wiki/Article_(publishing)#News_articles
524. Blog post. https://en.wikipedia.org/wiki/Blog
525. Chatbot. https://en.wikipedia.org/wiki/Chatbot

Figure 5.2.1: Generating text messages.

526. Generate_text.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.2: Generating emails.

528. Generate_email.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.3: Generating poems.

529. Generate_poem.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.4: Summarizing Wikipedia articles.

530. Generate_summary.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.5: Generating blog posts.

531. Generate_blogpost.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.6: Generating a chatbot from FAQ training data.

532. Make_chatbot.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.2.7: Engaging with a chatbot via CLI.

533. Make_chatbot.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Table 5.3.1: Types of machine-generated audio data.

534. TTS model. https://en.wikipedia.org/wiki/Speech_synthesis

Table 5.3.1: Generating TTS files locally with python (in terminal).

535. Generate_tts.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.3.2: Manipulating audio (in terminal).

536. Generate_filtered.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.3.3: Remixing audio (in terminal).

537. Generate_remix.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Figure 5.4.1: Voice chatbot as a form of mixed data.

538. Make_vchatbot.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_5_generation

Datasets

539. ENRON dataset (emails)
540. NUS-SMS-corpus (text messages)
541. Blog corpus (blogs)
542. 20 newsgroups (news)

Libraries

543. Textgenrnn.
544. Sumy.
545. Chatterbot.
546. NLTK.
547. Spacy.
548. Librosa.
549. Soundfile.
550. Random.
551. Chatterbot.
552. Pyttsx3.
553. PocketSphinx.
554. Keras.
555. Tensorflow.

Chapter 6: Visualizations

Table 6.1.1: Visualization libraries in python.

556. Matplotlib. https://matplotlib.org/tutorials/introductory/sample_plots.html
557. Seaborn. https://seaborn.pydata.org/
558. Ggplot. http://ggplot.yhathq.com/
559. Bokeh. https://bokeh.pydata.org/en/latest/
560. Pygal. http://pygal.org/en/stable/
561. Plotly dash. https://plot.ly/
562. Geoplotlib. https://github.com/andrea-cuttone/geoplotlib
563. Gleam. https://github.com/dgrtwo/gleam
564. Missingno. https://github.com/ResidentMario/missingno
565. Leather. https://leather.readthedocs.io/en/0.3.3/

Figure 6.2.1: Visualizing streaming audio data in python.

566. Audio_stream.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.2.1: Visualizing streaming audio data in python.

567. Audio_path.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.2.2: Making spectrograms with librosa.

568. Audio_spectrogram.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.2.3: Plotting many audio files by spectrum and oscillograms.

569. Audio_plotmany.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.2.4: Plotting many audio files by spectral power density.

570. Audio_plotspd.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.2.4: Plotting many audio files as KNN clusters.

571. Audio_cluster.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

FIgure 6.3.1: Plotting words in real-time.

572. Text_stream.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.2: Word stream visualization.

573. Text_path.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.3: Word frequency plots.

574. Text_freqplot.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.4: Wordcloud plots.

575. Text_wordcloud.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.5: Parsed tree plots.

576. Text_tree.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.6: Named entity visualization.

577. Text_entity.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.7: Word network plots.

578. Networkx. https://networkx.github.io/
579. Text_network.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.8: tSNE plots.

580. Text_tsne.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.3.9: Plotting many tNSE embeddings.

581. Text_tsne_many.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.4.1: Mixed features CLI plot.

582. Mixed_stream.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.4.2: Mixed features path plot.

583. Mixed_path.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.4.3: Making videos of plots using opencv.

584. Mixed_video.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.5.1: Visualizing streaming meta features.

585. Meta_stream.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.5.2: Visualizing multiple streaming meta features.

586. Meta_multi.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Figure 6.5.2: Visualizing non-streaming meta features.

587. Meta_nonstream.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_6_visualization

Libraries

588. Matplotlib
589. Seaborn
590. Ggplot
591. Bokeh
592. Pygal
593. Plotly Dash
594. Geoplotlib
595. Gleam
596. Missingno
597. Leather
598. Librosa
599. Soundfile
600. Sounddevice
601. NLTK
602. Spacy
603. NetworkX
604. Sounddevice
605. OpenCV
606. Keras
607. Scikit-learn

Other references

608. Kaggle. https://www.kaggle.com/vinayshanbhag/visualizing-audio-data

609. Medium. https://medium.com/@LeonFedden/comparative-audio-analysis-with-wavenet-mfccs-umap-t-sne-and-pca-cb8237bfce2f

610. Galvanize. https://blog.galvanize.com/data-science-projects-classifying-and-visualizing-musical-pitch/

Chapter 7: Designing voice computers

Table 7.1.1: Types of assembled voice computing hardware.

611. Desktop computer. https://en.wikipedia.org/wiki/Desktop_computer
612. iMac. https://www.apple.com/imac/
613. Laptop. https://en.wikipedia.org/wiki/Laptop
614. Macbook pro. https://www.apple.com/macbook-pro/
615. Raspberry pi. https://en.wikipedia.org/wiki/Raspberry_Pi
616. Jasper. http://jasperproject.github.io/documentation/
617. Mycroft. https://mycroft.ai/
618. Ardunio. https://gist.github.com/xmfcx/3e6be68dd655cd7058085702e0bcb4b0
619. Cell phone. https://en.wikipedia.org/wiki/Mobile_phone
620. iPhone. https://www.apple.com/ios/siri/
621. Amazon Echo. https://en.wikipedia.org/wiki/Amazon_Echo
622. Google assistant. https://developers.google.com/actions/
623. Apple HomePod. https://www.apple.com/homepod/

Table 7.1.2: Types of voice computing software.

624. Alexa voice service. https://developer.amazon.com/alexa-voice-service
625. Google Assistant SDK. https://developers.google.com/assistant/sdk/overview
626. Cortana skills kit. https://developer.microsoft.com/en-us/cortana
627. Jovo software framework. https://github.com/jovotech/jovo-framework-nodejs
628. Jasper. http://jasperproject.github.io/documentation/
629. Mycroft. https://mycroft.ai/
630. Nala. https://github.com/jim-schwoebel/nala

Table 7.2.1: Hardware considerations for voice computers.

631. Unit cost. https://en.wikipedia.org/wiki/Unit_cost
632. User. https://en.wikipedia.org/wiki/User_(system)
633. Central processing unit. https://en.wikipedia.org/wiki/Central_processing_unit
634. Microphone distance.https://en.wikipedia.org/wiki/Microphone
635. Wifi. https://en.wikipedia.org/wiki/Wi-Fi
636. Electric power. https://en.wikipedia.org/wiki/Electric_power
637. Cloud storage. https://en.wikipedia.org/wiki/Cloud_storage
638. Bluetooth protocol. https://en.wikipedia.org/wiki/List_of_Bluetooth_protocols

Figure 7.2.1: Cost-driven hardware design flow diagram.

639. Motherboard. https://en.wikipedia.org/wiki/Motherboard
640. Form factor. https://en.wikipedia.org/wiki/Form_factor_(design)

641. Arduino. https://en.wikipedia.org/wiki/Arduino
642. Raspberry pi. https://en.wikipedia.org/wiki/Raspberry_Pi
643. Microphone. https://en.wikipedia.org/wiki/Microphone
644. Sound card. https://en.wikipedia.org/wiki/Sound_card
645. CPU. https://en.wikipedia.org/wiki/Central_processing_unit
646. Graphics card. https://en.wikipedia.org/wiki/Video_card
647. Hard disk. https://en.wikipedia.org/wiki/Hard_disk_drive
648. Monitor. https://en.wikipedia.org/wiki/Computer_monitor
649. Wifi chip. https://en.wikipedia.org/wiki/Wi-Fi
650. Bluetooth. https://en.wikipedia.org/wiki/Bluetooth
651. Loudspeaker. https://en.wikipedia.org/wiki/Loudspeaker
652. Power supply. https://en.wikipedia.org/wiki/Power_supply
653. Computer case. https://en.wikipedia.org/wiki/Computer_case

Table 7.2.1.1: List of microphones, their utility, cost

654. Condensor microphone (cost). Amazon.
655. Dynamic microphone (cost). Amazon.
656. Ribbon microphone (cost). Amazon.
657. MEMS microphone (cost). Amazon.
658. Electret microphone (cost). Amazon.
659. Noise-cancelling microphone (cost). Amazon.

Table 7.2.1.2: Types of microphone arrays.

660. UMA-8 Microphone Array. https://www.minidsp.com/products/usb-audio-interface/uma-8-microphone-array
661. Respeaker. Amazon.
662. Element 14. https://www.element14.com/community/docs/DOC-86328
663. Blue Yeti microphone. Amazon.

Table 7.2.2.1: List of [Raspberry Pi]-upgradeable sound cards

664. HiFiBerry DAC+ Standard. https://www.hifiberry.com/shop/boards/hifiberry-dacplus-rca-version/
665. IQaudIO PiDAC+. http://iqaudio.co.uk/hats/9-pi-digiamp-0712411999650.html
666. JustBoom DAC Hat. https://www.amazon.com/Audio-Amplifier-Raspberry-Class-D-192KHz/dp/B01M32PGX4/ref=sr_1_1?ie=UTF8&qid=1532963601&sr=8-1&keywords=justboom+dac+hat

Table 7.2.2.2: List of PC-enabled sound cards and their costs.

667. Asus Xonar DSX PCIe 7.1 (cost and image). Amazon.
668. Asus Xonar GHX PCIe GX2.5 (cost and image). Amazon
669. Creative Sound Blaster Audigy FX (cost and image). Amazon.
670. Creative Sound Blaster ZxR PCIe (cost and image). Amazon.
671. Creative Sound Blaster Z (cost and image). Amazon.
672. Creative Sound Blaster Audigy PCIe RX (cost and image). Amazon
673. Asus Essence STX II (cost and image). Amazon

Figure 7.2.3.1: Different form factors in motherboard designs.

674. Form factor (image). https://en.wikipedia.org/wiki/Form_factor_(design)

Table 7.2.3.1: List of desktop motherboards and their costs (2018).

675. 5x5 motherboard by intel.
676. ASUS Q87T/CSM LGA 1150 Intel Q87 HDMI SATA 6Gb/s USB 3.0 Thin Mini-ITX Intel Motherboard For AiO And Ultra Slim System.
677. Gigabyte GA-H110M-A LGA1151 Intel H110 Micro ATX DDR4 Motherboard
678. MSI Pro Series Intel B250 LGA 1151 DDR4 HDMI USB 3.1 ATX Motherboard (B250 PC MATE).
679. MSI ProSeries AMD Ryzen B350 DDR4 VR Ready HDMI USB 3 micro-ATX Motherboard (B350M PRO-VDH).

Table 7.2.4.2: List of desktop CPUs and their costs.

680. AMD Turion X2 Ultra ZM-84 TMZM84DAM23GG Mobile CPU Processor Socket S1G2 638pin 2.3GHz 2MB (cost and image). Amazon.
681. Intel E6550 2.33 dual core 4mb 1333 mhz cpu - SLA9X (cost and image). Amazon.
682. XCSOURCE ESP32S Development Board (cost and image). Amazon.
683. Intel Celeron D 336 2.8GHz CPU (cost and image). Amazon.
684. AMD FX-4300 Quad-Core Vishera Processor 3.8GHz Socket AM3+, Retail FD4300WMHKBOX (cost and image). Amazon.
685. Intel Computer CPU 1.7 8 BX80660E52609V4 (cost and image). Amazon.

Figure 7.2.5.1: List of RAM memory and prices.

686. A-tech micron 4GB RAM
687. Corsair 8GB RAM
688. Corsair 16GB RAM
689. Corsair 32GB RAM

Table 7.2.6.1: List of storage mediums and costs.

690. USB flash drive. Amazon.
691. SD card. Amazon.
692. Hard disk. Amazon.
693. Cloud provider. AWS pricing.

Table 7.2.7.1: List of computer monitors and costs.

694. LED monitors (cost). Amazon.
695. LED monitor (image). Amazon.
696. Touchscreen monitors (cost). Amazon.
697. Touchscreen monitor (image). Amazon.
698. Projector (cost). Amazon.
699. Projector (image). Amazon.

Table 7.2.8.1: Types of radiofrequency transceivers and costs.

700. Hack RF-one SDR.

701. Ubertooth One.
702. Yarstick One USB transceiver.
703. Seeedstudio Kiwi SDR.

Figure 7.2.9.1: Types of IoT-enabled devices that can connect to voice computers.

704. Figure taken from GE's connect WiFi kitchen webset. http://www.geappliances.com/ge/connected-appliances/

Table 7.2.9.1: List of WiFi network adapters and costs.

705. Plugable USB 2.0 Wireless N 802.11n 150 Mbps Nano WiFi Network Adapter.
706. TP-Link TL-WN881ND N300 PCI-E Wireless WiFi network Adapter card for pc.
707. Feb Smart Wireless Dual Band N600 (2.4GHz 300Mbps or 5GHz 300Mbps) PCI Express (PCIe) Wi-Fi Adapter Network Card.
708. ASUS Dual-Band Wireless-AC1900 PCI-E Adapter (PCE-AC68).

Table 7.2.10: List of bluetooth transceivers and costs.

709. Whitelabel Bluetooth 4.0 USB Dongle Adapter.
710. Gigabyte GC-WB867D-I REV Bluetooth 4.2/Wireless AC/B/G/N Band Dual Frequency 2.4Ghz/5.8Ghz Expansion Card.
711. Intel Dual Band Wireless-AC 7260 2x2 Network plus Bluetooth adapter (7260.HMWWB.R).

Table 7.2.11.1: Speaker types and their utility.

712. Headphones. https://en.wikipedia.org/wiki/Headphones
713. Dynamic speaker. https://en.wikipedia.org/wiki/Loudspeaker
714. Piezoelectric speaker. https://en.wikipedia.org/wiki/Piezoelectric_speaker
715. Electrostatic loudspeaker. https://en.wikipedia.org/wiki/Electrostatic_loudspeaker
716. Loudspeaker. https://en.wikipedia.org/wiki/Loudspeaker
717. Ribbon loudspeaker. https://en.wikipedia.org/wiki/Loudspeaker

Table 7.2.11.2: List of speaker types and costs.

718. ARVICKA Blue LED USB Speakers- Wired Laptop Speakers 2.0 Channel.
719. DOSS Touch Wireless Bluetooth V4.0 Portable Speaker with HD Sound and Bass (Black).
720. Anker Soundcore Bluetooth Speaker.
721. ION Audio Tailgater (iPA77) | Portable Bluetooth PA Speaker with Mic, AM/FM Radio, and USB Charge Port.

Table 7.2.12.1: List of power supply / battery types and costs (2018).

722. Kuman Lithium Battery Pack Expansion Board Power Supply.
723. energyShield 2 Pro - Arduino Battery, Solar Powered, Eco Mode, Real Time Clock, and Fuel Gauge.
724. Rosewill Gaming Power Supply, Arc Series 750 Watt (750W).
725. NEXGADGET 42000mAh power

Table 7.2.13: List of housing types and costs.

726. Custom raspberry pi housing.
727. Custom arduino housing.
728. DIYPC Skyline-06-WG.
729. Thermaltake View 71 RGB.

Figure 7.2.13: Outer housing of the Apple Homepod in terms of its interior parts.

730. Apple HomePod. https://www.apple.com/homepod/

Figure 7.3.1.1: Using PyBluez for bluetooth data transmission.

731. PyBluez. https://github.com/pybluez/pybluez
732. Bluetooth.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.2.1: Using Wireless module to connect to a WiFi network.

733. Wireless. https://pypi.org/project/wireless/
734. Wifi.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.3.1: Connecting to Arduino devices through COM ports.

735. Pyserial. https://pythonhosted.org/pyserial/
736. Pyserial.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.1: Using transcription tools for wake words.

737. Wake_transcribe.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.2: Performance of various wakeword engines (python).

738. Wake word benchmarks. https://github.com/Picovoice/wakeword-benchmark

Figure 7.3.4.3: Hotword detection with PocketSphinx.

739. Wake_pocket.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.4: Hotword detection with Snowboy.

740. Wake_snowboy.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.5: Training keywords with porcupine from CLI.

741. Wake_porcupine.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.6: Hotword detection with Porcupine.

742. Wake_porcupine.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.5.1: Building a keyword corpus for training a PocketSphinx language model.

743. ./data/corpus.txt. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.3.4.2: Getting model files from the sphinx knowledge base generator.

744. LMTool. http://www.speech.cs.cmu.edu/tools/lmtool-new.html

Figure 7.3.5.3: Implementing a custom transcription model.

745. Transcribe_custom.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.1: Meet Nala - the open source voice assistant.

746. Nala. https://github.com/jim-schwoebel/nala

Figure 7.4.1.1: Setting up environment variables.

747. Medium. https://medium.com/@himanshuagarwal1395/setting-up-environment-variables-in-macos-sierra-f5978369b255

Figure 7.4.2.1: Recording user queries.

748. Record.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.3.1: Function to transcribe audio.

749. Transcribe.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.4.1: Specifying wake word engines.

750. Wakeup.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.5.1: Customizing text-to-speech engine.

751. ./nala/Nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.6.1: Registration process.

752. ./nala/nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Table 7.4.2.2: Modifiable settings.

753. ./nala/settings.json (after registering). https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.7.1: Loading databases.

754. ./nala/nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.8.1: Visualizing the intent loop.

755. Created by author to describe how Nala understands user queries.

Figure 7.4.8.2: Coding the intent loop.
756. ./nala/nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.8.3: Closing the intent loop.
757. ./nala/nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.9.1: Creating an action.
758. ./nala/actions/makajoke.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.9.2: Initializing import statements and helper functions for actions.
759. ./nala/actions/makajoke.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.9.3: using sys.argv[1] to pass through directory information.
760. ./nala/actions/makajoke.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.9.4: How to update the database while coding actions.
761. ./nala/actions/makajoke.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Table 7.4.8.1: Action-intent pairs for Nala.
762. ./nala/actions (folder) and ./nala/nala.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_7_design

Figure 7.4.10: Piecing it together.
763. Nala. https://github.com/jim-schwoebel/nala

Hardware Platforms
764. iMac
765. Macbook pro
766. Jasper
767. Mycroft
768. Arduino
769. iPhone / Siri
770. Amazon Echo
771. Google Assistant
772. Apple HomePod

Software platforms
773. Alexa voice service (AVS)

774. Google assistant SDK
775. Jovo software framework
776. Jasper voice bot software
777. Mycroft AI
778. Nala

Python libraries

779. Pybluez
780. Wireless
781. Wifi
782. Pyserial
783. PyVISA
784. PocketSphinx
785. Snowbird
786. Porcupine
787. PocketSphinx Language models
788. PyKaldi
789. LIUM diarization models
790. Sys
791. Os
792. Pyttsx3

Other resources

793. Voice recognition project with Arduino. http://www.instructables.com/id/Speech-Recognition-with-Arduino/
794. Jasper python library. https://maker.pro/raspberry-pi/tutorial/the-best-voice-recognition-software-for-raspberry-pi
795. Google assistant - tutorial. https://developers.google.com/actions/
796. Sonos. https://musicpartners.sonos.com/?q=node/442
797. USA today. https://www.usatoday.com/story/tech/news/2018/07/07/a-foolish-take-many-americans-are-talking-to-their-devices/36563399/

Chapter 8: Server architectures

Figure 8.1.1: The client-server model.

798. Mozilla Firefox (edited image). https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction

Figure 8.1.2: Web browsers that can play back audio media via HTML5 tag.

799. HTML5 audio (edited image). https://en.wikipedia.org/wiki/HTML5_audio

Figure 8.1.3: How WebRTC architectures work within the client-server model.

800. WebRTC. https://webrtc.org/
801. HTML5rocks. https://www.html5rocks.com/en/tutorials/webrtc/basics/

Table 8.1.1: Types of servers.
802. Tower server.
803. Rack server.
804. Blade server.

Table 8.1.2: Types of server software.
805. Apache web server.
806. Nginx web server (Engine-X).
807. LiteSpeed web server.
808. Lighttpd.

Table 8.1.3: Common database types and their pros/cons.
809. MongoDB.
810. Postgres.
811. MySQL.
812. JSON.

Figure 8.1.4: Load balancing server traffic.
813. Load balancing. https://en.wikipedia.org/wiki/Load_balancing_(computing)

Table 8.1.3: Types of servers.
814. Servers (table copied from here). https://en.wikipedia.org/wiki/Server_(computing)

Figure 8.1.3: Monolithic vs. microservice architectures.
815. Martin Flower’s blog (image credit). Link.

Figure 8.1.4: Comparing virtual machines (VMs) and container environments.
816. Docker. https://www.docker.com/what-container
817. VMWare virtual machines.
https://www.vmware.com/support/ws5/doc/ws_newguest_setup_simple_steps.html

Figure 8.1.5: Using Kubernetes as a container orchestration platform.
818. Redhat (images). link.

Figure 8.1.6: Using Apache Kafka for data pipelines.
819. Scotch.io (image). link.

Figure 8.2.1: Spinning up a web server with Flask framework.
820. Flask_project.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.2.2: Spinning up a web server with Django framework.
821. Django_server.py.
https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

FIgure 8.2.3: How NGINX server work with python.

822. Patrick Software Blog. link.

Figure 8.2.4: Installing and configuring a NGINX-WSGI server with Flask.

823. Homebrew. https://brew.sh/

Figure 8.2.5: Using Postman for testing GET/POST requests.

824. flask_project.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.3.1: Installing mongoDB from the terminal and launching it.

825. Installing mongodb. https://docs.mongodb.com/manual/installation/

Table 8.3.1: Common mongoDB commands in python executed with the pymongo module.

826. Pymongo. https://api.mongodb.com/python/current/
827. Mongo_commands.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.3.2: Overcoming security message to install Robo3T on mac operating systems.

828. Screenshots from the installation process. https://robomongo.org/

Figure 8.3.3: Setting up a new mongoDB database connection locally using Robo3T.

829. Screenshots from the setup process. https://robomongo.org/

Table 8.4.1: Some things that you can do with Kafka.

830. Introduction to Apache Kafka. https://www.confluent.io/blog/introduction-to-apache-kafka-for-python-programmers/
831. TechBeacon. https://techbeacon.com/what-apache-kafka-why-it-so-popular-should-you-use-it

Figure 8.4.1: How to launch a kafka cluster from the command line (port 9092).

832. Kafka quickstart. https://kafka.apache.org/quickstart

Figure 8.4.2: Making a kafka producer.

833. Kafka_producer.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.4.3: Making a kafka consumer.

834. Kakfa_consumer.py https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Table 8.5.1: Variables needed to setup a minio client.

835. Minio client SDK for python. https://github.com/minio/minio-py

Figure 8.5.1: Spinning up a minio client in python as an alternative to S3.

836. Minio client SDK for python. https://github.com/minio/minio-py

837. Minio.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.6.1: Authentication with the OAuth2 protocol.

838. Digital Ocean (image). https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2

Figure 8.6.2: How to use Auth0 for user authentication.

839. Auth0 python client. https://github.com/auth0/auth0-python
840. Auth0.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Table 8.7.1: Common docker terms and definitions.

841. Docker commands and best practices cheat sheet. https://zeroturnaround.com/rebellabs/docker-commands-and-best-practices-cheat-sheet/

Table 8.7.2: Useful docker commands.

842. Docker Hub. https://docs.docker.com/docker-hub/
843. Using docker with python (YouTube video). https://www.youtube.com/watch?v=VhabrYF1nms

Table 8.8.1: Common software testing techniques.

844. Software testing. https://en.wikipedia.org/wiki/Software_testing
845. Types of automated tests. https://www.atlassian.com/continuous-delivery/different-types-of-software-testing

Table 8.8.2: Useful unittest() comparison functions.

846. Unittest. https://docs.python.org/2/library/unittest.html

FIgure 8.8.1: Using the unittest module.

847. Unittest. https://docs.python.org/2/library/unittest.html
848. Unittest.py. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Figure 8.8.2: Running automated tests in a docker container.

849. ./docker/Dockerfile. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Table 8.9.1: Common GitHub terms and definitions.

850. Assembled from author’s intuition and memory.

Table 8.9.2: Common GitHub CLI commands.

851. GitHub cheat sheet. https://education.github.com/git-cheat-sheet-education.pdf

Figure 8.9.1: Setting up the Heroku CLI.

852. Heroku. https://devcenter.heroku.com/articles/heroku-cli

Table 8.9.3: Common Heroku CLI commands.

853. Heroku command line. https://devcenter.heroku.com/categories/command-line

Figure 8.9.2: Deploying code through Heroku.

854. Deployment. https://devcenter.heroku.com/categories/deployment

Table 8.10.1: Common kubernetes terms and definitions.

855. Glossary of terms. https://kubernetes.io/docs/reference/glossary/?fundamental=true

Table 8.10.2: Getting used to the kubectl CLI.

856. Kubectl installation. https://kubernetes.io/docs/tasks/tools/install-kubectl/

Figure 8.10.1: Combining docker with kubernetes.

857. Techbeacon (image). https://techbeacon.com/one-year-using-kubernetes-production-lessons-learned

Figure 8.10.2: Sample .yml file to spin up NGINX server.

858. ./nginx.yml. https://github.com/jim-schwoebel/voicebook/tree/master/chapter_8_server

Python libraries

859. Unittest() framework
860. Pytest
861. Auth0
862. Minio
863. Kafka-python
864. Json
865. Sqlite3
866. Pyycopg
867. Pymongo
868. Flask-pymongo
869. Jinja
870. ReST API / Flask
871. Gunicorn
872. Uwsgi
873. Virtualenv
874. Flask
875. Django

Software tools

876. VMware workstation pro
877. Docker

878. Heroku
879. Kubernetes
880. GCP Kubernetes Engine documentation
881. Github cheat sheet
882. AWS S3
883. Robo 3T
884. Postman

Other references

885. RTP.
886. Client-server model
887. Microservices vs. monoliths
888. Basics of web architecture
889. Web servers.
890. MongoDB
891. Postgres
892. MySQL
893. JSON
894. HTML5 audio tag
895. HTML5 video tag
896. WebRTC API
897. Media Capture Streams API
898. Mediastreams Recording API
899. Nginx web server
900. Difference between containers and virtual machines
901. VMware workstation pro
902. Docker
903. Kubernetes

Chapter 9: Security, legal, and ethical considerations

Figure 9.1.1: The voice security framework (VSF).

904. Created by author.

Figure 9.1.1.1: List of cloud providers by market share.

905. Parkmycloud (image). https://www.parkmycloud.com/blog/aws-vs-azure-vs-google-cloud-market-share/

Table 9.1.1.1: Types of ciphers for encrypting/decrypting data.

906. Pycryptodome. https://pycryptodome.readthedocs.io/en/latest/

Figure 9.1.1.2: How to generate a secure public key.

907. Generate_password.py. https://github.com/jim-schwoebel/voicebook/blob/master/chapter_9_security/generate_

password.py

Figure 9.1.1.3: How to encrypt an audio file.

908. Encrypt_file.py. https://github.com/jim-schwoebel/voicebook/blob/master/chapter_9_security/generate_password.py

Figure 9.1.1.4: How to decrypt an audio file.

909. Decrypt_file.py. https://github.com/jim-schwoebel/voicebook/blob/master/chapter_9_security/generate_password.py

Table 9.1.5.1: Security certifications.

910. HIPAA Certification
911. HITRUST Certification
912. PCI DSS

Table 9.1.6.1: Voice security vulnerabilities.

913. Voice squatting.
914. Voice Masquerading.
915. Voice Adversarial Attack.
916. Weak voice authentication / physical security.
917. DophinAttack.

Table 9.1.7.1: List of blockchain platform relevant to voice computing.

918. Voise.
919. Video Coin.
920. CellTrust blockchain.
921. Langnet.
922. Snips AIR.
923. Flask blockchain tutorial.

Figure 9.2.1: The voice legal framework (VLF).

924. Figure made by author.

Table 9.2.1.1: Important laws and their descriptions.

925. Wiretap Act of 1968.
926. California Connected Televisions Statute.
927. One / two party consent laws.
928. Child online privacy protection rule.
929. Computer Fraud and Abuse Act.
930. General Data Protection Regulation.
931. Health Insurance Portability and Accountability Act.

Table 9.2.2.1: Recent court cases and laws affecting consumer privacy.

932. Personal Data Notification and Protection Act.
933. United States vs. Carpenter.

Table 9.2.3.1: Rights you have for sound recordings under the US Copyright Act of 1976.

934. Copyright law of the United States. https://en.wikipedia.org/wiki/Copyright_law_of_the_United_States

Figure 9.3.1: The voice ethics framework (VEF).

935. Figure made by author.

Table 9.3.1: Risks voice computers pose to under users.

936. Table made by author from intuition.

Table 9.3.2: Actions the larger voice community can take to reduce consumer risks.

937. Table made by author from intuition.
938. Common voice project. https://voice.mozilla.org/en
939. Audioset. https://research.google.com/audioset/
940. American with disabilities act of 1990. https://en.wikipedia.org/wiki/Americans_with_Disabilities_Act_of_1990
941. Federated machine learning models. https://ai.googleblog.com/2017/04/federated-learning-collaborative.html
942. Copyright law of the United States. https://en.wikipedia.org/wiki/Copyright_law_of_the_United_States
943. Patents. https://en.wikipedia.org/wiki/Patent.

Chapter 10: Getting involved

Figure 10.1.1: YouTube Video of NeuroLex's story.

944. Voicecamp demo day. https://www.youtube.com/watch?v=oiCY3dcAbX0

Table 10.1.1: Summary of voice biomarker studies.

945. García, Adolfo M., et al. "How language flows when movements don't: An automated analysis of spontaneous discourse in Parkinson's disease." Brain and Language 162 (2016): 19-28. http://www.sciencedirect.com/science/article/pii/S0093934X16300256
946. Khodabakhsh, Ali, et al. "Evaluation of linguistic and prosodic features for detection of Alzheimer's disease in Turkish conversational speech." EURASIP Journal on Audio, Speech, and Music Processing 2015.1 (2015): 9. https://asmp-eurasipjournals.springeropen.com/articles/10.1186/s13636-015-0052
947. König, Alexandra, et al. "Automatic speech analysis for the assessment of patients with predementia and Alzheimer's disease." Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring 1.1 (2015): 112-124. http://www.sciencedirect.com/science/article/pii/S2352872915000160
948. Pestian, John P., et al. "A machine learning approach to identifying the thought markers of suicidal subjects: a prospective multicenter trial." Suicide and

life-threatening behavior (2016). http://onlinelibrary.wiley.com/doi/10.1111/sltb.12312/full

949. Bedi, Gillinder, et al. "Automated analysis of free speech predicts psychosis onset in high-risk youths." npj Schizophrenia 1 (2015): 15030. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4849456/

Figure 10.2.1: The NeuroLex Innovation Fellows program timeline.

950. Application link. http://innovate.neurolex.co

Table 10.3.1: List of open source voice datasets.

951. Common Voice Dataset.
952. Google Audioset.
953. Urban Sound dataset.
954. NeuroLex Disease Dataset.
955. NeuroLex Emotion Dataset.
956. Mivia audio events dataset.
957. Karoldvl.

Table 10.3.2: List of some open source voice projects.

958. Librosa
959. Pyaudioanalysis
960. Soundfile
961. Sounddevice
962. SpeechRecognition
963. PocketSphinx
964. SIDEKIT

Table 10.4.1: Voice computing conferences (2018).

965. Smart Voice Summit
966. Conversational Interaction Conference
967. Bottish
968. Superbot
969. SpeechTEK
970. Voice & Beyond
971. Conversational Commerce Conference
972. Business of Bots
973. Voicecon
974. Connections: Connected Home Conference
975. Voice Summit
976. ML Conf
977. The Voice of Healthcare Summit
978. Conversational Commerce Conference
979. All About Voice
980. AWS re:INVENT
981. Voice Conference

982. CES
983. The Alexa Conference
984. Lingofest

Table 10.4.1: List of voice-related research conferences.
985. List abridged. http://www.cstr.ed.ac.uk/research/conferences/sby-cdate.html

Figure 10.6.1: The voice tech landscape.
986. Voice tech landscape. https://medium.com/point-nine-news/voice-tech-landscape-150-startups-mapped-and-analysed-82c5adaf710
987. Amazon Alexa.
988. Facebook.
989. Google Assistant.
990. Microsoft Cortana.
991. IBM Watson.
992. Sonos.
993. Bose.
994. Twilio.

Table 10.7.1: List of accelerators to launch your voice startup.
995. Y-Combinator. https://www.ycombinator.com/rfs/#voice
996. Techstars. https://www.techstars.com/
997. Betaworks. https://betaworks.com/voicecamp/
998. Alexa accelerator. https://www.techstars.com/programs/alexa-program/

NeuroLex resources
999. NeuroLex's story (video).
1000. The Innovation Fellows Program.
1001. TRIBE FAQ document.
1002. TRIBE 2 Demo Day.
1003. Train diseases repository.

Other references
999. Stanford University.
1000. Carnegie Mellon University (Language technologies institute).
1001. Johns Hopkins University (Center for Speech Language Technology).
1002. Georgia Tech.
1003. University of Pennsylvania (Department of Linguistics).
1004. Massachusetts Institute of Technology (EECS).
1005. University of Washington.
1006. UC Berkeley (Computational Linguistics).