Audio Segmenting and Natural Language Processing in Oral History Archiving

by

Holly Anne Rieping

A.S. Computer Science
Pikes Peak Community College, 2020
B.S. Computer Science and Engineering
Massachusetts Institute of Technology, 2021

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2022

Author .....
Department of Electrical Engineering and Computer Science
January 21, 2022

Certified by .....
Kurt E. Fendt
Senior Lecturer in Comparative Media Studies/Writing
Thesis Supervisor

Accepted by .....
Katrina LaCurts
Chair, Master of Engineering Thesis Committee

Audio Segmenting and Natural Language Processing in Oral History Archiving

by

Holly Anne Rieping

Submitted to the Department of Electrical Engineering and Computer Science
on January 21, 2022, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Traditional archives preserve physical historical records, documents, artifacts, etc. and tell a story of some historical significance. As the digital age progresses, digital archives have become more commonplace and have given wider access to archival resources and knowledge to the general public. With wider access, historically marginalized groups now have the means to share stories that have typically been excluded from the dominant discourse. As a result, we are faced with both the challenge and the opportunity to tell and preserve stories from these groups and foreground diverse voices in these digital archives. Additionally, we are faced with the challenge of having an abundance of materials, both digitized and born digital, to use in an archive, and can utilize various computational methods to assist in the curatorial process of a digital archive by organizing the materials or finding connections between different materials that would otherwise take hundreds of hours for an archivist to do.

Using materials from the MIT Black Oral History Project, this thesis first explores ways to process digitized audio interviews through audio segmentation, using techniques including silence detection and speaker diarization, with the goal of creating a more flexible way to explore interviews in a digital oral history archive. Second, this thesis uses named entity recognition to experiment with metadata extraction for an archive. Next, this thesis explores ways to discover connections between segments of interviews by using topic modeling with LDA and LSI and topic classification using machine learning methods to identify topics, similarities, and dissimilarities across interviews. Finally, this thesis discusses how these computational methods may enhance the telling of diverse stories in digital oral history archives.

Thesis Supervisor: Kurt E. Fendt
Title: Senior Lecturer in Comparative Media Studies/Writing

Acknowledgments

I would first like to acknowledge my family for all their love and support throughout my time at MIT. I wouldn't be here without it!

I'd like to acknowledge my Mom and Dad for helping me create data sets used in this thesis, introducing me to computer science, teaching me to explain technical things in a non-technical way, and giving me the support necessary for me to attend MIT.

I'd like to acknowledge my Grandma and Grandpa and Grammie and Grandaddy for always encouraging me in my education.

I'd like to acknowledge my brother Harrison for tutoring me in my first coding class.

I'd like to acknowledge my brother Holden for always reminding me to be proud of myself.

I'd like to acknowledge Willem for helping me create data sets used in this thesis and for love and support the past three years.

I'd like to acknowledge Kurt for being the most supportive, encouraging, and caring thesis supervisor I could have asked for. Thank you for giving me the opportunity to work on this thesis and all your guidance throughout the process!

I'd like to acknowledge Debbie Douglas and the MIT Museum for providing access to the MIT Black Oral History Project materials and feedback throughout the early development stages.

I'd like to acknowledge Stefanie Mueller for being a supportive advisor throughout my time at MIT and advocating for me in my hardest semesters.

I'd like to acknowledge Kimberlie Koile for supporting me as a 6.034 TA and giving me the once in a lifetime chance to teach at MIT.

Finally, I'd like to acknowledge Mr. Hoit for teaching me to love calculus (calc is life!), helping me get into MIT, and helping me realize I belonged at MIT.

Contents

List of Figures **9**

List of Tables **11**

1 Introduction **13**

2 Background and Related Work **17**

2.1 The MIT Black Oral History Project 17

2.2 Storytelling in Archives 18

2.3 Audio Segmentation 18

2.4 Speech to Text 20

2.5 Metadata and Natural Language Processing 20

2.6 Topic Modeling and Coherence 22

2.7 Topic Classification 24

3 Project Development **27**

3.1 Goals of the Future Archive 27

3.2 Interview Selection 28

3.3 Audio Segmentation Approaches 29

3.3.1 Split by Silence 29

3.3.2 Voice Classification Model 31

3.3.3 Speaker Diarization 33

3.4 Speech to Text Approaches 35

3.5 Named Entity Recognition and Text Vectorization 36

3.6 Topic Modeling and Classification Approaches 39
3.6.1 Topic Modeling using LDA and LSI 39
3.6.2 Topic Data Collection 41
3.6.3 Topic Classification Approaches 44

4 Results **49**
4.1 Voice Classification & Diarization Performance 49
4.2 Metadata Extraction 53
4.3 Topic Classification Accuracy and Understandability 55

5 Conclusion **59**
5.1 Further Development 59
5.2 Takeaways 61

A Transcripts **63**

B Software Libraries and Packages **69**
B.1 Audio Segmentation 69
B.2 Speech to Text 69
B.3 Machine Learning 70
B.4 Natural Language Processing 70
B.5 Topic Modeling and Classification 70

Bibliography **73**

List of Figures

3-1 Silence Detection 30

3-2 Text Vectorization Examples 39

3-3 Topic Frequency in Full Corpus (all segments) 43

3-4 Topic Frequency in Transcripts with Duplicates vs. No Duplicates . . 46

3-5 Topic Frequency in Corpus with Duplicates Removed 47

4-1 Voice Classification Model Accuracy 50

4-2 Difference of Average Loudness (dB) between Interviewee and Dr. Williams 51

4-3 spaCy vs. Otter.ai: Dr. Williams' Question 54

4-4 spaCy vs. Otter.ai: Kofi Annan's Answer 54

List of Tables

3.1 Ten Selected Interviews 28
3.2 Named Entity Recognition of Dr. Williams' Question 37
3.3 Named Entity Recognition of Kofi Annan's Answer 37
3.4 LDA Topic Modelling Probability Equations 40
3.5 LSI Topic Modelling Probability Equations 42
3.6 List of Topics Defined by Me 43

4.1 Accuracy of Voice Classification Models for Ten Interviews 50
4.2 Segmentation Comparison of Kofi Annan's Interview (Tape 1.1) . . . 52
4.3 Segmentation Comparison of Gregory Chisholm's Interview (Tape 2.1) 52
4.4 Topic Classification Model Accuracies 56
4.5 Topic Classification Model Predictions for Transcripts A.4 and A.5 . . 57

Chapter 1

Introduction

Burdick et al. define the Digital Humanities as, "...new modes of scholarship and institutional units for collaborative, trans-disciplinary, and computationally engaged research, teaching, and publication" [6]. Through this field, we can utilize computational skills to engage with the humanities from a different perspective and find new understandings from viewing the two fields as a unified one. As we progress in the digital era, the questions of whose stories are told, how those stories are told, and how those stories are remembered become even more prevalent. This thesis will be rooted in these questions and explore their place in alternative storytelling and foregrounding diverse voices in active, digital archives.

An archive is traditionally a physical location that preserves documents and artifacts of historical significance, like the Library of Congress. This is also an example of an institutional archive, an archive funded and maintained by an institution with professional curators. Typically, the content in an institutional archive stays within the status quo of the dominant history, community, or discourse. Conversely, a personal archive could be as informal as a collection of home videos and keepsakes and is not governed by the structure that an institutional archive has.

A digital archive aims to preserve information digitally rather than physically, like the Library of Congress Digital Collections. Digital archives can contain both digitized and born digital materials; this thesis works with digitized recordings. As such, digital archives have the ability to tell stories through all forms of media as well

as tell stories and provide information in a more widely accessible way than a physical archive. Anyone with internet access can view public digital archives, or even create their own to tell their own stories.

Within the digital archive movement, community archives have thrived. Flinn describes community archives as, "...the grassroots activities of documenting, recording and exploring community heritage in which community participation, control, and ownership of the project is essential" [10]. Community archives are more inclusive and diverse than traditional or institutional archives, which typically only told the stories of the victors in history. As such, community archives can reintroduce marginalized stories to the narrative. Archives created by and maintained by minority groups present a unique opportunity to preserve and tell the stories that never made it to a traditional archive.

The data for this thesis comes from a series of over 200 cassette tape interviews that Dr. Clarence G. Williams conducted in the late 1990s to document the Black experience at MIT. These interviews were part of the MIT Black History Project, which was founded in 1995 by Dr. Williams with the goal of bringing, "...an additional humanistic interface to one of the world's premier science, engineering, and research driven educational institutions" [1]. These interviews tell the unique story of the Black experience at MIT over a period of over 50 years, and as such contribute to the larger story of the Black experience at primarily white institutions. The MIT Museum has now digitized the tapes, and is preparing an exhibit with them. They also want to have an accompanying digital archive for the digitized interviews that would allow users to further explore the content outside of the exhibit. The primary goal of the digital archive is to have users explore the original interview audio in short segments with enough context that will allow them to build their own understanding of the interviews and discover connections they would not find without exploring multiple interviews.

This thesis will focus on first automating ways to divide the tapes into smaller audio segments for a digital archive, and second using topic modelling and classification to find connections and similarities between segments of different interviews in

order to enhance and speed up the curatorial process for a digital archive consisting of these tapes as well as enable unique ways to explore a digital audio archive of oral histories. The goal of this thesis is explore how these computational methods could contribute to the development or organization of the MIT Museum exhibit or any future digital archive with these interviews. Additionally, this thesis explores how segmenting these interviews and detecting topics in the segments could allow users of a future digital archive to explore the interviews in a more flexible way, without having to listen to interviews in full to discover new topics and connections. This thesis also discusses the possibilities of these computational methods to enhance any digital oral history archive by creating ways to define, detect, and discover topics and connections between audio using those topics.

Chapter 2

Background and Related Work

2.1 The MIT Black Oral History Project

In the mid-to-late 1990s, Dr. Clarence G. Williams conducted 181 interviews of MIT students, alumni faculty, and administration, both black and non-black, who were involved in the MIT Black community. In 2001, 81 of these interviews, originally recorded on mini-cassette tapes, were transcribed, edited, and published in the book *Technology and the Dream: and the Dream: Reflections on the Black Experience at MIT, 1941-1999* [35]. The remaining 100 interviews were published on an accompanying CD to the book.

In 2021, the MIT Museum began work on an exhibit on the Black experience at MIT that incorporates these interviews. During the Fall 2021 semester, the MIT Museum collaborated with the course CMS.635/835: Designing Active Archives to design an online, active archive for the interviews that could accompany the exhibit. This thesis experiments with various computational ways to achieve the goals of this future archive. Ten interviews were selected for experimentation in this thesis; the criteria for the selection as well as more information on these interviews and the future archive are discussed in Chapter 3.

2.2 Storytelling in Archives

The data for this archive presents a unique opportunity to give a platform to minority voices at a primarily white institution as well as preserve the history of that minority group's experience at MIT. Another aspect of the data that makes it especially empowering to the community is that the interviews were conducted within the Black community at MIT by members of the Black community at MIT. Shilton and Srinivasan explain the goal of representative archiving as, "to preserve the articulation of community identity" and define an empowered narrative as, "...records and histories spoken directly by traditionally marginalized communities, embedded within the local experience, practice, and knowledge of that community" [31]. This future archive has the potential to tell the history of the MIT Black community in an empowered, representative way, and uplift those traditionally marginalized voices to share the experience that would otherwise be forgotten.

This thesis was done with the goal of letting these interviews speak for themselves as much as possible. The technical work done on the interviews did not alter the interviews other than segmenting them in order to preserve the work done by Dr. Williams and the MIT Black community to gather these interviews.

2.3 Audio Segmentation

The full interviews can be up to multiple hours in length, so the MIT Museum has been manually editing the interviews into smaller clips for use in their exhibition. The smaller clips would be easier for the exhibition's audience to understand and explore than a full interview. This thesis experiments with different approaches to segmenting the audio automatically as a way to save time when curating the interviews. The smaller clips would allow users of the future digital archive to explore the archive without having to listen to an interview in full, similar to the purpose of the smaller clips at the physical exhibition.

This thesis experiments with segmenting audio files using only the audio files, and

only transcribing the audio after the segmentation. The MIT Museum's exhibit and the future archive both plan to use the original audio files as the primary materials, so I wanted to develop a method for segmenting the long files into shorter ones without needing to transcribe them first. One technique for audio segmentation is to split by "silence" in the audio. Grammatically, punctuation marks signify where one should pause or take a breath when reading aloud. Thus, a period of "silence" in an audio can typically be equated to the end of a sentence, phrase, or thought. So, splitting by silence could keep the context and content necessary to understand a shorter clip of audio without listening to the entire interview. The Python library PyDub [28], developed by James Robert, can detect a silence in a given audio file given the silence threshold and the minimum length of a silence. Using these parameters, any segment of audio that has a length greater than or equal to the minimum length of a silence and that has a decibel value of less then or equal to the silence threshold will be considered silence.

Another technique commonly used to segment audio is speaker diarization. Park et al. describe speaker diarization as, "...a task to label audio or video recordings with classes that correspond to speaker identity, or in short, a task to identify 'who spoke when'" [21]. In other words, speaker diarization divides the audio into segments by speaker. Wang et al. explain the four steps of a typical speaker diarization system as follows [33] :

1. Speech segmentation: Segment the input audio into short sections that contain a single speaker

2. Audio embedding extraction: Extract audio features from the segmented audios

3. Clustering: Determine the number of speakers and cluster the segments together by speaker

4. Resegmentation: Output the audio segmented by speaker

Commercial speech-to-text services like Otter.ai perform speaker diarization when transcribing audio in real time or from an uploaded file. However, the services have

no information on how many speakers are in an audio file when it performs the diarization. This thesis explores how speaker diarization may be improved when the number of speakers is known ahead of diarization.

2.4 Speech to Text

Since the data for this archive is all audio-based, transcripts must be made of the cassette tape interviews to perform text-based analysis on the interviews. Plenty of pre-made speech to text APIs exist for this purpose, like the Google Cloud Speech-to-Text API, Amazon Transcribe on AWS, or IBM Watson Speech-to-Text. Kim et al. compared the performance of many of these automatic speech recognition systems against each other and human transcriptions, and found overall that human transcription still remains the most accurate [13]. However, commercial speech-to-text services like Otter.ai and HappyScribe provide extremely accurate speech-to-text results that typically out-perform the open source APIs. Keeping ethics and accessibility in mind for this archive, transcripts for the audio-only resources are necessary and could benefit the archive of these interviews in the future. Additionally, the transcripts of the interviews will be necessary to use natural language processing techniques to further analyze the interviews and find meaningful connections.

2.5 Metadata and Natural Language Processing

There are four main types of metadata: descriptive, structural, administrative, and markup languages [27]. Descriptive metadata is used structural metadata is used to create or explain relationships between parts of resources, for finding and understanding a resource, administrative metadata contains the information necessary to contain and view a digital file, and markup languages interact with the content of a file and can explain things like formatting of text. This thesis only works with descriptive metadata. Natural language processing techniques used on the transcripts of the interviews can extract some keywords and named entities that we can use as

some of our descriptive metadata for each interview.

In *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*, Bird et al. discuss various different natural language processing techniques as well as explain how to implement them using the Python Natural Language Toolkit library [4]. One technique that could be utilized to extract descriptive metadata from the interview transcripts is named entity recognition. Bird et al. define named entities as proper nouns or noun phrases that refer to specific people, places, dates, organizations, etc. Named entity recognition aims to identify all instances of these named entities in a text. After finding the named entities in a text, relation extraction could be used to find connections and relationships between the named entities through regular expressions.

Another natural language processing technique that can be used to find connections between texts or segments of texts is text vectorization, which converts text into a numeric representation. Benfort et al. describe a few common techniques for encoding text vectorization [3]. Frequency vector encoding fills a vector with the frequency of each word in the text. One-hot encoding uses a vector as a corpus and gives each word a values of 1 if it is present in the given text and a 0 if it is not present in the text. Term frequency-inverse document frequency (TF-IDF) encoding fills the vector with the relative frequency of a word in the given text compared to other texts in the corpus. This TF-IDF relative frequency is calculated [15]:

$$tfidf(t, d, D) = tf(t, d) * idf(t, D) \tag{2.1}$$

Where t is the term we are counting the frequency of, d is the document we are reading, and D is the corpus. The term frequency is calculated:

$$tf(t, d) = 1 + log(f_{t,d}) \tag{2.2}$$

Where $f_{t,d}$ is the number of times the term t appears in the document d. The inverse

document frequency is calculated:

$$idf(t, D) = log(1 + \frac{N}{n_t}) \tag{2.3}$$

Where N is the number of documents in the corpus D and n_t is the number of documents that contain the term t. Because of the logarithmic functions, the TF-IDF value will always be in the inclusive range [0,1]. A value closer to zero means the term is less informative for a document, and a value closer to one means the term is more informative.

Finally, distributed representation encoding fills a vector with word similarities between documents in the corpus, unlike the individual word values for each individual document in the previous three encoding methods. The word2vec algorithm [17] and the GloVe algorithm [23] are popular implementations of distributed representation encoding.

This thesis experiments with one-hot encoding, frequency encoding, TF-IDF encoding, and implementations of the word2vec and GloVe algorithms.

2.6 Topic Modeling and Coherence

Topic modeling is an unsupervised machine learning algorithm that reads in a set of documents, detects patterns of words and phrases in the documents, and outputs groups of keywords that define "topics" in that set of texts. There are two popular approaches to topic modeling: latent semantic allocation or indexing (LSA or LSI) and latent dirichlet analysis (LDA).

The documents used in both algorithms are raw texts, so they require pre-processing before beginning the modeling. First, punctuation marks and stop words are removed from the texts. Next, the texts are tokenized into a list of words. From there, the individual words are stemmed or lemmatized. In the English language, stemming is the process of removing the end of a word to get the root of that word, while lemmatizing is the process of using vocubulary and analysis to remove just the suffix of a

word to return the base form of that word [15]. For other languages, the approaches to pre-processing may require different steps.

The LSI approach developed by Deerwester et al. [9] is based on the distributional hypothesis, the idea that words with similar distributions have similar meanings, derived from Harris' work in semantics [11]. In LSI, each document is a bag of words, and the frequency of each word is computed with the TF-IDF method described in Equations 2.1, 2.2, and 2.3. A matrix is then filled with the TF-IDF value of each term in each document. Next, the matrix is decomposed into a product of three matrices (the original document-term matrix, a document-topic matrix, and a term-topic matrix) using singular value decomposition. Using the largest TF-IDF values from the document-topic and term-topic matrices, we can find the most frequent keywords in the original document-term matrix and use those to define the topics.

The LDA approach developed by Blei, Ng, and Jordan [5] is also based on the distributional hypothesis. Similar to LSI, each document is treated as a bag of words. However, LDA creates n-grams of words, or a continuous sequence of n words, to create topic groups. Additionally, LDA assumes that both the distribution of topics in a document and the distribution of words in a topic are Dirichlet distributions, causing the sum of the weighted keywords of all the topics output by LDA to be one.

Topic Coherence is a score regularly used as a metric to say how good or bad a topic model's output topics are in describing the given corpus. This thesis will use two popular Topic Coherence measures: C_v coherence and UMass coherence. Röder et al. found these two measures to be the most accurate [30] that are included in libraries like Gensim [25].

The C_v coherence score is based on the Pointwise Mutual Information (PMI) Score described below in Equation 2.4.

$$PMI_{cv}(w_i, w_j) = log(\frac{P(w_i, w_j) + \epsilon}{P(w_i) * P(w_j)}) \tag{2.4}$$

$P(w_i)$ is the probability of finding a word w_i in a document in the model, $P(w_j)$ is the probability of finding a word w_j in a document in the model, and $P(w_i, w_j)$ is the

probability of finding both words w_i and w_j in a document in the model. ϵ is a small value used to ensure that the logarithm of zero is never calculated. The C_v coherence is then calculated with this PMI score on a sliding window:

$$C_v = \sum_{i<j} PMI_{cv}(w_i, w_j) \tag{2.5}$$

In some cases, the C_v score uses word occurrence in external sources like Wikipedia to confirm the probabilities in the given corpus [19]. In the Gensim library, only the corpus is used for these probabilities.

The UMass coherence score is also based on a PMI Score, but the equation is slightly different than the one used for the C_v score:

$$PMI_{umass}(w_i, w_j) = log(\frac{P(w_i, w_j) + \epsilon}{P(w_j)}) \tag{2.6}$$

Then, given a list of words T with a length of M, the UMass score is calculated:

$$UMass(T) = \sum_{i=2}^{M} \sum_{j=1}^{i-1} PMI_{umass}(w_i, w_j) \tag{2.7}$$

2.7 Topic Classification

Topic classification has the same goal as topic modeling, to identify the topic or topics of a text, but unlike topic modeling, topic classification is a supervised machine learning algorithm. As such, the documents used to train a topic classification model must be tagged with pre-defined topics. Minaee et al. describe rule based systems and machine learning methods as the two primary ways to approach topic classification [18]. Rule based systems require extensive knowledge of the defined topics to write rules that classify texts as a topic. Conversely, machine learning methods are data-driven and can be trained to classify texts as a topic. This thesis will focus on using machine learning methods to perform topic classification.

Feed forward neural networks are a simpler approach to topic classification. The texts are pre-processed and vectorized with an encoding method like one-hot encoding,

TF-IDF, word2vec, or GloVe. The vectors are passed through the layers of the model, and the final layer performs the topic classification using a classification algorithm like logistic regression, Naive Bayes, or Support Vector Machines (SVMs). Logistic regression assumes a linear relationship between features and classes, and outputs a probability for the classification. Naive Bayes assumes that all the features of a piece of data are independent, calculates posterior probabilities for each feature, and uses those to classify the data. SVMs draws "boundaries" between classes and maximizes the difference between them. Each "boundary" is binary, so a multi-label classifier for n classes would be a combination of n support vectors [29].

Recurrent neural networks (RNNs) used for topic classification primarily use the structure of the text and word dependencies as features for the model. The texts are pre-processed and vectorized, similar to the feed forward neural networks, but the vectors focus more on the sequence of the words in addition to the content. Long Short-Term Memory (LSTM) models are a popular design for topic classification as it "remembers" long term dependencies it has seen in the documents it is trained on [18]. Additionally, RNNs with LSTM may perform better on long texts, like the Multi-Timescale LSTM neural network developed by Liu et al [14].

Chapter 3

Project Development

This chapter explains the design choices made during the project's development and discusses what paths were taken, what paths were not taken, and why, within the context of these interviews and the goals of the future archive.

3.1 Goals of the Future Archive

In the Designing Active Archives class, the following goals were established for the future archive:

1. The archive should serve as a complimentary, digital resource for the MIT Museum's exhibition on the MIT Black Experience.
2. The archive should use the unedited interview audio as the primary material.
3. The archive should allow users to explore the interviews without having to listen to a full interview in chronological order.
4. The archive should allow users to explore multiple topics across multiple interviews, without needing to hear a full interview to have the necessary context.
5. Users should be able to build their own understanding of the interviews and discover connections between interviews that they would not find by only listening to an interview in full.

3.2 Interview Selection

As previously mentioned, 181 interviews were conducted by Dr. Clarence G. Williams for the MIT Black Oral History Project. I chose to work with ten of these interviews for my experimentation and analysis. Table 3.1 lists the interviewee, the years the spent at MIT, their field of study while at MIT, and what degrees (undergraduate, master, or doctoral) they earned at MIT, if any. I selected these interviews with the goal of creating a digital archive of the interviews. I chose MIT students, both black and non-black as well as current students and alumni, as the primary audience for the archive. As such, all ten selected selected interviews are from individuals who had been MIT students. I chose five male interviewees and five female interviewees to have equal gender representation. Each individual had a different field of study at MIT, so the interviews cover a large spread of MIT's academic offerings. Additionally, the interviews include individuals who attended MIT for only undergraduate studies, attended MIT for only graduate studies, or attended MIT for both, so the interviews cover both undergraduate and graduate perspectives of MIT. Additionally, the collective time that the interviewees spent at MIT spans from 1941 to 1990, covering a range of generational and historical perspectives.

Interviewee	Years at MIT	Field of Study
Kofi Annan	1971-1972	Management (SM)
Gregory Chisholm	1969–1980 1982–1989	Mechanical Engineering (SB, SM, PhD)
Harvey Gantt	1968-1970	City Planning (MCP)
Gloria Green	1949–1952	Architecture (undergraduate studies)
Darian Hendricks	1985–1989	Art and Design (SB)
Shirley Ann Jackson	1964–1973	Physics (SB, PhD)
Denise Loyd	1981–1990	Aeronautics and Astronautics (SB, SM, PhD)
Jennie Patrick	1973-1979	Chemical Engineering (ScD)
Victor Ransom	1941–1943 1946–1948	Electrical Engineering (SB)
Jennifer Rudd	1964–1968	Life Sciences (SB)

Table 3.1: Ten Selected Interviews

3.3 Audio Segmentation Approaches

3.3.1 Split by Silence

The digitized interviews range from 20 minutes to over two hours in length. The attention span of the typical user is too short to listen to an interview in full, and especially too short to listen to multiple full interviews in one sitting. As such, I wanted to find a way to segment the interviews into shorter, digestible audio clips that still contained enough context so that the listener could understand the clip without having to listen to a full interview. Additionally, a main goal of the future archive is to allow users to explore multiple topics across interviews. The smaller clips, once categorized by topic as discussed in later sections of this thesis, could let users explore one audio segment on one topic before being recommended a similar audio segment that covers the same topic.

I chose to work with Python for all of the technical work in this thesis because of its wide variety of libraries [1] available for processing both audio and text and for creating various designs of neural nets for different data formats. All of my code and analysis did not need to run in real-time, so speed was a lower priority than accuracy, making Python a good choice for this work. I used the PyDub library [28] for the silence detection and audio editing portions of the work. PyDub supports any type of audio format that the ffmpeg library [32] supports, so I was able to try processing the files in both MP3 and WAV formats to see how compression affected the processing, if at all. It has the capability to process audio by the millisecond, so I could work with very small or larger increments of time. It also has built-in functions that could split by silence and allow you to define the maximum decibel level and minimum length a segment of audio must be in order to be considered silence. Figure 3-1 graphs the time vs. the decibel level of the first 50 seconds of an interview in the blue line. The red dashed line shows the maximum decibel at y=-34 db. If the minimum length of silence was defined as four seconds, the algorithm would detect silences between 20 and 25 seconds and 30 and 34 seconds because the decibel level was below the

[1] Appendix B lists all the libraries and packages used throughout this thesis

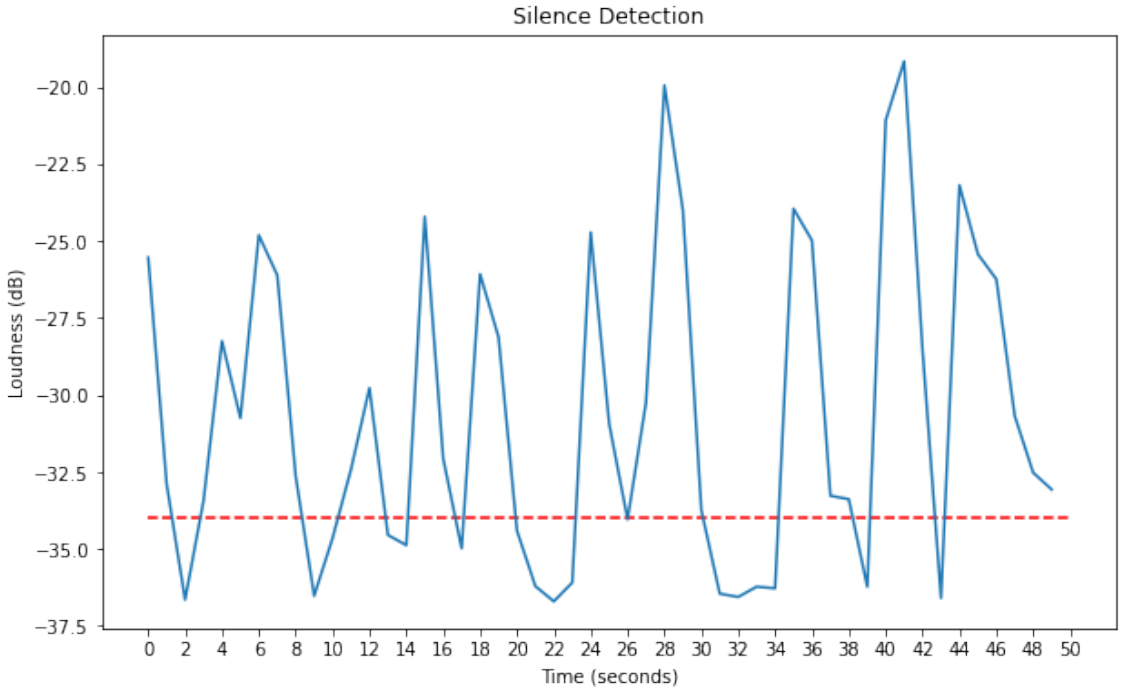

Figure 3-1: Silence Detection

maximum decibel level for at least four seconds at those times.

The first approach to segmenting the audio was to detect silences in the audio and split the interview into clips at those silences. Intuitively, silences tend to correlate with ends of sentences, phrases, or thoughts. The goal was that by splitting at silences, I could create clips that contained enough context to understand the content. However, there were three primary issues with the interviews that made this approach unsuccessful for this material. First, the interviews have a low audio quality since they were originally recorded on mini-cassette tapes, then digitized. CDs and digital audio formats like MP3 files have a range of 20Hz to 20kHz, which extends past the typical hearing range of an adult. Conversely, cassette tapes can't accurately play back audio below 40Hz and above 10kHz [7]. As a result, the static and low voice volume made it difficult to find a good definition for silence, if there was one to find at all. Second, the interviewees had no knowledge of the questions ahead of time, so their answers were given in a more stream-of-consciousness format. Because of this, silences were more sporadic and correlated more with pauses to think rather than pauses after the end of a sentence. Segments could be as long as ten minutes or as

short as two seconds. Splitting at these silences resulted in losing both context and valuable content. Finally, there was no consistent way to detect silence in different interviews. Each interview was recorded in a different location, at a different time of day, with different speakers, and with the recorder placed at different spots in relation to the speakers. As such, the ambient noise, the acoustics, and the level at which the interviewees spoke all varied between interviews, so there was no single frequency level that could be defined as "silence" for every interview. Additionally, as each interviewee spoke with different styles, the minimum length of silence varied between them as well. So, each interview had a different definition of silence that had to be found manually through trial and error, which was too time-consuming for the scope and goals of this thesis.

3.3.2 Voice Classification Model

The second approach I took was to first segment the audio by silence, using a high enough frequency as my silence level to create clips of around a minute or less, run those segments through a voice classification model [2], and regroup the segments by the model's predicted speaker. The goal of this approach was to segment the interviews by speaker, using silence to define the smaller segments first.

All but one of the interviews are between two people: Dr. Clarence G. Williams as the interviewer and a second speaker as the interviewee. The single outlier is an interview containing two interviewees. Using this information, I designed a voice classification model that, for each interview, would be trained on audio clips from the interview and classify a given clip as either Dr. Williams as the speaker or the interviewee as the speaker. I based the design of the model off of Arias' work on building a voice classification neural net for 30 different speakers [2]. Rather than creating one model to classify every speaker across multiple interviews, I chose to create a binary classification model for each interview, so as to decrease the chance of misclassifying one speaker as multiple different speakers.

[2] A neural net trained on different speaker's voices and used to predict which speaker spoke an input audio

For my first model, I used four fifteen second long clips for each speaker as my training data, and two sets of three fifteen second long clips for each speaker as my validation and test data. Each audio clip was labeled with its speaker, then the three data sets were put into separate data frames using Pandas [20] with columns for the audio clip's file name and the speaker of that clip. Because there were only two speakers in each interview I worked with, I used scikit-learn [22] to encode the data frame speaker labels as '0' for Dr. Williams and '1' for the interviewee. For each data set, I used Librosa [16] to extract multiple spectral audio features from the clips. The Librosa "mfcc" function extracted the mel-frequency cepstral coefficients (MFCCs) [3] of an audio. The "chroma_stft" function extracted a chromagram [4] from a waveform or power spectrogram. The "melspectrogram" function extracted a mel-scaled spectrogram [5]. The "spectral_contrast" function extracted the spectral contrast [6] of the audio. The "tonnetz" function extracted the tonal centroid features [7] of the audio.

Once the features were extracted, I used Keras [8] to build the neural net. Keras is a modular and user-friendly high-level library used to build neural nets that provides access to other open-source machine learning platforms. I used a sequential model with a dense input layer [8] using a rectified linear unit (ReLU) activation function [9]. I used two hidden layers that were also dense and used a ReLU activation function. The output layer was also dense, but used a sigmoid activation function [10] since the final output would be binary, or a prediction between the two speakers. I compiled the model using a binary cross entropy loss [11], once again because the model output would be binary, and I used the Adam optimizer [12]. I trained the model with a batch

[3] Coefficients that represent the short-term power spectrum of a sound
[4] A representation of an audio's spectral energy
[5] A spectrogram in the mel-scale
[6] The difference between the peaks and valleys in the audio's spectrum
[7] A representation of the tonal space of the audio
[8] A neural net layer that is heavily connected to its neighboring layers
[9] A piecewise linear function that outputs the input if positive, zero otherwise
[10] A sigmoid function used to determine the output of a neuron
[11] The mean of the negative log of the probabilities output by the neural net, used to train the model on its incorrect and correct predictions
[12] An optimization algorithm used for training neural nets

size [13] of 256 and 50 epochs [14].

Once the model was trained, I used it to try and improve upon the segments found by silence as discussed in the previous section. I first segmented the interviews by silence, then used the voice classification model to predict which speaker was speaking in that audio clip. I would then concatenate any chronologically neighboring audio clips that were predicted to have the same speaker, in an attempt to segment the interview by speaker. Using the interview between Dr. Williams and Kofi Annan as an example, I used a silence length definition of one second that resulted in a model accuracy of 84% in predicting the audio clip speaker. I also used a silence length of one and a half seconds that increased the model accuracy to 89%.However, because the model had a relatively low accuracy, audio clips were easily misclassified. This, in addition to the large variation in clip length that the silence detection created, exacerbated the disparity in clip length.

For my second model, I changed my training, validation, and test data to all be two second clips of audio. I used 120 clips for training data, and two sets of 40 clips for validation and testing data. I used the same features, neural net details, and training technique as the first model. Once the model was trained, I used it in my speaker diarization system described in the next section.

3.3.3 Speaker Diarization

The final approach I took was to create a speaker diarization system that would segment the interviews by speaker. The algorithm I used for each interview for my first speaker diarization approach is as follows:

1. Train the voice classification model on audio clips of n=2 seconds from the interview, labeled with the speaker of that audio clip, as discussed in the previous section

2. Divide the interview into clips with a duration of n=2 seconds

[13] How many samples from the training data are used in each round of training
[14] The number of rounds the neural net is trained on

3. Classify each n=2s audio clip using the voice classification neural net

4. Iterate through the list of audio clips and find which neighboring clips have the same speaker

5. Concatenate the audio clips of the same speaker and export

Training the voice classification model this way and using it on shorter clips resulted in 97.6% accuracy for the previously discussed interview of Kofi Annan, which was a significant improvement on applying it to longer split-by-silence clips.

This approach had promising results as it successfully detected each "question" in the interviews, but there were many singleton audio clips with two second lengths that had no neighbors with the same speaker. These singleton audio clips were generally caused by active listening throughout the interview. Consistently throughout every interview, Dr. Williams interjects with a few "mmhmm"s or "yeah"s to indicate to his interviewee that he is following along with their thoughts. Since this adds to the personal and intimate nature of listening to these interviews, we wanted to keep these small clips with the surrounding context of the interviewee's answer that Williams is listening to rather then separating them from that context. To achieve this, step 5 of the above algorithm needed to be modified to account for a minimum length a group of audio clips must be to be concatenated together. In this case, a minimum length of ten seconds generally indicated that the speaker was contributing something more substantial than an act of agreement. So, step 5 of the above algorithm became:

5. Concatenate neighboring audio clips for Speaker 1 (Dr. Williams, the interviewer) that have a minimum length of ten seconds

6. Concatenate the remaining clips for Speaker 2 (the interviewee) and export

This was the final approach I chose to segment the interviews for analysis in future sections. The results of this approach are discussed in more detail in Chapter 4.

3.4 Speech to Text Approaches

The book *Technology and the Dream: Reflections on the Black Experience at MIT, 1941–1999* by Clarence G. Williams [35] contains transcripts of a selection of the 181 interviews conducted, but the transcripts were edited for clarity and readability. Because one of the future archive goals is to use the original interview audio, the transcripts that I experimented with later in this thesis needed to match the audios. As such, I chose to transcribe the smaller segments of the interviews that were segmented by speaker as described in the previous section.

I first tried using open source speech-to-text APIs like the Google speech-to-text API to transcribe the interviews. Smaller clips of audio had a higher success of transcription than longer clips due to the limitations of the open source APIs, so clips would require extra segmentation in order to achieve the best accuracy, but this was counter-intuitive to the original segmenting by speaker. Additionally, I found that the graininess and quietness of the interviews, resulting from its original cassette format, resulted in a high lack of accuracy in the transcription. In order to improve the audio quality for the transcription, I ran it through high and low-filter passes using the PyDub library to remove some of the white noise. A high pass filter removes any noise below a certain frequency by attenuating those frequencies and selecting the frequencies above. Conversely, a low pass filter removes any noise above a certain frequency by attenuating those frequencies and selecting the frequencies below [34]. The high pass filter would reduce the white noise, and the low pass filter would balance the audio after the high pass filter. This improved the accuracy of the transcription. The final issue with the interviews was the various accents of the speakers. The open source APIs often handled accents with little accuracy, thus losing the meaning of a sentence or a phrase with as little as one incorrectly transcribed word. One memorable instance was an interviewee, Philip Clay, who had a heavy southern accent saying, "I made honor society" being transcribed as, "I Moderna society".

Ultimately, commercial speech-to-text programs like Otter.ai and HappyScribe resulted in the highest level of accuracy due to their superior computational resources

and proprietary algorithms. Rather than further experiment with open source APIs in an attempt to replicate this accuracy, I chose to move forward with these commercially generated transcripts for my analysis. Appendix A contains samples of these transcripts generated by Otter.ai.

3.5 Named Entity Recognition and Text Vectorization

Once I generated the transcripts for the segmented audio clips, I could begin to extract descriptive metadata using natural language processing techniques. I chose to work with spaCy [12], an open-source natural language processing Python library, because it provided robust, pre-trained pipelines in addition to the API. All of the interviews I worked with were recorded in English, but spaCy provides pipelines in 19 different languages, so this work could be easily applied to other oral history projects recorded in languages other than English. Additionally, the English pipeline could be downloaded with a small, medium, or large corpus, giving more flexibility in trade-offs between speed and corpus size. I chose to work with the large corpus pipeline since none of the metadata extraction had to happen with a real-time user, so speed was not as much of a priority as accuracy.

I first tried named entity recognition to extract proper nouns and phrases that referred to specific, people, places, dates, etc. The left column of Transcript A.1 is a transcript generated by Otter.ai of the first question Dr. Clarence G. Williams asks in his interview with Kofi Annan, and the left columns of Transcripts A.2 and A.3 have Kofi Annan's answer in two parts. Table 3.2 shows the results of running named entity recognition on the full transcript of Dr. Williams' first question, and Table 3.3 shows the results of running named entity recognition on the full transcript of Kofi Annan's answer.

Based on these results, the entity type categories that provide the most unique and descriptive metadata are GPE (Geopolitical Entity), DATE, NORP (National-

Named Entity	Start Character	End Character	Entity Type
One	43	46	CARDINAL
Ghana	278	283	GPE
the 1940s	287	296	DATE
50s	301	304	DATE

Table 3.2: Named Entity Recognition of Dr. Williams' Question

Named Entity	Start Character	End Character	Entity Type
a few years ago	21	36	DATE
91	61	63	CARDINAL
19	83	85	DATE
Ghana	237	242	GPE
three	342	347	CARDINAL
five years ago	421	435	DATE
Ghana	548	553	GPE
Ghana	780	785	GPE
the 40s	790	797	DATE
50s	807	810	DATE
British	1040	1047	NORP
de Jenkins	1469	1479	PERSON
three	1786	1791	CARDINAL
three feet by	1793	1806	QUANTITY
three feet	1807	1817	QUANTITY
no Reader's Digest	2338	2356	ORG
UN	2404	2406	ORG

Table 3.3: Named Entity Recognition of Kofi Annan's Answer

ities or Religious or Political Groups), and ORG (Organization). CARDINAL and QUANTITY need more context to provide meaning for these pieces of text. Additionally, the raw NER extraction does not remove duplicates, but counts each time any named entity appears.

An important aspect of interactive archives is the ability to find connections between artifacts, and an easy connection for a computer to find between texts is how similar they are. spaCy's pipelines include a "similarity" function that calculates the similarity between two vectors. To test it, I computed the similarity between the transcripts of Dr. Williams' question and Kofi Annan's answer using the function.

The result said the two texts were 98.448% similar. Since they are a matching question and answer, intuitively we can say that they should be similar, but 98.448% seemed too high a similarity for the large difference in text length and content. I then removed the stop words, or the most commonly used words in the English language, from the two transcripts and calculated the similarity to be 93.114%. The transcripts with the stop words removed can be seen in the right hand columns of Transcripts A.1, A.2, and A.3. By removing the stop words, the similarity decreased as we would intuitively expect, but the number still seemed to high. In an attempt to understand the similarity function, I worked with a toy example using the two sentences "You are not as funny as them" and "I went to the store". Their similarity was 75.194%. Removing their stop words made the sentences become "funny" and "went store", now with a similarity of 28.097%. In both cases, each sentences uses words unique from the other, including the stop words. Additionally, the sentiment of each sentence was clearly different from the other, but with the stop words they were still considered highly similar. With these results, I chose to experiment with text vectorization as a way to discern similarity between texts.

For text vectorization, I chose to work with the Natural Language Toolkit (NLTK) [4], another open-source natural language processing Python library. Similar to spaCy, NLTK, has multiple trained models and corpora in multiple languages and sizes, to allow for flexibility in this thesis and other oral history projects done in other languages. Figure 3-2 shows frequency encoding and TF-IDF encoding used to vectorize three sentences. The stop words were removed from the three sentences, and the corpus was created as a bag of words of the three sentences. The frequency encoding example shows how similarities can be found between different sentences, with the "parents" column being an example of an overlap between the sentences. The TF-IDF example shows how vectorization can show how important a word is to an individual sentence, with "uneducated" and "people" being nearly twice as important as "parents" in the third sentence from Patrick.

The results of this metadata extraction and experimentation with these natural language processing techniques are further discussed in Chapter 4.

```
Jackson :  I grew up in Washington, DC, in a family of six— four children and our parents.
Hendricks :  My parents are both from Trinidad.
Patrick :  My parents were uneducated people.

Frequency Encoding:
 children   dc   family   grew   parents   people   six   trinidad   uneducated   washington
----------  ---  -------  -----  --------  -------  ----  ---------  -----------  -----------
        1    1        1      1         1        0     1          0            0            1
        0    0        0      0         1        0     0          1            0            0
        0    0        0      0         1        1     0          0            1            0

TF-IDF Encoding:
 children       dc   family     grew   parents   people      six   trinidad   uneducated   washington
----------  -------  -------  -------  --------  -------  -------  ---------  -----------  -----------
 0.396875  0.396875  0.396875  0.396875  0.2344   0        0.396875  0           0            0.396875
 0         0         0         0         0.508542 0        0         0.861037    0            0
 0         0         0         0         0.385372 0.652491 0         0           0.652491     0
```

Figure 3-2: Text Vectorization Examples

3.6 Topic Modeling and Classification Approaches

3.6.1 Topic Modeling using LDA and LSI

To perform topic modelling, I used the spaCy large English corpus, the NLTK stop words corpus, and Gensim [26]. Gensim is an open-source Python library used to process texts using various unsupervised machine learning algorithms. Gensim includes built in functions for Word2Vec, Latent Semantic Indexing (LSI), Latent Dirichlet Allocation (LDA), and more, allowing for flexibility in the methods of pre-processing and unsupervised analysis that can be performed on raw text.

After segmenting the interviews and transcribing the segments, I put the transcripts into a JSON file as my corpus and converted the JSON into a pandas [20] dataframe. I first chose to run LDA on the interview segments, and based the design off of Prabakaran's work with topic modelling emails [25]. To pre-process my raw transcripts, I removed any newline characters, then ran Gensim's "simple_preprocess" function to remove punctuation from the text then tokenize it. Next, I used removed the stop words from the text, used Gensim's pre-trained "Phrases" model to detect bigrams in the text, and lemmatized the text. Once the pre-processing was done, I used Gensim's "Dictionary" function to create a dictionary from the corpus. Finally, I used Gensim's "LdaModel" to build the LDA model for my dictionary and pre-processed corpus.

Topic	Topic Equation
1	0.033*"african" + 0.030*"wife" + 0.029*"imagine" + 0.022*"discuss" + ' '0.019*"okay" + 0.017*"country" + 0.014*"effect" + 0.013*"want" + ' '0.013*"leader" + 0.013*"talente"
2	0.033*"work" + 0.024*"first" + 0.018*"change" + 0.017*"graduate" + ' '0.017*"professor" + 0.015*"course" + 0.015*"undergraduate" + 0.015*"parent" ' '+ 0.014*"later" + 0.014*"figure"
3	0.048*"school" + 0.030*"kid" + 0.019*"remember" + 0.018*"father" + ' '0.016*"go" + 0.016*"neighborhood" + 0.016*"math" + 0.015*"move" + ' '0.015*"high" + 0.015*"grow"
4	0.048*"period" + 0.026*"medicine" + 0.023*"biology" + 0.021*"college" + ' '0.021*"pre" + 0.019*"significant" + 0.019*"early" + 0.017*"finished" + ' '0.017*"maintain" + 0.016*"pharmacist"
5	0.030*"take" + 0.025*"energy" + 0.022*"care" + 0.020*"baptize" + ' '0.015*"end" + 0.015*"tree" + 0.011*"academically" + 0.011*"read" + ' '0.010*"clear" + 0.009*"pass"
6	0.044*"opportunity" + 0.029*"important" + 0.025*"motivation" + 0.018*"give" ' '+ 0.018*"interest" + 0.017*"architecture" + 0.017*"thing" + 0.017*"ability" ' '+ 0.016*"step" + 0.015*"science"
7	0.046*"know" + 0.042*"people" + 0.037*"think" + 0.022*"thing" + ' '0.013*"sense" + 0.013*"win" + 0.012*"play" + 0.012*"enjoy" + 0.012*"last" + ' '0.011*"lose"
8	0.063*"say" + 0.026*"guy" + 0.020*"person" + 0.019*"people" + 0.019*"tell" ' '+ 0.017*"really" + 0.014*"black" + 0.013*"white" + 0.013*"realize" + ' '0.011*"body"
9	0.027*"work" + 0.018*"physics" + 0.016*"mit" + 0.014*"time" + 0.014*"think" ' '+ 0.013*"get" + 0.012*"student" + 0.012*"people" + 0.011*"thing" + ' '0.009*"course"
10	0.073*"know" + 0.043*"go" + 0.026*"get" + 0.022*"think" + 0.019*"say" + ' '0.018*"school" + 0.016*"come" + 0.015*"really" + 0.014*"year" + ' '0.014*"student"

Table 3.4: LDA Topic Modelling Probability Equations

I generated 10 topics, which are shown in Table 3.4. The "Topic Equation" is an equation telling how much weight a keyword has for that topic, with the weights adding up to 0.1 for a single topic, and 1.0 for all ten topics. Because topic modeling is an unsupervised machine learning method, it represents the topics as these equations, but we can infer a more human-readable topic based on the equations. For example, topic 3 could be about an interviewee's early education and upbringing with keywords like "school" and "kid", while topic 6 could be about an interviewee's goals

and motivations with keywords like "opportunity", "important", and "motivation". However, some of the topics are not as clear to infer an understanding for, like topic 5, which has keywords that intuitively appear highly unrelated. Additionally, some of the topics seem to overlap with each other like topics 7 and 10, which both have "know" and "think" as highly weighted keywords, causing blurred lines with the topics instead of creating distinct topics. This model resulted in a C_v coherence score of 0.316 and a UMass coherence score of -4.0378, which are both too low for the level of topic detection I wanted to perform on the interviews.

I tried to create an LSI model next. I used the same pre-processing steps, then built an LSI model with ten topics using Gensim's "LsiModel". Table 3.5 lists the weighted keyword equations generated with LSI. More topics overlap with this LSI model than the LDA model, making the topics even less distint. For example, topics 1, 3, 4, 7, 9, and 10 all contain combinations of the keywords "mit", "school", and "student". In fact, six of ten of the topics contain the keyword "mit".

The LSI model resulted in a C_v coherence score of 0.266 and a UMass coherence score of -1.096, which are both still too low for the quality of topic detection I wanted. As such, I chose to experiment with topic classification to see if supervised machine learning would improve these results.

3.6.2 Topic Data Collection

In order to build a topic classification model, a set of topics had to be defined for the interviews. Every interview has a similar structure, with Dr. Williams asking about each interviewee's childhood, education, experiences at MIT, and careers. I defined ten topics that I found occurred the most throughout the interviews in Table 3.6. The "Topic Indicators" column lists what type of content in an interview segment indicated that it was about that topic, and the "# of Segments" column lists how many interview segments fell into that topic category. Three volunteers and I went through the transcripts of each interview segment from the ten segmented interviews and identified which topic or multiple topics were in that segment. One of my volunteers was an MIT 2020 alumnus, and my other two volunteers were parents of an

Topic	Topic Equation
1	0.614*"know" + 0.367*"go" + 0.209*"school" + 0.188*"get" + 0.183*"think" + ' '0.177*"say" + 0.147*"people" + 0.125*"really" + 0.123*"mit" + 0.118*"year"
2	-0.670*"know" + 0.248*"say" + 0.200*"black" + 0.151*"school" + 0.146*"work" ' '+ 0.144*"get" + 0.143*"time" + 0.137*"people" + 0.108*"kid" + 0.104*"come"
3	-0.370*"school" + 0.351*"say" + -0.260*"go" + 0.213*"people" + ' '-0.188*"remember" + -0.183*"kid" + 0.183*"thing" + -0.167*"parent" + ' '0.161*"student" + -0.159*"neighborhood"
4	-0.571*"say" + 0.281*"mit" + 0.228*"student" + 0.161*"work" + 0.159*"time" ' '+ -0.152*"person" + 0.141*"come" + 0.139*"program" + -0.122*"really" + ' '-0.120*"white"
5	0.282*"physics" + -0.268*"black" + -0.253*"people" + -0.221*"student" + ' '0.205*"course" + -0.164*"come" + 0.161*"material" + 0.151*"take" + ' '0.149*"apply" + 0.144*"think"
6	0.294*"work" + 0.209*"physics" + -0.194*"really" + -0.193*"think" + ' '-0.185*"interest" + -0.179*"father" + -0.178*"anyway" + -0.167*"mit" + ' '-0.158*"place" + -0.146*"catholic"
7	-0.245*"physics" + -0.245*"student" + 0.229*"go" + 0.219*"work" + ' '-0.212*"black" + 0.197*"think" + 0.194*"get" + -0.187*"school" + ' '-0.178*"course" + -0.173*"mit"
8	-0.433*"go" + 0.268*"think" + 0.250*"people" + 0.192*"course" + ' '-0.188*"scholarship" + 0.163*"know" + -0.158*"mit" + 0.141*"always" + ' '-0.136*"say" + -0.120*"come"
9	0.252*"go" + -0.238*"school" + -0.184*"thing" + 0.176*"student" + ' '-0.165*"really" + 0.146*"think" + 0.142*"come" + 0.138*"remember" + ' '-0.126*"high" + -0.123*"mean"
10	0.262*"people" + 0.243*"class" + -0.195*"think" + 0.174*"course" + ' '-0.148*"need" + -0.136*"interest" + 0.130*"student" + 0.125*"go" + ' '0.122*"first" + -0.116*"mit"

Table 3.5: LSI Topic Modelling Probability Equations

MIT student, so all familiar with the MIT-specific content the interviews contain. Additionally, two of my volunteers were male and one was female, so as to have equal gender representation in the topic identification. Figure 3-3 graphs the frequency of the topics in the segments from the ten interviews. Each bar represents one of the ten topics and the height shows how many segments were tagged with that topic. The ten interviews did not provide an even spread of the topics, with "MIT" and "Education" each having well over 100 segments while "Career", "Advice", "Anecdote",

and "Accomplishment" all had less than 50 segments each.

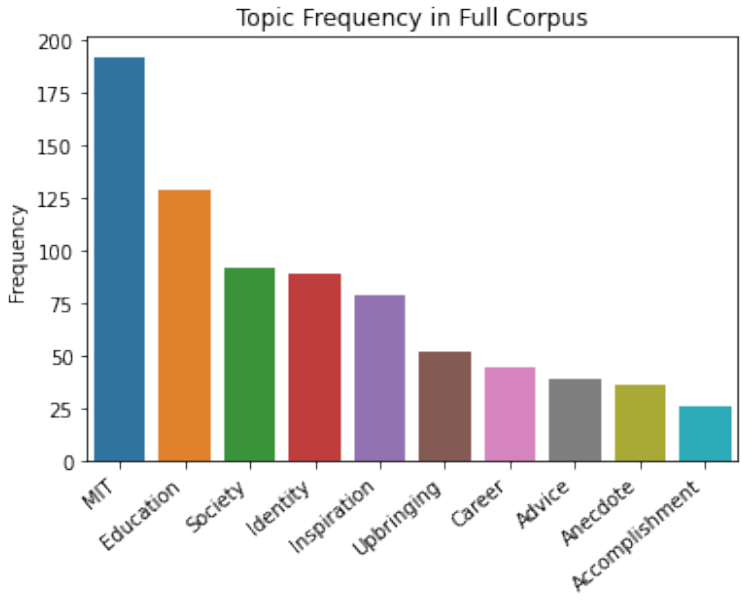

Figure 3-3: Topic Frequency in Full Corpus (all segments)

Topic	Topic Indicators	# of Segments
Upbringing	Early life and childhood, family	52
Education	Elementary, middle, high school, or college	129
Society	Race, gender, historical events, opinions on society	92
Identity	Personal race and gender, self-view, personality traits	89
Inspiration	Role models, inspiring events, goals, motivation	79
Accomplishment	Awards, degrees, "first"s (e.g. first Black woman to graduate MIT)	26
MIT	Experiences at MIT, organizations at MIT, anything related to MIT	192
Career	How they chose their career, how they started, where they are now, career goals	44
Advice	Advice they've received and advice they've given	39
Anecdote	Anecdote or story going into detail on their interactions with other individuals	36

Table 3.6: List of Topics Defined by Me

3.6.3 Topic Classification Approaches

I chose to experiment with machine learning systems to create a topic classification system.Using the transcripts tagged with the ten topics described in the previous section, I first pre-processed them in a similar way as I did for topic modeling. I converted the text to lowercase, removed the punctuation, and removed the stop words using the NLTK stop word corpus. I used the NLTK to tokenize the remaining words, used the NLTK Snowball stemmer [24] to stem the words, and used the NLTK WordNet corpus and WordNetLemmatizer to lemmatize the words.

I used two text encoding methods: Scikit-learn's TF-IDF vectorizer and Gensim's word2vec vectorizer. I used three pre-trained feed forward neural networks with the following classification algorithms: Logistic Regression, Naive Bayes, and Support Vector Machine (SVM). Overall, I created six models using every combination of the vectorizers and models. For naive Bayes, I used a multinomial model when using TF-IDF vectorization and a Gaussian model when using word2vec vectorization. For the SVM, I used a radial basis function (RBF) kernel.

I trained each of these models with three different approaches to separating the training, validation, and testing data from the total 778 tagged transcripts. I first created my training, validation, and test data by randomly selecting 64% of the tagged transcripts as training data, 16% as validation data, and 20% as testing data. The random selection ensured that each data set received a proportional amount of each topic. However, because each transcript could be tagged with more than one topic, transcripts could appear multiple times in the data sets with different topics. If transcript A appeared in the total tagged transcripts two times, once tagged with "MIT" and once tagged with "Career", but one copy was given to the training data set and one copy was given to the testing data set, the model would learn transcript A as "MIT" and correctly predict its topic to be "MIT", but the testing data set would say the correct topic is only "Career". The model would be ultimately be confused with these duplicates, resulting in a very low and inaccurate accuracy score.

For my second approach, I wanted to remove this issue of duplicate transcripts. I

separated my tagged transcripts into a JSON file that contained only transcripts that appeared multiple times under different topics and a JSON file that contained only transcripts that appeared once. There were 617 total transcripts in the file of duplicates and 160 transcripts in the file with no duplicates, resulting in a 79.305%/20.695% split. I separated the tagged transcripts that had duplicates into 80% for my training data and 20% for my validation data. I used the transcripts that had no duplicates and only appeared once as my testing data. Figure 3-4 shows the distribution of topics in the set of transcripts with duplicates and set of transcripts without duplicates. The distribution of both is similar to the distribution of all 778 topics as seen in Figure 3-3, so this data splitting could result in more accurate training results.

For my third approach, I tried removing the duplicate transcript issue in a different way. I created a JSON file that removed all but one of every duplicate, resulting in 397 total files. I randomly selecting 64% of the tagged transcripts as training data, 16% as validation data, and 20% as testing data. Figure 3-5 shows the topic distribution of these 397 files, which still has a distribution similar to Figure 3-3, so this data could result in accurate training results as well. However, since the data set is significantly smaller than the full corpus, it could lose some accuracy.

The results of these six models and the three data set approaches are further discussed in Chapter 4.

The next machine learning method was to create and train a simple feed forward neural net on the tagged transcripts rather than use a pre-trained model. Because one transcript could be tagged with multiple topics, I first used Pandas to reshape the data. Rather than transcript A appearing three times in the data with topics 1, 2, and 3, transcript A appeared once in the data tagged with the list of topics [1,2,3]. This reduced the rows of data from 778 to 397, but the data could now take advantage of the multiple topics that one transcript could have. I used the Sci-kit learn "test_train_split" function to randomly divide the data into 70% training, 15% validation, and 15% testing data.

I used One-hot encoding to vectorize the list of topics for each transcript. For example, if the list of all potential topics was [MIT, Career, Advice], a transcript tagged

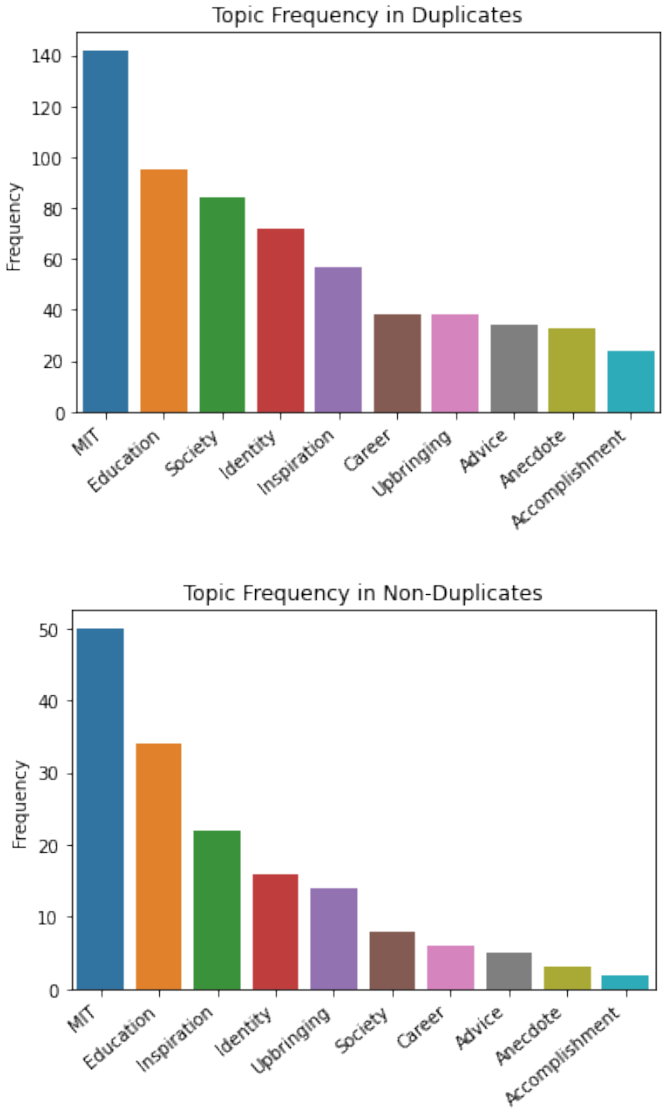

Figure 3-4: Topic Frequency in Transcripts with Duplicates vs. No Duplicates

with the topics "MIT" and "Career" would have its topics vectorized as [1,1,0]. Next, I pre-processed the transcripts by making them lower case, removing the punctuation, and tokenizing the words.

For my first iteration of the model, I removed the stop words and stemmed and lemmatized the remaining words using the NLTK stop words, Snowball stemmer, and WordNet lemmatizer. I then created a vocabulary for the transcripts by removing any duplicates from the remaining words after the pre-processing. Using this vocabulary set, I created a TF-IDF text vectorizer using a Keras "TextVectorization" layer that

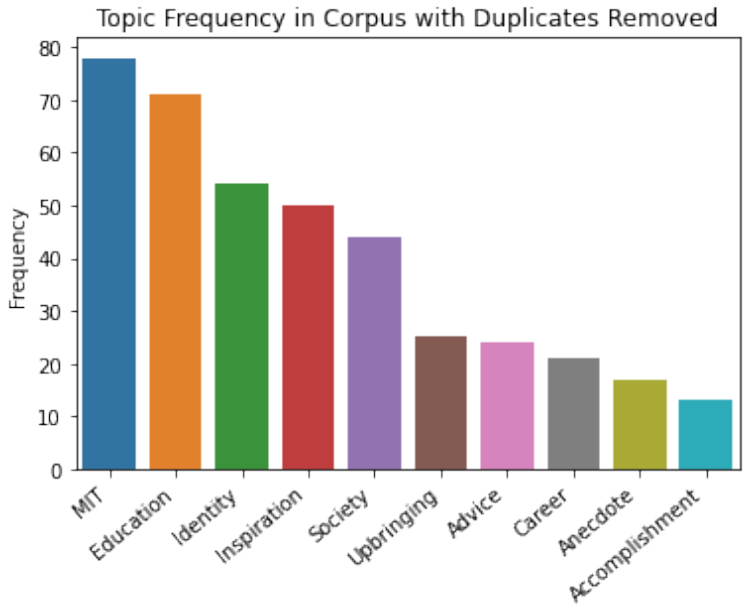

Figure 3-5: Topic Frequency in Corpus with Duplicates Removed

detected bigrams. I used this vectorizer to vectorize the transcripts in the training, validation, and testing data.

To create the model, I started with a Keras Sequential model that used the binary crossentropy loss and the Adam optimizer. I used three Dense layers: the first with an input size of 512, the second with an input size of 256, and the third with an input the size of the generated vocabulary. The first two layers used an ReLU activation and the last layer used a sigmoid activation. I trained the model for 20 epochs with a batch size of 64.

The full pre-processing resulted in my vocabulary having only 27 words in it, which was far too small, and resulted in a model accuracy of 25%. So, for my second model iteration I changed my pre-processing to only making the transcripts lower case, removing punctuation, tokenizing words, and removing duplicates. This resulted in a vocabulary of 6,336 words, and increased the accuracy of the model to 61.67%. In a third model interation I changed the data division to be 80% training, 10% validation, and 10% testing data and changed the TF-IDF to focus on single words rather than bigrams, which caused the vocabulary to increase to 7,066 words and the accuracy to increase to 65%. In a fourth iteration, I added two dropout layers to the model, and

trained the model for 40 epochs, which increased the accuracy to 70%.

Based on these results, I made a recurrent neural net (RNN) that would output, for a given transcript, a probability for each topic. This would allow the model to predict how much of a topic a transcript has, rather than just listing which combination of topics a transcript contained. I split the data into 64% training, 16% validation, and 20% testing. I one-hot encoded the topics for each text and pre-processed the text like I did in the previous model, resulting in a vocabulary size of 4,634.

The model included an input layer, an embedding layer, a long short-term memory layer, and a dense output layer with ten neurons to correspond to the ten possible topics. The embedding layer allowed the model to find patterns in the words of the transcripts. I used a pre-trained GloVe [23] vector to vectorize the transcripts for this embedding layer, based on the vocabulary size. I chose to use a pre-trained vectorizer to take advantage of the larger corpus it was trained on, compared to my relatively small vocabulary. The long short-term memory layer allowed the model to find patterns in the training transcripts based on word order. The output layer used a sigmoid activation. In comparison to the previous model, this model took advantage of more of the language structures in the text than using only dense layers. I used a binary crossentropy loss and the Adam optimizer, and trained the model for ten epochs with a batch size of 128. The model resulted in 42.5% accuracy, but the results were less ambigious and more human readable than the previous model. The results of both the sequential model and the RNN model are discussed further in Chapter 4.

Chapter 4

Results

This chapter discusses the results of each computational approach of the thesis. When applicable, it compares my results to those of similar commercial products as a benchmark. Additionally, this chapter applies my computational methods on the selected corpus on ten interviews to begin the collection of a new data set that would enable the user-centered exploration of segments across interviews in the future archive.

4.1 Voice Classification & Diarization Performance

Table 4.1 lists the ten interviews I ran my audio segmentation approaches on. The "2s Model Accuracy" column lists the accuracy of the voice classification model that was trained using two second audio clips. This accuracy was calculated by counting the number of correct speaker predictions the model made out of 40 two second clips, 20 of the interviewer and 20 of the interviewee, from each side of tape recorded.

Figure 4-1 graphs the number of tape sides recorded against the model accuracy for each interview. The male interviewees are marked with blue dots, and the female interviewees are marked with red squares. The number of tape sides recorded, or the overall length of the interview, had no impact on the accuracy of the model's accuracy, but the gender of the interviewee did. Overall, male interviewees had a generally higher model accuracy than female interviewees. The male interviewees had a mean accuracy of 93.439% while the female interviewees had a mean accuracy

of 91.96%.

Interviewee	# of Tape Sides Recorded	Model Accuracy
Kofi Annan	2	97.6%
Gregory Chisholm	2	86.1%
Harvey Gantt	2	93.6%
Gloria Green	2	92.7%
Darian Hendricks	4	93.8%
Shirley Ann Jackson	3	94.8%
Denise Loyd	3	92.6%
Jennie Patrick	4	89.3%
Victor Ransom	3	96.1%
Jennifer Rudd	2	90.4%

Table 4.1: Accuracy of Voice Classification Models for Ten Interviews

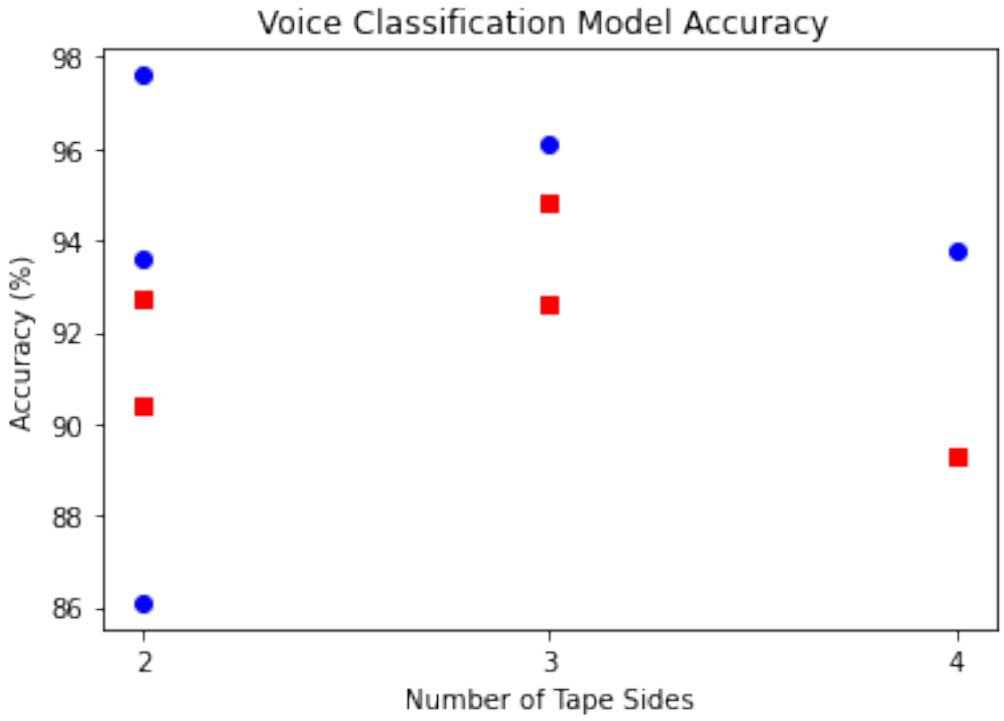

Figure 4-1: Voice Classification Model Accuracy

While I had expected female interviewees to have an average higher accuracy than the male interviewees because the interviewer, Dr. Williams, is male and it might be easier to distinguish the two voices if they were of different genders, the data showed otherwise. One possible explanation for this could be that for most of the male

interviewees, the difference between their average loudness (measured in decibels) and Dr. Williams average loudness was often greater than the same difference for the female interviewees. Figure 4-2 graphs these differences on the x-scale, with the male interviewees marked as blue dots and the female interviewees marked with red squares. Three out of five of the female interviewees had an average loudness difference less than one, while four out of five of the male interviewees had an average loudness difference greater than one.

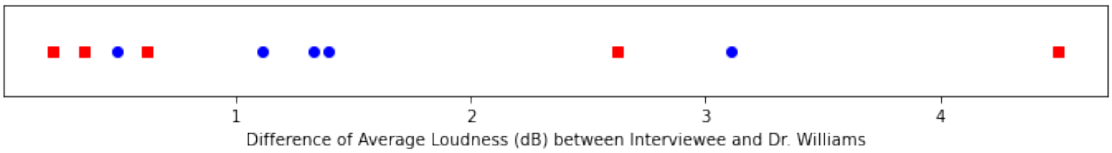

Figure 4-2: Difference of Average Loudness (dB) between Interviewee and Dr. Williams

When it transcribes an audio clip, Otter.ai performs speaker diarization. Kofi Annan's interview had the highest model accuracy, so I compared my segmentation of his interview with Otter.ai's segmentation. I generated Otter.ai's segmentation by transcribing the full audio in Otter.ai, then segmenting by the speaker. I generated my segmentation by running my speaker diarization algorithm, then transcribing each segment in Otter.ai. Table 4.2 lists the identified speaker and start time for each segmented audio clip, with my segmenting results on the left and the Otter.ai results on the right.

My segmenting approach resulted in the same number of segments as Otter.ai, and eleven of the fourteen segments had a start time within five seconds of each other. The largest time differences are in the ninth and tenth segments, where my segmenting started at 14:44 and 14:54, but Otter.ai started at 12:05 and 12:12 respectively. Otter.ai's segmentation was correct in this interview, so my segmentation was only correct for twelve out of fourteen segments, or had an 85.714% success rate.

Gregory Chisholm's interview had the lowest model accuracy rate of 86.1%, so Table 4.3, compares my results to Otter.ai's results, and lists the actual speakers and start times for this interview. My method resulted in seven segments, one of

My Speaker	My Start Time	Otter.ai Speaker	Otter.ai Start Time
Williams	0:00	Williams	0:04
Annan	0:28	Annan	0:32
Williams	4:04	Williams	4:12
Annan	4:32	Annan	4:33
Williams	5:52	Williams	5:52
Annan	6:14	Annan	6:15
Williams	8:54	Williams	8:55
Annan	9:18	Annan	9:18
Williams	14:44	Williams	12:05
Annan	14:54	Annan	12:12
Williams	15:10	Williams	15:12
Annan	15:46	Annan	15:45
Williams	15:56	Williams	15:59
Annan	16:34	Annan	16:32

Table 4.2: Segmentation Comparison of Kofi Annan's Interview (Tape 1.1)

them being four seconds of silence at the beginning that was classified at Chisholm. However, the remaining six segments all had the correct speaker classification and were all within 5 seconds of the actual start times. Counting the first segment, my method was 85.714% successful. Without counting the first segment, it would be fully successful. In comparison, Otter.ai had nine segments, with only three of them correctly classified, giving it a 33.333% success rate.

My Speaker	My Start Time	Otter.ai Speaker	Otter.ai Start Time	Actual Speaker	Actual Start Time
Chisholm	0:00			Silence	0:00
Williams	0:04	Williams	0:05	Williams	0:05
Chisholm	0:26	Chisholm	0:24	Chisholm	0:30
Williams	4:40	Williams	1:44	Williams	4:40
Chisholm	4:56	Chisholm	2:29	Chisholm	5:00
Williams	12:10	Williams	7:26	Williams	12:13
Chisholm	12:28	Chisholm	12:27	Chisholm	12:28
		Williams	13:52		
		Chisholm	14:04		
		Williams	19:05		

Table 4.3: Segmentation Comparison of Gregory Chisholm's Interview (Tape 2.1)

Based on these results, the voice classification model accuracy is not always an indicator for how accurately the speaker diarization will segment the audio. Additionally, commercial transcription products like Otter.ai may underperform in speaker diarization for low quality audio recordings.

4.2 Metadata Extraction

In Chapter 3.4, I showed the results of running named entity recognition on a question asked by Dr. Williams and an answer given by Kofi Annan using spaCy in Tables 3.2 and 3.3. When transcribing audio, Otter.ai extracts "summary keywords" from the text it creates. The MIT Museum has been using Otter.ai to transcribe some of the interviews for their exhibition and highlighted selections of these summary keywords to use as tags or metadata for each interview. Figures 4-3 and 4-4 compares the named entities extracted with spaCy with the summary keywords extracted by Otter.ai from the transcripts of the same question asked by Dr. Williams and answer given by Kofi Annan. Barely any overlap exists between the two text sets for both transcripts, other than the location "Ghana". The named entities extracted more proper nouns like "the 1940s", "UN", "Reader's Digest", and "British", while the summary keywords extracted more of the sentiment with concepts like "independence", "inhibition", and "monumental". In this case, the named entities are more useful for categorizing the people, places, organizations, and times mentioned in an audio segment while the summary keywords could be used to help categorize the topic or sentiment of a segment. However, both sets of data could be useful in the organization of the future archive. The combination of the data tells a fairly concise summary of each segment, which could be used to detect similarities between segments within one interview or across interviews. For example, these two segments both mention Ghana and the 1940s and 50s, as well as familial words like "relatives", "sisters", "brothers", "father", and "nephews". This high level of overlap between both the named entities and the summary keywords could indicate that, within this one interview, these segments could be a question and answer pair. Additionally, an audio segment from a different

interview with familial words in the summary keywords could be connected to either of these segments as a related or similar segment that discusses family as a topic.

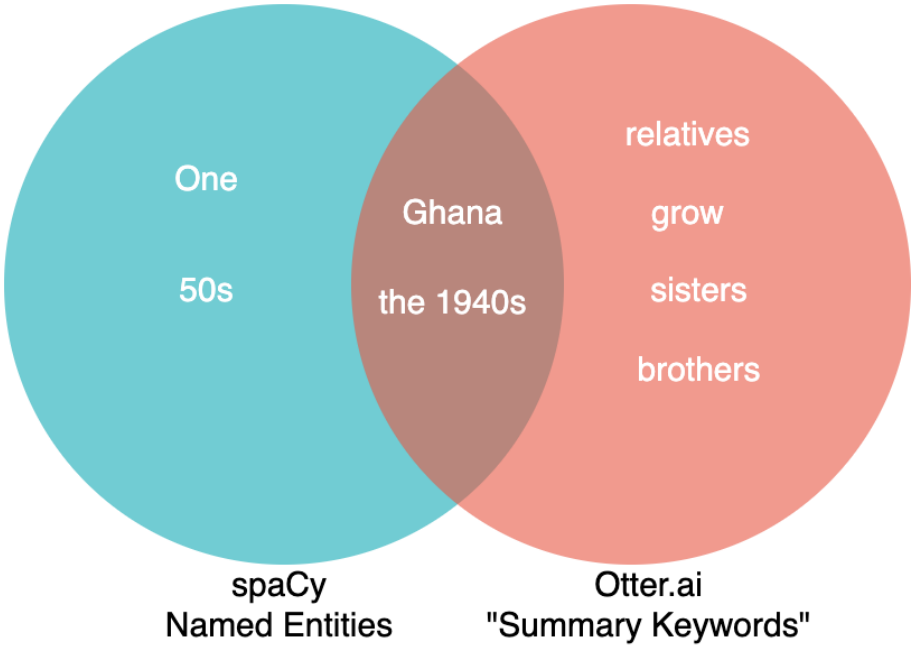

Figure 4-3: spaCy vs. Otter.ai: Dr. Williams' Question

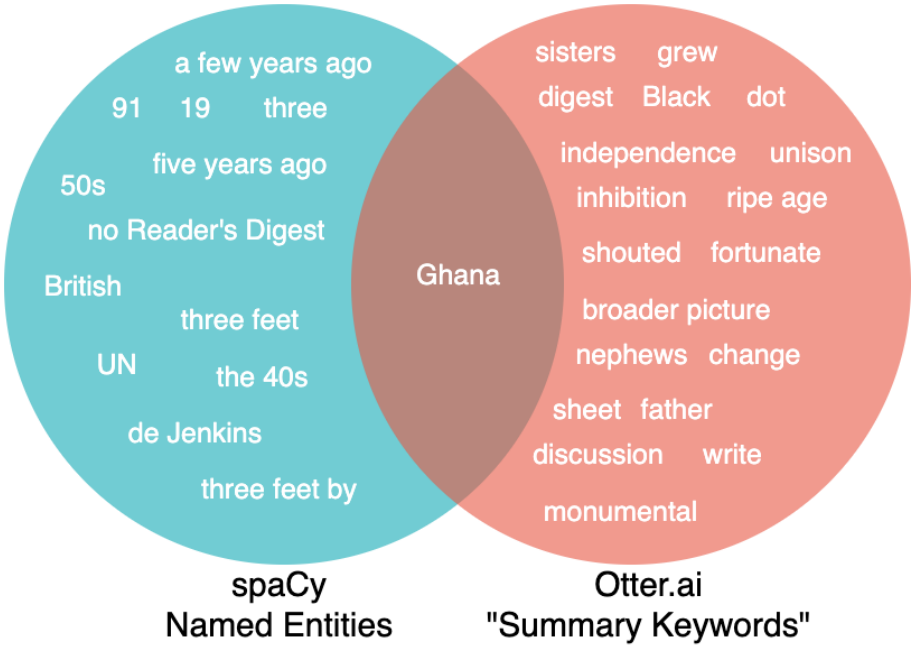

Figure 4-4: spaCy vs. Otter.ai: Kofi Annan's Answer

4.3 Topic Classification Accuracy and Understandability

Table 4.4 lists the types of pre-trained feed forward models and the text vectorization techniques used to perform topic classification in the "Model Approach" column, and the remaining columns list the data segmenting approach used to create the training, validation, and testing data. The cells list the model accuracies for each approach for each model. As a refresher, the three data segmenting approaches mentioned in Table 4.4 were:

1. Randomly select the train, validate, and test data from the full corpus of 778 transcripts

2. Separate the duplicate transcripts (i.e. transcripts tagged with more than one topic) from the non-duplicate transcripts. Pick the train and validate data from the duplicate transcripts and the test data from the non-duplicates.

3. Remove all but one of each duplicate transcript from the corpus, creating a corpus of 397 transcripts with no duplicates. Randomly select the train, validate, and test data from this new corpus.

As previously mentioned, the data in approach 1 created confusion for all of the models by training the model on some duplicates, testing on others, resulting in low accuracies in every model, regardless of the model or vectorization approach used. Approaches 2 and 3 removed the duplicate issue in different ways, but approach 2 had more data overall and resulted in overall higher accuracies than both approach 1 and 2. Models with TF-IDF vectorization tended to have higher accuracies than the models using word2vec vectorization.

The first neural net approach significantly improved on these results, creating a 70% accurate model capable of categorizing transcripts with multiple topics, as the volunteers had done with the transcripts originally. By using all of the created data, training a TF-IDF vectorizer on the data itself, and fine tuning training details, the

Model Approach	Data Segmenting Approach 1	Data Segmenting Approach 2	Data Segmenting Approach 3
Logistic Regression (TF-IDF)	14.7%	40.6%	27.8%
Logistic Regression (word2vec)	14.1%	10.599%	22.799%
Naive Bayes (TF-IDF)	26.3%	32.499%	25.3%
Naive Bayes (word2vec)	2.6%	9.999%	7.599%
SVM, sigmoid kernel (TF-IDF)	17.9%	32.499%	27.8%
SVM, sigmoid kernel (word2vec)	26.3%	31.2%	15.2%

Table 4.4: Topic Classification Model Accuracies

neural net model was able to nearly double the highest accuracy of the previously mentioned models. This approach begins to replicate how humans would interact with these audio segments and transcripts by not restricting the classification of one segment to a single topic. The accuracy is still relatively low for the level of accuracy needed to accurately predict similarities and connections between transcripts, but this could be improved with further design and iterations.

The second neural net only had a 42.5% accuracy, but the results are output in a way that is less ambiguous and more understandable to humans. I input a transcript of a segment from Darian Hendricks' interview, seen in Transcript A.4. The transcript was tagged by a volunteer with the topics "MIT", "Education", and "Society". I also input a transcript of a segment from Jennifer Rudd's interview, seen in Transcript A.5. This transcript was tagged by a volunteer with the topics "MIT', "Society", and "Inspiration". Table 4.5 shows the model's predictions of probabilities for each topic for these interviews.

For Transcript A.4, "MIT" and "Society" have the highest probabilities, both being over 0.5, and "Education" has the third highest probability with a probability of 0.384. The remaining topics all have probabilities below 0.256. In this instance, the model's probability predictions intuitively match the human tagging of this interview

in an understandable way, despite its low accuracy.

For Transcript A.5, "MIT" had the highest probability, with "Education", "Identity", and "Society" having the next highest, in that order. In this instance, the model's predictions only partially align with the human tagging.

Topic	Probability Predictions for Transcript A.4	Probability Predictions for Transcript A.5
Upbringing	0.210	0.119
Education	0.384	0.327
Society	0.501	0.192
Identity	0.265	0.256
Inspiration	0.176	0.166
Accomplishment	0.058	0.042
MIT	0.586	0.459
Career	0.179	0.072
Advice	0.061	0.073
Anecdote	0.181	0.120

Table 4.5: Topic Classification Model Predictions for Transcripts A.4 and A.5

Because the two transcripts were both tagged with "MIT" and "Society" by a human, if a user were to explore the future archive looking for segments about both MIT and society, these two segments should appear. Additionally, if the user were to begin with the segment in Transcript A.4, the ultimate goal of the archive would be to recommend the segment in Transcript A.5 to them since it is about MIT and society as well, but could also lead them to explore segments with the "Inspiration" topic. These probabilities for each topic output by the model could help with this segment recommendation and creating connections between segments across different interviews, once the model accuracy was increased.

Chapter 5

Conclusion

5.1 Further Development

I got fairly good results in my audio segmentation using my final speaker diarization approach, but pre-processing the audio files to remove some of the grainiess might improve the results of the segmentation. It did not significantly improve the results of splitting by silence, but since the voice classification model is trained on the audio files themselves, it has the potential to make improvement here. The time stamps for the segmentation would still be output so that the original, unprocessed files could still be used in the future archive. Additionally, I could perform more fine grain speaker diarization by segmenting the audio into one second clips rather than two second clips to see if the accuracy improves.

Having better transcripts could improve all the results for topic modeling and classification. While the Otter.ai generated transcripts were better than any open-source speech-to-text API generated transcripts, they were still not as accurate as a human-made transcript would be. Any gibberish in the transcripts impacted the data and results of the topic modeling and classification. One clear solution would be to find the human-made transcripts that were typed as a first draft of the final edited transcripts in *Technology and the Dream:Reflections on the Black Experience at MIT, 1941-1999* [35] and work with those. However, those transcripts only exist in paper format with the MIT Museum, and would require digitization before they could be

used for topic modeling and classification. Another option would be to pay for new human-made transcripts with services provided by companies like HappyScribe. A final, more experimental option would be to compare the Otter.ai generated transcripts with the edited transcripts from *Technology and the Dream* and use natural language processing to fill in any gaps or replace gibberish words with real words from the edited transcripts.

The tagged transcripts my volunteers and I worked on resulted in an uneven distribution of the topics. While 192 transcripts were tagged with "MIT", only 26 transcripts were tagged with "Accomplishment". Right now, the topic classification models were fairly certain when classifying a transcript with "MIT", but were less sure when classifying other topics. Generating more transcripts and ensuring an even distribution of the topics in the training, validation, and testing data used for the topic classification could increase the accuracy of the models. Additionally, only 160 of the 778 tagged transcripts were only tagged with one topic. Using more transcripts with only a single topic could also increase the accuracy of the models, as the models would have less confounding variables to work with in the training.

I only experimented with types of feed forward and RNN topic classification models, but there are more complex neural nets that have been experimented with for topic classification. Additionally, there are more pre-trained models that could be used as a starting point and fine tuned to this data set. Experimenting with these could improve the accuracies as well.

Rule based systems require immense depth of knowledge to make, which is why I did not experiment with them as a topic classification method in this thesis. However, they can be incredibly accurate in classification. If MIT Museum curators wrote patterns they noticed while developing their exhibit with this interviews, or if users of the future archive wrote any findings they discovered, those could be transformed into rules for a rule based system.

Finally, I would like to further experiment with using the results of topic classification to find similar audio segments. This could lead into a recommender system for the future archive that would assist users in exploring the interviews by topic,

similarities, or even dissimilarities.

5.2 Takeaways

Throughout this thesis, I experimented with different approaches to audio segmentation, metadata extraction, and topic detection in audio files and their transcripts. This thesis worked within the framework of goals for a future archive, and primarily the goals to allow users to explore the interviews without having to listen to them in full and in order, to allow users to explore multiple topics across multiple interviews while having enough context, and to allow users to build their own understanding of the interviews and discover connections through their exploration.

My design for speaker diarization system with a minimum length threshold for concatenating clips was the most successful audio segmentation method, outperforming commercial speaker diarization in some cases. Training the voice classification model on the audio itself before segmenting it gave the system an advantage over commercial services that would segment the audio without having heard it before. The audio segmentation by speaker resulted in audio segments that contained enough context for users of the future archive to be able to listen to segments and understand them without having to listen to the full interview. However, given an appropriately design user interface, the segmentation still allows users to listen to parts of the same interview in or out of order, should they choose to.

The named entities extracted from running named entity recognition on the transcripts combined with the summary keywords extracted during Otter.ai's transcription provide important keywords that tell the essence of an interview segment. A combination of these two data sets could be used to find connections within an interview, such as a question and answer pair, or find connections between segments across interviews.

Topic modeling and topic classification are heavily influenced by the data they are given, so it is essential to have clean data. The accuracies in my models did not exceed 70%, but the topic classification results have shown potential for computation-

ally drawing connections between audio segments across interviews. The probability predictions given each potential topic frees the topic classification model from being restricted to identifying only one topic per segment. Additionally, segments with similar probability predictions for one or more topics could be identified as similar, and this data could be used to detect connections between segments based on topic similarity. The combination of the metadata extraction and the topic classification opens up possibilities for recommender systems in the future archive to help guide user exploration, while still giving them the freedom to discover connections in their exploration.

This thesis helps support some of the key goals of the future archive computationally, but it can also help further the goals of minority archives and participatory archives. The data for topic classification in this thesis had to be manually created by volunteers, but users of the future archive could tag segments with pre-defined topics or suggest their own topics and enhance the robustness of the topic classification on these interviews. Such a system could even lead to possibilities in detecting subtopics within interviews and allow users to create fine grain connections between segments that the computer could then learn based on their own personal knowledge. Another possible feature of the future archive would be to allow users to create "playlists" of interview segments they believe are connected in some way. This could generate a data set that could then be used to train a model to automatically detect connections between segments without manually calculating similarity between topic probabilities. The participation in the future archive would benefit the computational robustness behind the archive, which could lead to greater participation if users saw how many possible connections there are, leading into a cycle benefiting the users, the archive, and the minority community whose story is being preserved, told, and listened to. If these computational methods can lead users to discover not only connections between audio segments, but also connections between the audio segments and their own experiences, then these stories could be understood, internalized, and immortalized by the users rather than silenced, marginalized, and forgotten as they have been in the past.

Appendix A

Transcripts

Full Transcript	Transcript with Stop Words Removed
Okay, I'm here with the Secretary General. One thing that would be very helpful is that could you tell us something about your family background and who your parents were their occupation. In it brothers and sisters and other relatives have known what it was like to grow up in Ghana in the 1940s and 50s.	Okay, I'm Secretary General. thing helpful tell family background parents occupation. brothers sisters relatives known like grow Ghana 1940s 50s.

Transcript A.1: Question Asked by Dr. Clarence G. Williams

Full Transcript	Transcript with Stop Words Removed
Both my parents died a few years ago. But at the ripe age of 91, for my father and 19, my mother my father did work in the commercial area. He did work for a branch of union labor was one, I became one of the directors of the company in Ghana. He also comes from a family where he could have chosen to be achieved if he wanted to. And I have three sisters and a brother. I had a twin sister who unfortunately passed away five years ago. And I have a brother here in the states that was in the business and all the sisters and nephews and nieces in Ghana, but I do have my day sister sentence who are also studying, write them out to give them a chance and together to prepare themselves for the future. And I think he raised an interesting question, what is it like growing up in Ghana, in the 40s, and the 50s. I was fortunate in that I grew up at a time when the struggle for independence, whatever it was to speak, is, as a young person growing up seeing the struggle for independence, the discussion about independence, the role of the British and when the ganja should take, taking place around me at my school, and with friends, and my father and friends were all very actively engaged in these discussions. And now was also fortunate enough to see the success of that operation. And so I grew up in an atmosphere where I thought change was possible, or was possible, and you can do things. And so I didn't have a sense of inhibition that you shouldn't have been de Jenkins, because I lived it, and I saw it happen.	parents died years ago. ripe age 91, father 19, mother father work commercial area. work branch union labor one, directors company Ghana. comes family chosen achieved wanted to. sisters brother. twin sister unfortunately passed away years ago. brother states business sisters nephews nieces Ghana, day sister sentence studying, write chance prepare future. think raised interesting question, like growing Ghana, 40s, 50s. fortunate grew time struggle independence, speak, is, young person growing seeing struggle independence, discussion independence, role British ganja take, taking place school, friends, father friends actively engaged discussions. fortunate success operation. grew atmosphere thought change possible, possible, things. didn't sense inhibition shouldn't de Jenkins, lived it, saw happen.

Transcript A.2: Answer Given by Kofi Annan, Part 1

Full Transcript	Transcript with Stop Words Removed
So you walked out with a feeling that change is possible, it can be done, how the monumental and that was a wonderful feeling for a young person to have an I record a particular one of teachers I was in this boarding school came up and put a price sheet of paper, three, three feet by three feet, with a little black.on, the right hand corner of the board and said, Boys, what do you see the about 40 of us in the class. And we all shouted in unison, a black dot. And it pulled back a step back and said so not a single one of you saw the broad white sheet of paper. You all saw the black dot that says the awful number human nature they never see the goodness of things and the broader picture was focus and don't go through life. With that I've never forgotten that often, because we are constantly doing the same no Reader's Digest it too terrible a series of articles about the UN. And I wrote to them using this example. And I said you and as a solid record of achievement, but you're focusing on the black dot. I believe your people deserve a better digest dinner.	walked feeling change possible, done, monumental wonderful feeling young person record particular teachers boarding school came price sheet paper, three, feet feet, little black.on, right hand corner board said, Boys, 40 class. shouted unison, black dot. pulled step said single saw broad white sheet paper. saw black dot says awful number human nature goodness things broader picture focus don't life. I've forgotten often, constantly Reader's Digest terrible series articles UN. wrote example. said solid record achievement, you're focusing black dot. believe people deserve better digest dinner.

Transcript A.3: Answer Given by Kofi Annan, Part 2

I looked at other student groups and saw what they were creating and said, Here we are as black students that a majority institution that has the world's eyes on it, that we could do something that puts, you know, black students on the map from MIT. And that particular concept was the creation of this black library that I talked to Professor Frank Jones about. And, you know, I guess, what I found was, again, that high school experience of there was a lot of alienation that also went on at MIT between black students. There was this whole thing about blacks, if you hung around white students who thought you were white, you know, if you didn't live in Chocolate City, what was your problem? How come you didn't? Why didn't you want to apply to Chocolate City, you know? So there was a lot of that going on, too. And I, I, I was shocked that here were intellectual, black students, people I never got to deal with when I was a high school student. People I even got to a point of perceiving didn't exist, you know. And now I got to college in order to deal with with them, and interact, and feel that we had a closeness or bond about the same type of high school experiences that were alienation from our own communities, etc. And I think what I ran into was more denial. I ran into a lot of students who denied they ever went through that experience. They said, Oh, I never was alienated when I was in high school. Yeah, all my friends were white, but all the black students love me too. And I was just like, I was like, I don't think so. But if you say, you know, and then I found that the things that the things that most of them would talk about was their own negative experiences in high school. They ended up repeating in college among their own community, you know, black against black and you know, I also found that I always considered this an interesting observation.

Transcript A.4: Segment from Darian Hendricks' Interview

All this happened after 1968 When Martin Luther King was assassinated. So at MIT, also, you know, Shirley and I got together and founded the Black Student Organization. And I came out of that response to Martin Luther King's assassination, and wanting to form some kind of group of black students on campus to get together in the positive sense of graduate mentioned that because they know that, yes. And just the backtrack a little bit about, you know, kind of the militancy I was getting into, when I was at MIT, you know, I did venture off campus a few times. And I remember going into Roxbury to hear Leroy Jones, you know, give a presentation with a sort of poetry but political message and kind of dramatic presentation and getting the audience involved in it. And I remember the line he always said about how they started picking up the black radio station on the, on the radio as they drove up from Newark, you know, and he says, Ah, we're approaching civilization. But they all closed the doors and they didn't allow whites in the room, you know. And then another time I was over in Roxbury, somewhere, lyses Stokely Carmichael speak, while the student and it might have been about the time I got my afro, you know, the senior year and so on. So, you know, when Martin was assassinated, and Malcolm had been killed my freshman year at MIT, that was in 19, spring of 1965, then Martin in 1968. So that's all during my college years, and then in between, my first trip to New York City, was with the big anti war rally, probably in 1967, we had a whole contingent of buses from New England, and that was my first experience demonstrating in the streets, all to all people feeling the UN Plaza, you know, and that was, that was part of that whole anti war era. And that sort of, you know, got me a little bit, you know, who the size in that area now, Vietnam for the Vietnamese, and later on South Africa for the South Africans? Yes. So, in any case, when I was in, left MIT right, after all of that, went to Wesleyan, they to you know, were had developed a black student organization, had the afro am Institute already, and we take over building and for us for a year and a half or so, and made certain demands, you know, administration capitulated on, you know, came to some understanding.

Transcript A.5: Segment from Jennifer Rudd's Interview

Appendix B

Software Libraries and Packages

This appendix lists the software libraries, packages, tools, and models used and discussed throughout this thesis that are outside of the default packages included in Python. Each item includes a link to the full documentation of the library or package.

B.1 Audio Segmentation

- **PyDub:** Used to detect silences in audio and segment audio
  `https://github.com/jiaaro/pydub`

- **ffmpeg:** Dependency of PyDub used to open and save audio files
  `http://www.ffmpeg.org`

- **Librosa:** Used to extract features from audio for training a neural net
  `https://librosa.org`

B.2 Speech to Text

- **Google Speech-to-text API:** Attempted to use for speech-to-text conversion
  `https://cloud.google.com/speech-to-text/`

- **Otter.ai** Commercial service used to generate transcripts
  `https://otter.ai`

- **HappyScribe:** Commercial service mentioned for speech-to-text generated transcripts and human-made transcripts
  `https://www.happyscribe.com`

B.3 Machine Learning

- **Keras:** Used to build and train neural nets for the voice classification and topic classification models
  `https://keras.io`

- **Tensorflow:** Dependency of Keras that the Keras API interacts with
  `https://www.tensorflow.org`

- **Pandas:** Used to convert training, validation, and testing data into a format that could be input into a neural net
  `https://pandas.pydata.org`

B.4 Natural Language Processing

- **spaCy:** Used to perform NLP techniques
  `https://spacy.io`

- **Natural Language Tool Kit:** Used to perform NLP techniques
  `https://www.nltk.org`

B.5 Topic Modeling and Classification

- **Gensim:** Used to perform LDA and LSI
  `https://radimrehurek.com/gensim/`

- **Scikit-Learn:** Used for pre-trained TF-IDF vectorizer and pre-trained logistic regression, Naive Bayes, and support vector machine models
  `https://scikit-learn.org/stable/`
- **GloVe:** Used the pre-trained vectorizer for a topic classification model
  `https://nlp.stanford.edu/projects/glove/`

Bibliography

[1] "The MIT Black History Project". https://www.blackhistory.mit.edu/about. Accessed: 2021-08-11.

[2] J. Arias. "How to build a Neural Network for Voice Classification". https://towardsdatascience.com/how-to-build-a-neural-network-for-voice-classification-5e2810fe1efa, May 2020. Accessed: 2021-09-14.

[3] B. Bengfort, R. Bilbro, and T. Ojeda. *Applied Text Analysis with Python.* O'Reilly Media, Inc., 2018.

[4] S. Bird, E. Klein, and E. Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* O'Reilly Media, Inc., 2009.

[5] D.. Blei, A. Ng, and M. Jordan. "Latent Dirichlet Allocation". *Journal of Machine Learning Research*, 3:993–1022, January 2003.

[6] A. Burdick, J. Drucker, P. Lunenfeld, T. Presner, and J. Schnapp. *Digital_Humanities*, chapter "A Short Guide to the Digital_Humanities", pages 121–136. MIT Press, Cambridge, MA, 2012.

[7] B. Butterworth. "Don't Call It a Comeback: Cassettes Have Sounded Lousy for Years (And Still Do!)". https://www.nytimes.com/wirecutter/blog/cassettes-comeback-sound-lousy/, October 2021. Accessed: 2022-01-10.

[8] F. Chollet et al. *Keras.* https://keras.io, 2015.

[9] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. "Indexing by Latent Semantic Analysis". *Journal of the American Society for Information Science*, 41(6):391–407, September 1990.

[10] A. Flinn. "Community Histories, Community Archives: Some Opportunities and Challenges". *Journal of the Society of Archivists*, 28:151–176, 2007.

[11] Z. Harris. "Distributional Structure". *WORD*, 10:146–162, 1954.

[12] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. *spaCy: Industrial-strength Natural Language Processing in Python*, 2020.

[13] J. Kim, C. Liu, R. Calvo, K. McCabe, S. Taylor, B. Schuller, and K. Wu. "A Comparison of Online Automatic Speech Recognition Systems and the Nonverbal Responses to Unintelligible Speech". arXiv:1904.12403, April 2019.

[14] P. Liu, X. Qiu, X. Chen, S. Wu, and X.-J. Huang. "Multi-timescale long short-term memory neural network for modelling sentences and documents". In *Proceedings of the 2015 conference on empirical methods in natural language processing*, page 2326–2335, 2015.

[15] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, 2008.

[16] B. McFee, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, D. Ellis, J. Mason, E. Battenberg, S. Seyfarth, R. Yamamoto, V. Andreevichmorozov, K. Choi, J. Moore, R. Bittner, S. Hidaka, Z. Wei, D. Hereñú, F. Stöter, P. Friesch, A. Weiss, M. Vollrath, T. Kim, and Thassilo. *librosa/librosa: 0.8.1rc2*, May 2021.

[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean. "Efficient Estimation of Word Representations in Vector Space". *1st International Conference on Learning Representations*, arXiv:1301.3781, 2013.

[18] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao. "Deep Learning Based Text Classification: A Comprehensive Review". *ACM Computing Surveys*, 54:1–40, 2021.

[19] D. Newman, J. Lau, K. Grieser, and T. Baldwin. "Automatic Evaluation of Topic Coherence". *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, pages 100–108, 2010.

[20] The pandas development team. *pandas-dev/pandas: Pandas.* `https://doi.org/10.5281/zenodo.3509134`, 2021.

[21] T. Park, N. Kanda, D. Dimitriadis, K. Han, S. Watanabe, and S. Narayanan. "A Review of Speaker Diarization: Recent Advances with Deep Learning". *Computer, Speech, and Language*, 72:101317, November 2021.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. "Scikit-learn: Machine Learning in Python". *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[23] J. Pennington, R. Socher, and C. Manning. "GloVe: Global Vectors for Word Representation". *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[24] M. Porter. "Snowball: A Language for Stemming Algorithms". `http://snowball.tartarus.org/texts/introduction.html`, October 2001. Accessed: 2021-01-09.

[25] S. Prabhakaran. "Topic Modeling with Gensim (Python)". `https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/#6importnewsgroupsdata`, March 2021. Accessed: 2021-12-17.

[26] R. Řehůřek and P. Sojka. "Software Framework for Topic Modelling with Large Corpora". In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, May 2010.

[27] J. Riley. *Understanding Metadata: What is Metadata, and What is it For?: A Primer*. National Information Standards Organization (NISO), Baltimore, MA, 2017.

[28] J. Robert. *Pydub*. `https://github.com/jiaaro/pydub`, 2011.

[29] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, NJ, 3 edition, 2010.

[30] M. Röder, A. Both, and A. Hinneburg. "Exploring the Space of Topic Coherence Measures". In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408. ACM, 2015.

[31] K. Shilton and R. Srinivasan. "Participatory Appraisal and Arrangement for Multicultural Archival Collections". *Archivaria*, 63:87–101, September 2007.

[32] S. Tomar. "Converting video formats with FFmpeg". *Linux Journal*, 2006(146):10, 2006.

[33] Q. Wang, C. Downey, L. Wan, P. Mansfield, and I. Lopez Moreno. "Speaker Diarization with LSTM". In *Proceedings of the IEEE International Conference on Acoustics, Speech, Signal Processing*, page 5239–5243, 2018.

[34] J. Watkinson. *The Art of Sound Reproduction*. Focal Press, Woburn, MA, 1998.

[35] C. Williams. *Technology and the Dream: Reflections on the Black Experience at MIT, 1941-1999*. The MIT Press, Cambridge, MA, February 2001.