System Requirements Specification

for

RTube Kaldi Research Team

Version 2.0

Prepared by

Tabitha O'Malley, Milan Haruyama, David Serfaty,

Tahmina Tisha, Adam Gallub

CS 490 RTube Kaldi Research Team Fall 2023

10/31/2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Tabitha, Milan, Tisha, David, Adam, Max	09/29/23	Start the document	V1
Tabitha	10/25/23	Formatting Revision History Editing/Formatting Appendix Writing Section: 1.5	V2.1
Tabitha	10/26/23	Writing Sections: 1.2, 2.2, 2.5, 3.1	V2.2
Tabitha	10/27/23	Writing Requirements: 3.1	V2.3
Milan	10/27/23	Writing and Editing Requirements: 3.1	V2.4
David	10/27/23	Writing and Editing Requirements: 3.1	V2.5
Tabitha	10/29/23	Writing Section: 2.3, 2.4, 2.5	V2.6
David	10/29/23	Writing Section:2.3, 2.4, 2.5	V2.7
Tabitha	10/30/23	Writing Section: 4, 4.1, 4.2, 4.3, 4.4, 5.1, 5.2, 3	V2.8
Milan	10/30/23	Editing All Sections Writing Section: 5.1, 5.2, 2.1	V2.9
David	10/30/23	Writing Section: 2.1, 5.3	V2.10
Tabitha	10/31/23	Update Model Editing Sections	V2.11
Milan	10/31/23	Editing All Sections	V2.12
David	10/31/23	Editing, 2.1, 3, Appendix A, 4.1, 4.2, 4.1.3, 4.2.3, 4.3, 4.3.3, 2.2 Writing Section: 5.2	V2.13

1. Introduction

1.1 Purpose

The RTube web application shall provide an interface for users to track the flight paths of aircrafts, and to listen to live ATC tower radio chatter with the option to transcribe speech to text in real time. The live speech-to-text transcription is performed using an ASR model developed using the Kaldi ASR toolkit, which is the focus of this document.

1.2 Document Conventions

Primary Body	**Times New Roman, 11 point**
Section Headers	**Times New Roman, 18 point, bold**
Subsection Header	**Times New Roman, 14 point, bold**
Pending Review/Update	**Yellow highlight**
Complete Overhaul	**Red highlight**
Completed	**Green highlight**
Important Terms	**Bold text within document body**

1.3 Intended Audience and Reading Suggestions

The intended audience for the RTube web application is aircraft pilots and ATC operators. One of the more specific applications for RTube shall be for ERAU instructors and student pilots to evaluate communications between the student and ATC operators. This document contains a detailed description of the product and the requirements given by the customer.

In order to improve understanding of this document and the RTube project itself, it is highly recommended to read documentation related to phonetics, aviation phraseology, ASR, NLP, and the Kaldi toolkit.

1.4 Product Scope

As an aircraft pilot, learning to communicate with Air Traffic Control (ATC) is a daunting task. Despite being designed to mitigate miscommunication, aviation phraseology is highly intricate and requires hundreds of hours of training to learn its idiosyncrasies. While there exist a few resources (such as the website LiveATC) for student pilots to study aviation phraseology, these resources do little to accommodate the major learning curve present, especially since they do not provide live transcriptions of speech for student pilots to read.

As such, the RTube web application shall bridge the learning gap faced by many student pilots by providing the ability to transcribe live ATC transmissions into text in real time, as well as providing a live interface for users to track the flight paths of aircrafts. The live speech-to-text transcription is performed using an automatic speech recognition (ASR) model developed using the Kaldi ASR toolkit. With the ability to transcribe speech to text in real time, student pilots can dramatically reduce the amount of training required to understand spoken aviation phraseology. In addition, the live interface helps students better understand different contexts that specific phrases are used in.

1.5 References

Kaldi Team Product Vision Statement. Kaldi Team. 19 September 2023. Kaldi Drive.

“About Pandas.” *Pandas*, pandas.pydata.org/about/. Accessed 26 Oct. 2023.

“LiveATC FAQ.” *LiveATC.Net - Listen to Live Air Traffic Control over the Internet!*, www.liveatc.net/faq/. Accessed 26 Oct. 2023.

“A Python Library to Read/WRITE EXCEL 2010 Xlsx/XLSM Files¶.” *Openpyxl*, openpyxl.readthedocs.io/en/stable/. Accessed 26 Oct. 2023.

“What Is Kaldi?” *Kaldi*, www.kaldi-asr.org/doc/about.html. Accessed 26 Oct. 2023.

2. Overall Description

2.1 Product Perspective

The product developed by the Kaldi Research Team shall be an ASR model created using the Kaldi ASR Toolkit. The toolkit was first produced by Johns Hopkins University in 2007, and comes packaged with a sample ASR model that serves as a baseline for this project.. Though a self-contained product on its own, the ASR model developed by the Kaldi Research Team shall be combined in the future with the NeMo project in the form of the RTube web application. The RTube web application shall use the ASR model to transcribe live ATC transmissions in real time. Since this project solely focuses on ASR development, the scope is significantly more limited than that of the NeMo project.

2.2 Product Functions

2.2.1 Receive .WAV audio files for processing.
2.2.2 Transcribe audio into text.
2.2.3 Output a .txt file with the words in the audio file.

2.3 User Classes and Characteristics

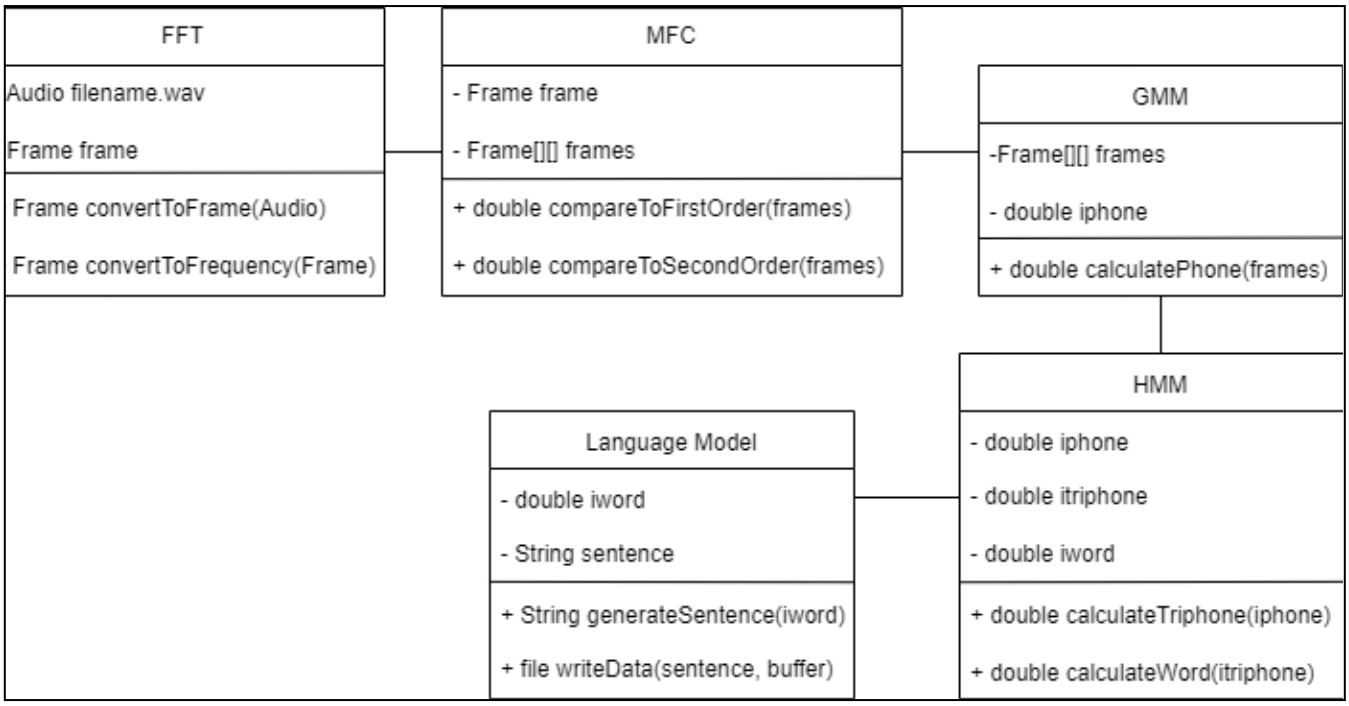

Figure 1: Kaldi Toolkit Sample ASR Model Class Diagram V1.3

FFT: Fast Fourier Transform - responsible for partitioning audio into frames, and converting audio data from change-over-time to change-over-frequency.

- Inputs: .WAV files
- Outputs: Frames

MFC: Mel-Frequency Cepstrum - responsible for converting frames into an array of MFCCs.

- Inputs: Frames
- Outputs: Array of MFCCs.

GMM: Gaussian Mixture Model - responsible for converting the MFCCs to phone indices.

- Inputs: Array of MFCCs
- Outputs: Phone Index

HMM: Hidden Markov Model - responsible for converting phone indices to triphones indices, and triphones indices to word indices.

- Inputs: Phone Index
- Outputs: Word Index

Language Model: responsible for predicting words based off of previously predicted words in order to create sentences.

- Inputs: Word Index
- Outputs: Sentences

2.4 Operating Environment

The ASR model developed with the Kaldi ASR Toolkit shall operate through the RTube web application. The ASR model connects to the RTube web application through the Python Flask API.

2.5 Design and Implementation Constraints

Design Constraints:

- Minimum of 12 GB of VRAM for ASR model to run in an hour or less.
- Minimum storage size of 12.5 GB (13,479,809,024 bytes).

Implementation Constraints:

- Input file shall be a .WAV file
- Audio shall be in General American English.

2.6 Assumptions and Dependencies

The development of the ASR model to be used in the RTube web application is constrained by several factors. These include but are not limited to: the assumption that all speech from aircraft pilots and ATC is both clear and direct; the assumption that all audio input shall be in General American English; the assumption that the audio to be transcribed contains low radio interference and background noise; the assumption that ATC transmissions are obtained from the Daytona Beach International Airport (DAB); the assumption that the RTube web application is constrained to the state of Florida; and the assumption that 30 hours of training data is sufficient to produce a transcription accuracy of at least 80%.

The development of the aforementioned ASR model is also dependent on the developers having computers with enough video memory and storage to both store and run the Kaldi ASR toolkit in order to develop the ASR model.

3. Software Interface Requirements

.WAV file	Req. 1.1.1	The system shall only accept .WAV files as input.
Fast Fourier Transform (FFT)	Req. 2.1.1	The system shall partition the .WAV file into 25ms frames.
	Req. 2.1.2	The system shall use a FFT to convert the audio data within the frames from change-over-time to change-over-frequency.
	Req. 2.2.1	The system shall obtain each frame in 10 millisecond intervals starting from time equals 0.
	Req. 2.2.2	The system shall use the formula [10n, 10n + 25] to obtain the closed time intervals of each frame (e.g., [0, 25], [10, 35], [20, 45], …).
Mel-Frequency Cepstrum (MFC) & Mel-Frequency Cepstral Coefficients (MFCCs)	Req. 3.1.1	The system shall maintain the order of the frames.
	Req. 3.1.2	The system shall convert each frequency segment frame into a MFC.
	Req. 3.2.1	The system shall partition each frame into 12 segments.
	Req. 3.2.2	The system shall append an additional “common” segment.
	Req. 3.3.1	The system shall convert all 13 segments into MFCCs.
	Req. 3.3.2	The system shall derive the first order derivative of the 13 MFCCs.
	Req. 3.3.3	The system shall derive the second derivative of the 13 MFCCs.
	Req. 3.3.4	The system shall produce a 39-dimensional array.
	Req. 3.3.5	The system shall index each dimension by time in 10 millisecond increments (e.g., n = 1 represents 10 milliseconds).
	Req. 3.3.6	The system shall use the first set of thirteen dimensions to represent thirteen the MFCCs.
	Req. 3.3.7	The system shall use the second set of thirteen dimensions to represent the first order derivatives of the MFCCs.
	Req. 3.3.8	The system shall use the third set of thirteen dimensions to represent the second order derivatives of the MFCCs.
Gaussian Mixture Model (GMM)	Req. 4.1.1	The system shall use a GMM to plot the data from the MFC to approximate the phones.
	Req. 4.1.2	The system shall compare the location of the point created by the MFC to the different phone locations.
	Req. 4.1.3	The system shall return the numerical index of the closest phone via a probability calculation in 39 dimensions.
	Req. 4.2.1	The system shall chart one index of the array from the MFCC at a time.
	Req. 4.3.1	The system shall only chart the points from the phone database once.

Hidden Markov Model (HMM)	Req. 5.1.1	The system shall use the HMM to build sets of triphones from the phone indices given by the GMM.
	Req. 5.1.2	The system shall return the numerical index of the closest triphone.
	Req. 5.2.1	The system shall use a second HMM to approximate the words by combining the previously constructed triphones.
	Req. 5.2.2	The system shall return the numerical index of the closest word.
	Req. 5.3.1	The system shall maintain the order of the phone indices.
	Req. 5.3.2	The system shall maintain the order of the triphone indices.
Language Model	Req. 6.1.1	The system shall use the Language Model to predict the current word based on the lexicon and the two previous words.
	Req. 6.1.2	The system shall use the labels from the language data to train the Language Model.
	Req. 6.2.1	The system shall filter out unknown words and copy them into a non-lexicon library.
.txt file	Req. 7.1.1	The system shall only write transcribed words into a .txt file.
	Req. 7.2.1	The system shall not write punctuation or grammatical marks into the .txt file.
	Req. 7.3.1	The system shall transcribe words found in the lexicon in uppercase.
	Req. 7.4.1	The system shall transcribe unknown words in lowercase based on the closest approximation.

4. System Features

The following section goes into detail about the system features mentioned in Section 2.2 of this document. The section begins with the reception of .WAV files, followed by the transcription of the audio into text, and ends with the output of the transcribed text.

4.1 Receive .WAV audio files for processing

4.1.1 Description and Priority

Receiving the .WAV file from the user is of high priority. Without it, the sample ASR model cannot transcribe audio into text, ergo cannot proceed to the next steps.

4.1.2 Stimulus/Response Sequences

The user shall input .WAV files into the sample ASR model. The model will respond to the file input by starting the transcription of the audio.

4.1.3 Functional Requirements

4.1.3.1 The system shall throw an exception upon the input of a file other than .WAV.

4.1.3.2 The system shall initiate the transcription process upon a .WAV being inputted.

4.1.3.3 The system shall throw an exception upon multiple files being inputted at once.

4.1.3.4 The system shall only accept input from a command-line interface (CLI).

4.2 Transcribe audio into text

4.2.1 Description and Priority

The purpose of the ASR model is to transcribe audio into text by utilizing a database of frequencies, phones, triphones, and words, as well as a series of models for each. For this project, transcription is a high priority feature.

4.2.2 Stimulus/Response Sequences

UC1	2	Divide into frames of 25 milliseconds, every 10 milliseconds, allowing for 3 frames to overlap
	3	Converts frames from time-change to frequency-change
	4	Calculates the changes in frequency between frames from up to 2 time derivations before and after
	5	Calculates the most probable phone based on the changes in frequency
	6	Builds triphones using the most probable phones
	7	Builds words using the most probable triphones
	8	Uses the previous two words to predict the following word
	9	Uses the language model to assemble the words into sentences

4.2.3 Functional Requirements

4.2.3.1 The system shall partition the audio data into 25-millisecond long frames in 10 millisecond intervals starting from time equals 0 milliseconds.
4.2.3.2 The system shall convert the audio data in the frames from change-over-time to change-over-frequency.
4.2.3.3 The system shall partition each frame into 12 segments and 1 “common” segment.
4.2.3.4 The system shall convert all 13 segments of each frame into MFCCs.
4.2.3.5 The system shall derive the first order derivative of all 13 MFCCs.
4.2.3.6 The system shall derive the second order derivative of all 13 MFCCs.
4.2.3.7 The system shall create a 39-dimensional array.
4.2.3.8 The system shall use the first set of 13 dimensions to store the original 13 MFCCs.
4.2.3.9 The system shall use the second set of 13 dimensions to store the first order derivatives of the MFCCs.
4.2.3.10 The system shall use the last set of 13 dimensions to store the second order derivatives of the MFCCs.
4.2.3.11 The system shall plot the data from each frame, element by element, using a GMM to determine the most probable phone.
4.2.3.12 The system shall use the GMM to return the index of the most probable phone from the phone database.
4.2.3.13 The system shall use the indices given by the GMM to construct triphones in the HMM.
4.2.3.14 The system shall insert the triphones created in the first HMM into a second HMM to assemble a word.
4.2.3.15 The system shall use the Language Model to predict which words were said by the word constructs from the HMM and the lexicon.
4.2.3.16 The system shall use the Language Model to assemble sentences using the two most recently constructed words.

4.3 Output a .txt file with the words in the audio file

4.3.1 Description and Priority

The sample ASR model shall output a .txt file containing the transcribed words from the audio file.

4.3.2 Stimulus/Response Sequences

UC1	10	Transfers sentences to a text file

4.3.3 Functional Requirements

4.3.3.1 The system shall write the sentence output from the Language Model into a .txt file called "out.txt".
4.3.3.2 The system shall store "out.txt" in the folder: /model_name/s5.
4.3.3.3 The system shall transcribe words stored in the lexicon in uppercase.
4.3.3.4 The system shall transcribe words not stored in the lexicon in lowercase and spelled phonetically.
4.3.3.5 The system shall not transcribe punctuation or grammatical markings.
4.3.3.6 The system shall overwrite "out.txt" with new data upon every instance of the model.
4.3.3.7 The system shall append transcribed words not stored in the lexicon into a separate library.
4.3.3.8 The system shall not write to "out.txt" if the audio file was not in the .WAV format.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The overarching goal of the Kaldi Research portion of the RTube project is to create an ASR model with the ability to transcribe live ATC transmissions in real time. As such, the following performance requirements are necessary in order to approach the speed and accuracy required to perform said transmissions in real time.

Performance	Req. 9.1.1	The system shall have a transcription accuracy of 80%.
	Req. 9.1.2	The system shall return a text file “out.txt” within 5 minutes of activation.

5.2 Safety Requirements

Due to the importance of student pilots to read accurate transcriptions when listening to live ATC transmissions, the following safety requirements are necessary to ensure the ASR model accurately transcribes inputted audio.

Safety	Req. 10.1.1	The system shall filter out words that are not in the lexicon.
	Req. 10.1.2	The system shall insert filtered words into a non-lexicon library.
	Req. 10.1.3	The system shall output the percentage of words not found in the lexicon.

Appendix: Glossary

Term	Definition
ATC	***Air Traffic Control;*** the service that elicits communications between pilots and helps to prevent air traffic accidents.
ASR	***Automatic Speech Recognition;*** the ability for computers to recognize and translate spoken speech.
CLI	***Command-Line Interface;*** text-based interface that allows interaction from the user to the computer program.
DNN	***Deep Neural Network;*** a machine learning technique that represents learning and processing data in artificial neural networks.
ERAU	***Embry Riddle Aeronautical University;*** an aviation-centered university located in Daytona Beach, Florida.
FFT	***Fast Fourier Transform;*** algorithm used to obtain the spectrum or . frequency content of a signal.
GMM	***Gaussian Mixture Model;*** used to model the transition of phones.
HMM	***Hidden Markov Model;*** used to model transitions of tri-phones and words.
IPA	***International Phonetic Alphabet:*** an alphabetic system of phonetic notation developed by the International Phonetic Association; used to represent speech sounds in a standardized format
Lexicon	*pertaining to speech;* a library of words that are understood by the language model
MFC	***Mel-Frequency Cepstral;*** algorithm used to transition from change-over-frequency to a 39 dimension array.
MFCC	***Mel-Frequency Cepstral Coefficients;*** the units used by the MFC.
NLP	***Natural Language Processing;*** the culmination of computer science, linguistics, and machine learning.
Phone	*pertaining to speech;* a distinct speech sound or gesture;
Phoneme	*pertaining to speech;* a set of phones that can distinguish one word from another
Triphone	*pertaining to speech;* a sequence of three consecutive phonemes
WER	***Word Error Rate;*** the rate at which error in words occurs